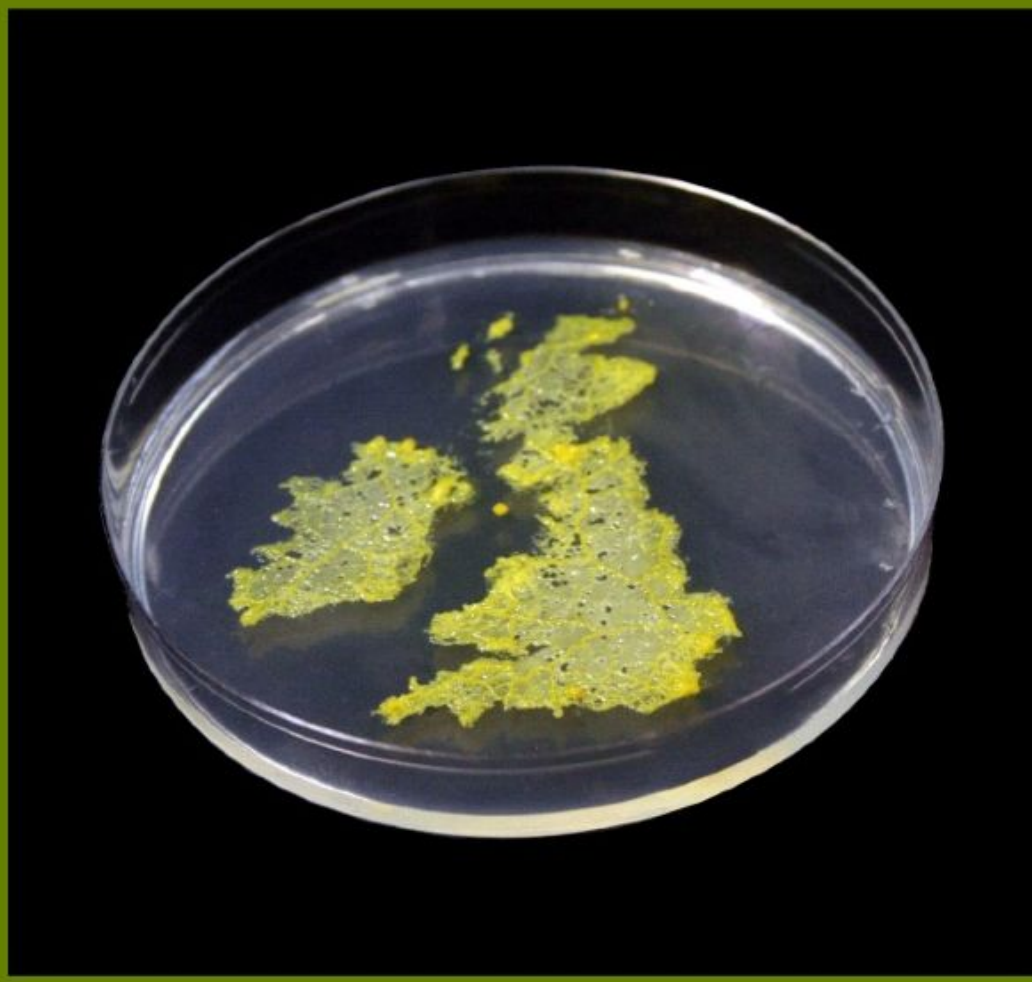
ARTIFICIAL LIFE XI



Proceedings of the Eleventh
International Conference on
the Simulation and Synthesis
of Living Systems

edited by
Seth Bullock, Jason Noble,
Richard Watson, and Mark A. Bedau

Artificial Life XI

Artificial Life XI

Proceedings of the Eleventh International Conference
on the Simulation and Synthesis of Living Systems

edited by Seth Bullock, Jason Noble, Richard Watson, and Mark A. Bedau

An ISAL Book

The MIT Press
Cambridge, Massachusetts
London, England

creative commons

Attribution-Noncommercial-No Derivative Works 3.0 United States

You are free:



to Share — to copy, distribute and transmit the work

Under the following conditions:



Attribution — You must attribute the work in the manner specified by the author or licensor (but not in any way that suggests that they endorse you or your use of the work).



Noncommercial — You may not use this work for commercial purposes.



No Derivative Works — You may not alter, transform, or build upon this work.

With the understanding that:

Waiver — Any of the above conditions can be **waived** if you get permission from the copyright holder.

Other Rights — In no way are any of the following rights affected by the license:

- Rights other persons may have either in the work itself or in how the work is used, such as **publicity** or privacy rights.

Notice — For any reuse or distribution, you must make clear to others the license terms of this work. The best way to do this is with a link to this web page.

Preface

The Eleventh International Conference on Artificial Life marks the twenty-first birthday of the conference series, which was founded in 1987 by Chris Langton at the Santa Fe Institute. As you might expect, over twenty-one years the community has grown, matured and stabilised around some key ideas, individuals and questions. However, while artificial life now, as then, continues to investigate the fundamental properties of living systems through simulating and synthesizing biological entities and processes in artificial media, there are signs that the field may be on the cusp of a second wave driven by new developments in molecular, cellular and systems biology, and renewed widespread interest in complex systems of many kinds.

The rise of synthetic biology (constructing artificial living cells, engineering with living biological materials, etc.) and systems biology, with its focus on biological organisations above the gene (e.g., proteomics, metabolomics, etc.), means that topics proper to artificial life are becoming key research areas across science and engineering. Additionally, the kinds of agent-based simulations and complex systems methodologies pioneered within artificial life are growing in importance within a large number of fields (ecology, economics, sociology, transport, etc.). This makes ALIFE XI a potential watershed event at which artificial life has the opportunity to engage with, and offer a stimulating home for, some of the largest and most interesting of modern research questions.

Over the last twenty-one years, the ALIFE conference series has, along with its European sister conference ECAL, played an important role as a meeting place for researchers from diverse disciplines. Biologists, physicists, chemists, computer scientists, engineers, economists, linguists, geographers, psychologists, mathematicians, anthropologists, philosophers, musicians and artists have come together to exchange ideas and inspiration. In doing so, many have found an informed audience for work that may have been considered peripheral to the interests of their home disciplines. Maintaining this diversity of ideas, tools, approaches and cultures is something that we feel is extremely valuable.

One measure of artificial life's success is that the community's research has had an influence on the mainstream work within adjacent disciplines. What was once somewhat marginal is increasingly central: witness the growth of interest in complexity, self-organisation, adaptation and simulation across a broad range of fields. Moreover, Alife has had a hand in the genesis of entirely new research communities in areas such as unconventional computing and self-* computing. If Alife is to remain healthy, there must be a continuing reason for researchers to keep coming back to the melting pot within which some of these ideas were incubated.

With this in mind, our aim was to make the meeting as open and attractive to researchers from as wide a range of disciplines as possible—our watchword was "inclusivity". In order to involve as many relevant kinds of academics as possible at ALIFE XI we have experimented with radical changes to the format and delivery of the conference. By allowing both full papers and abstracts to be submitted for presentation at the conference and inclusion in this proceedings, we hoped to engage not just with ALIFE's established core community, but also researchers from disciplines where conferences are not associated with published proceedings, e.g., many parts of biology.

For such conferences it is standard practice to submit abstracts only and for no lasting record of the conference to be published. We were also conscious that there are many Alife academics for whom a full conference paper may not look like a good return on the investment of time and effort required. For both of these groups of people, the opportunity to make a presentation based on a 500-word abstract may make the difference between attending the conference or not. The fact that ALIFE XI received roughly the same number of full paper submissions as ALIFE IX and X, but also received a similar number of abstracts on top of that is evidence that this has indeed been the case. Given this, we believe that the new format will maintain the character and quality of ALIFE whilst lowering the "barriers to entry" that may have existed in the past, opening it up to a more diverse and representative community of researchers.

ALIFE XI sees many firsts for the conference series. For the first time in its history, the conference visits Europe, being hosted in the United Kingdom by the University of Southampton at the nearby historic city of Winchester, known for its 11^{th}-century cathedral and 12^{th}-century castle. For the first time, the conference will be truly multi-track with around 180 talks taking place over three-and-a-half days. While it might not be ideal for delegates to have to choose between competing parallel sessions, we feel that allocating every accepted submission an oral presentation offers the best opportunity for each researcher to present their work clearly and effectively to the delegates that have a substantive interest in it. Also for the first time, the conference proceedings will be published by MIT Press as an open-access online volume containing all accepted papers and abstracts. Rather than receiving an expensive and heavy book at registration, conference delegates (and anyone else) will instead be able to freely access the entire conference proceedings online. In addition to obvious economic and environmental advantages, this arrangement should enable delegates to make informed decisions regarding which talks to attend at the conference, and should also increase the impact of conference papers outside the immediate community of delegates attending the conference.

We received a total of 275 submissions (145 full papers and 130 abstracts), each of which was reviewed by three referees. Of these, 95 full papers and 85 abstracts were selected for presentation at the conference and publication in this proceedings. At the time of writing we have registered over 250 delegates for the meeting itself. Our thanks to all those who submitted papers and abstracts to the conference.

We would like to take this opportunity to thank all who served on the programme committee for their assistance in shaping the direction that the field is taking and helping achieve a high level of quality across the accepted submissions:

Andrew Adamatzky
Fernando Almeida e Costa
Takaya Arita
Wolfgang Banzhaf
Lionel Barnett
Randall Beer
Josh Bongard
C. Titus Brown
Joanna Bryson
Chris Buckley
Larry Bull
Lola Ca namero
Angelo Cangelosi
Peter Cariani
John Cartlidge
Alastair Channon
Andy Clark
Dave Cliff
Bob Damper
Kerstin Dautenhahn
Manuel de Pinedo
Ezequiel Di Paolo
Marco Dorigo
Alan Dorin
Arantza Etxeberria
Chrisantha Fernando
Sevan G. Ficici
Dario Floreano
David Fogel
Daniel W. Franks
Nicholas Geard
Steve Grand
Patrick Grim
Jennifer Hallinan
Inman Harvey
Paulien Hogeweg
Owen Holland
Gregory Hornby
Phil Husbands
Auke Ijspeert
Frederic Kaplan
Jozef Kelemen
Jan Kim
Lex Kraaijeveld
Daniel Ladley

Tom Lenaerts
Kristian Lindgren
Hod Lipson
James Marshall
Barry McMullin
J. J. Merelo
Julian F. Miller
Melanie Mitchell
Chrystopher L. Nehaniv
Ulrich Nehmzow
Stefano Nolfi
Charles Ofria
Pierre-Yves Oudeyer
Alexandra Penn
Rolf Pfeifer
Andrew Philippides
Daniel Polani
Jordan Pollack
William Rand
Luis Rocha
Jonathan Rowe
Graeme Ruxton
Fabrice Saffre
Hiroki Sayama
Anil Seth
Eric Silverman
Kenny Smith
Andrew Smith
Susan Stepney
Tim Taylor
Charles Taylor
Rene te Boekhorst
Peter Todd
Elio Tuci
Richard Vaughan
Paul Vogt
James Watson
Barbara Webb
Michael Wheeler
Janet Wiles
Kai Willadsen
Hywel Williams
Peter Wills
Simon Worgan
Klaus-Peter Zauner

We would also like to acknowledge the valuable input of several additional reviewers:

Christos Ampatzis
Prasanna Balaprakash
Alexandre Campo
Anders L. Christensen
Rob Mills
Martin Nilsson Jacobi
Christopher Strelioff
Colin Tosh

It is extremely important to recognise the support offered to us by our sponsors: the School of Electronics and Computer Science at the University of Southampton, British Telecom, ProtoLife, Icosystem, EPSRC, the Southampton Life Sciences Interface Forum, the ESIGNET and SECSE projects, and the International Society for Artificial Life, as well as one anonymous donor. With their help we have been able to run a very affordable conference, and to offer significant financial assistance to over 30 delegates.

We were very lucky to secure the services of a truly excellent array of world-class keynote speakers: Andrew Ellington, Takashi Ikegami, Eva Jablonka, Stuart Kauffman, and Peter Schuster. Our thanks to them for taking the time and effort to contribute to the conference.

Andrew Ellington received his BS in Biochemistry from Michigan State University in 1981, and his PhD in Biochemistry and Molecular Biology from Harvard in 1988. As a graduate student he worked with Dr. Steve Benner on the evolutionary optimization of dehydrogenase isozymes. His post-doctoral work was with Dr. Jack Szostak at Massachusetts General Hospital, where his lab developed methods for the in vitro selection of functional nucleic acids and coined the term 'aptamer'. Dr. Ellington began his academic career as an assistant professor of Chemistry at Indiana University in 1992. In 1998 he moved to the University of Texas at Austin and is now the Fraser Professor of Biochemistry. Dr. Ellington's lab continues to develop functional nucleic acids for practical applications, including aptamer biosensors, allosteric ribozyme logic gates (aptazymes), and internalizing nucleic acids that can deliver siRNAs to cells. A next leap forward will hopefully be to develop synthetic genetic circuits that can perform amorphous computations. Ultimately, though, Dr. Ellington's first love remains origins of life research, which oddly melds with translational research initiatives in that it is the ultimate biotechnology challenge.

Takashi Ikegami is Associate Professor in the Department of General Systems Sciences, Graduate School of Arts and Sciences at the University of Tokyo. He is a long-standing member of the Artificial Life community, with work spanning a diverse array of concepts, such as chaotic

itinerancy, self-organisation, autopoiesis, and embodiment, applied to a range of behaviours including coevolution, learning, language, social behaviour and song, in systems of birds, robots, children, flies, cells, and even oil droplets. These interests are unified by a focus on understanding the fundamental behavioural dynamics of embedded, embodied, evolving and adaptive systems.

Eva Jablonka is a geneticist known especially for her work on epigenetic inheritance. Her research with Marion Lamb is in the vanguard of what has been described as an ongoing revolution within evolutionary biology. In their current book, they describe how the growing body of evidence for the evolutionary role of epigenetic processes is putting increasing pressure on the dominant neo-Darwinian paradigm. Jablonka is a professor at the Cohn Institute for the History of Philosophy of Science and Ideas at Tel Aviv University and was awarded the Landau Prize of Israel in 1981 and the Marcus prize in 1988.

Stuart Kauffman originally trained as a physician and is now a biologist and complex systems researcher. His primary work has been as a theoretical biologist studying the origin of life and molecular organization. His seminal models of autocatalytic sets, gene regulatory networks and fitness landscapes allowed him to develop an extremely influential account of the way in which self-organisation within biology can generate "order for free". He currently holds a chair spanning the departments of biological sciences and physics and astronomy at the University of Calgary where he is the director of the Institute for Biocomplexity and Informatics. A MacArthur Fellow and a Trotter Prize winner, his latest book *Reinventing the Sacred: A New View of Science, Reason, and Religion* was published this year.

Peter Schuster is a renowned biophysicist, known for his work with Manfred Eigen in developing the quasi-species model. His main research interests are bioinformatics and structure prediction of ribonucleic acids, the study of mechanisms of biological evolution by means of molecular models, the design of molecules for predefined purposes as well as the application of inverse methods in computational systems biology. He is full professor of theoretical chemistry at the University of Vienna. In 1992-1995 he was the founding director of the Institute of Molecular Biotechnology and head of its Department of Molecular Evolutionary Biology in Jena, Germany. He is member of the German Academy of Sciences Leopoldina, is the editor-in-chief of *Complexity* and is currently President of the Austrian Academy of Sciences. In 1995 Peter Schuster received the Phillips-Morris Award and in 1999 the Wilhelm-Exner Medal.

We would also like to recognise the invaluable assistance provided by the people who helped to make the conference happen: the postdocs, postgrads and MSc students in the SENSe group at ECS, Southampton, Denise Harvey, the group secretary, the ECS finance group, Joyce Lewis and Sarah Prendergast, for help with posters and other materials, C. Titus Brown for running the submissions website, MIT Press for agreeing to experiment with an entirely new model for delivering a conference proceedings, Hannah Lane and her team at the University of Winchester Conference Office who were outstanding in their support, efficiency and professionalism, and in particular, Nic Geard who has been central to the smooth running of the entire operation from start to finish.

By way of conclusion, it seems appropriate to remind the readers of this proceedings volume that, over the last two decades, some of the highly speculative ideas that were discussed at the field's inception have matured to the extent that whole new conferences and journals devoted to

them are being established: synthesising artificial cells, simulating massive biological networks, exploiting biological substrates for computation and control, and deploying bio-inspired engineering are all now cutting-edge practice. It is our intention that the ALIFE conference series continue to provide an opportunity for those working across these topics to get together and exchange ideas and results, showcasing the best current work in the field, highlighting new directions for investigation, and providing a platform for world-renowned keynote speakers. Our thanks to all who have attended and made this possible.

A note on the cover

We chose to promote the conference with an image that is in some sense itself an example of artificial life: a real organism artificially encouraged to adopt the shape of the host country (plus Ireland). The cover of this proceedings volume features a photograph of a single-cell creature, the slime mould *Physarum polycephalum*, that was grown over a period of between twelve and twenty-four hours in a petri dish. While we have tidied the image up a little, it is essentially undoctored. The slime mould was grown by Soichiro Tsuda, and photographed by Soichiro, Nic Geard and Seth Bullock. The initial idea was proposed by Richard Watson during a particularly creative lunch.

In order to achieve the shot, we used a piece of acetate with an appropriately shaped hole as a template, and grew the slime mould across this area. The network of microtubules that you can see forms spontaneously as the creature grows, and reflects the self-organised system of nutrient transport that the slime mould uses. Since *Physarum* does not enjoy acetate as a habitat, it is relatively easy to remove the template and leave behind the organism, which has adapted to the niche it was offered by creating a living map of the the United Kingdom (and Ireland).

The organising committee of Artificial Life XI,

Seth Bullock (Conference Chair)
Jason Noble (Program Chair)
Richard Watson (Proceedings Chair)
Mark Bedau

Southampton, June, 2008

Contents

Presented papers

Adaptive spam detection inspired by the immune system 1
Abi-Haidar and Rocha

Uncertainty and communication complexity in iterated cooperation games 9
Andras

Investigating patterns for the process-oriented modelling and simulation of space in complex systems 17
Andrews, Sampson, Bjørndalen, Stepney, Timmis, Warren and Welch

On preferred states of agents: how global structure is reflected in local structure 25
Anthony, Polani and Nehaniv

Information-theoretic aspects of control in a bio-hybrid robot device 33
Artmann, Tsuda and Zauner

Movement strategies for learning in visual recognition 41
Bermudez, Philippides and Seth

The emergence of specialization 49
Bersini

Autocatalytic replication of polymers revisited 56
Blundell

Behavior chaining: incremental behavioral integration for evolutionary robotics 64
Bongard

Evolving CSR strategies in virtual plant communities 72
Bornhofen and Lattaud

Building artificial personalities: expressive communication channels based on an interlingua for a human-robot dance 80
Bryden, Hogg, Popat and Wallis

Basic principles of adaptive learning through variation and selection 88
Burtsev

Design strategies for open-ended evolution 94
Cariani

On the ability of swarms to compute the 3-coloring of graphs 102
Cases, Hernandez, Graña and d'Anjou

The effects of periodic and continuous market environments on the performance of trading agents 110
Chaggar, Noble and Cliff

Measuring the robustness of a developmental system based on sequential growth rules 118
Cohen, Miodownik and Baum

"Soft" continuum robots: the interaction of continuous and discrete elements 126
Cowan and Walker

From single cell to simple creature morphology and metabolism 134
Cussat-Blanc, Luga and Duthen

A model chemical memory in an evolved animat 142
Dale

Protein folding with stochastic L-systems 150
Danks, Stepney and Caves

Unexpected evolutionary dynamics in a string based artificial chemistry 158
Decraene, Mitchell and McMullin

Predictive information and emergent cooperativity in a chain of mobile robots 166
Der, Güttler and Ay

Artificial-life ecosystems: what are they and what could they become? 173
Dorin, Korb and Grimm

Programmable architectures that are complex and self-organized: from morphogenesis to engineering 181
Doursat

Entropy production in an energy balance Daisyworld model 189
Dyke

Group formation and social evolution: a computational model 197
Geard and Bullock

A computational model of gene regulatory networks and its topological properties 204
Gonçalves and Costa

What makes the spatial prisoner's dilemma game sensitive to asynchronism? 212
Grilo and Correia

Being arranged in advance: quantum entanglement and biological feedback 220
Haruna

Misrepresentations 227
Harvey

NKα: non-uniform epistatic interations in an extended NK model 234
Hebbron, Bullock and Cliff

Why should you care? An arousal-based model of exploratory behavior for autonomous robot 242
Hiolle and Cañamero

The effects of payoff preferences on agent tolerance 249
Howley and O'Riordan

Analysis of a dynamical recurrent neural network evolved for two qualitatively different tasks: walking and chemotaxis 257
Izquierdo and Buhrmann

The evolution of evolvability in gene transcription networks 265
Izquierdo and Fernando

Energy, entropy and work in computational ecosystems: a thermodynamic account 274
Jacyno and Bullock

Evolving asynchronous cellular automata for density classification 282
Jeanson

Effects of signalling on the evolution of gene regulatory networks 289
Jenkins and Stekel

Evo-devo in silico: a model of a gene network regulating multicellular development in 3D space with artificial physics 297
Joachimczak and Wróbel

Evolving functional symmetry in a three dimensional model of an elongated organism 305
Jones, Jin, Sendhoff and Yao

Enrichment of interaction rules in a string-based artificial chemistry 313
Kelly, McMullin and O'Brien

Evolution and morphogenesis of differentiated multicellular organisms: autonomously generated diffusion gradients for positional information 321
Knabe, Schilstra and Nehaniv

Exploiting open-endedness to solve problems through the search for novelty 329
Lehman and Stanley

Adaptive multi-robot bucket brigade foraging 337
Lein and Vaughan

Tracking the evolution of chemical computing networks 343
Lenser, Matsumaru, Hinze and Dittrich

Simulated evolution of mass conserving reaction networks 351
Liekens, ten Eikelder, Steijaert and Hilbers

Distributed gradient optimization with embodied approximation 359
Litus and Vaughan

Emergence of glider-like structures in a modular robotic system 366
Lizier, Prokopenko, Tanev and Zomaya

The information dynamics of phase transitions in random Boolean networks 374
Lizier, Prokopenko and Zomaya

'Psychoanalysis' of a minimal agent 382
Manicka and Harvey

Conceptual structure in cellular automata: the density classification task 390
Marques-Pita and Rocha

The longevity of distinct cultures in an agent-based model of memetic drift 398
Matthews

Fitness transmission: a genealogic signature of adaptive evolution 404
Miconi

Anticipating future experience using grounded sensorimotor informational relationships 412
Mirza, Nehaniv, Dautenhahn and te Boekhorst

Common concepts in agent groups, symmetries, and conformity in a simple environment 420
Möller and Polani

KohonAnts: a self-organizing ant algorithm for clustering and pattern classification 428
Mora, Fernandes, Merelo, Ramos, Laredo and Rosa

Emergence of cooperation in N-player games on small world networks 436
O'Riordan, Cunningham and Sorensen

Optimal noise in spiking neural networks for the detection of chemicals by simulated agents 443
Oros, Steuber, Davey, Cañamero and Adams

Holey fitness landscapes and the maintenance of evolutionary diversity 450
Paperin, Sadedin, Green and Dorin

Conformist transmission and the evolution of cooperation 458
Peña

Hawks and doves in an artificial dynamically structured society 466
Pestelacci and Tomassini

Evolving morphological and behavioral diversity without predefined behavior primitives 474
Pichler and Cañamero

Complex systems models: engineering simulations 482
Polack, Hoverd, Sampson, Stepney and Timmis

Modelling stigmergic gene transfer 490
Polani, Prokopenko and Chadwick

The efficacy of group selection is increased by coexistence dynamics within groups 498
Powers, Penn and Watson

Mechanism as mind: what tensegrities and caterpillars can teach us about soft robotics 506
Rieffel, Trimmer and Lipson

"Embryo": an autonomic co-operative service management framework 513
Saffre and Shackleton

The limited utility of communication in simple organisms 521
Scheutz and Schermerhorn

Analysing honeybees' division of labour in broodcare by a multi-agent model 529
Schmickl and Crailsheim

Altruism amongst spatial predator-prey animats 537
Scogings and Hawick

Measuring emergence via nonlinear Granger causality 545
Seth

Modeling the neural basis of cognitive integration and consciousness 553
Shanahan and Connor

Evolving an agent-based model to probe behavioral rules in flocks of cowbirds 561
Smith

Evolutionary advantages of neuromodulated plasticity in dynamic, reward-based scenarios 569
Soltoggio, Bullinaria, Mattiussi, Dürr and Floreano

Can body language shape body image? 577
Steels and Spranger

Multiple functionalities of biochemical reaction networks 585
Steijaert, Liekens, ten Eikelder and Hilbers

An artificial chemistry-based model of economies 592
Straatman, White and Banzhaf

Homeodynamics in the game of life 600
Suzuki and Ikegami

How learning can guide evolution of communication 608
Suzuki and Arita

Self-assembly in physical autonomous robots: the evolutionary robotics approach 616
Tuci, Ampatzis, Trianni, Christensen and Dorigo

Evolution, development and environment toward adaptation through phenotypic plasticity and exploitation of external information 624
Tufte

Simulated trust: towards robust social learning 632
Vanderelst, Ahn and Barakova

A study of GasNet spatial embedding in a delayed-response task 640
Vargas, Di Paolo and Husbands

The connectivity of NK landscapes' basins: a network analysis 648
Verel, Ochoa and Tomassini

Adaptive growth processes: a model inspired by Pask's ear 656
Virgo and Harvey

The spatiality of swarms — quantitative analysis of dynamic interaction networks 662
von Mammen and Jacob

Optimal robot recharging strategies for time discounted labour 670
Wawerla and Vaughan

Hierarchical components and entity-based modelling in artificial life 678
Webster and Malcolm

Investigating emergence by coarse graining elementary cellular automata 686
Weeks, Polack and Stepney

Understanding robustness in random Boolean networks 694
Willadsen, Triesch and Wiles

Evolving referential communication in embodied dynamical agents 702
Williams, Beer and Gasser

Species selection of aging for the sake of diversity 710
Woodberry, Korb and Nicholson

Initial modelling of the alternative phenotypes hypothesis 717
Worgan and Mills

Passive and driven trends in the evolution of complexity 725
Yaeger, Griffith and Sporns

Fitness based identification of a robot structure 733
Zagal, Ruiz-del-Solar and Palacios

Presented abstracts

Topological properties of evolved robot brains 742
Adami, Chaumont, Edlund and Hintze

Evolutionary robotics and the morphological turn: an epistemological perspective 743
Almeida e Costa

Morphodynamics and perceptual worlds: conceptual approach and an experiment in evolutionary robotics 744
Almeida e Costa, Macinnes and Harvey

Smart plankton: a new generation of underwater wireless sensor network 745
Anguita, Brizzolara, Ghio and Parodi

Combinatorial optimization and self-disciplined computing by amoeba-based neurocomputer 746
Aono and Hara

Artificial mental life 747
Barandiaran and Di Paolo

Ruggedness and evolvability — an evolution's-eye view 748
Barnett

A functional account of minimal cellular life 749
Bedau

The arrow of complexity hypothesis 750
Bedau

Artificial life as cancer research: embodied agent modelling of blood vessel growth in tumours 751
Bentley, Bates and Gerhardt

Modelling spatial market dynamics with retail agents 752
Birkin and Heppenstall

Does coexistence solve the prebiotic information problem? 753
Branciamore, de Back and Gallori

How might group selection explain the major evolutionary transitions? 754
Bryden

Dynamical complexity of spatially embedded networks 755
Buckley, Barnett and Bullock

Learning drives the accumulation of adaptive complexity in simulated evolution 756
Burtsev, Anokhin and Bateson

Dynamically adapting parasite virulence to combat coevolutionary disengagement 757
Cartlidge

A measure for natural selection's contribution to the origins and maintenance of organismal complexity 758
Channon

An adaptive model of marine biogeochemistry in the Archaean 759
Clark and Williams

Phenotype-based evolution of complex food webs 760
de Back, Branciamore and Kampis

Life in time: the missing temporal dimension in autopoiesis 761
Di Paolo

Chemical organizations in living systems 762
Dittrich, Centler, Kaleta, Matsumaru and Speroni di Fenizio

Strategies for maintaining large robot communities 763
English, Gough, Johnson, Spanton, Sun, Crowder and Zauner

Abstract organization and material parts in ALife and synthetic biology 764
Etxeberria

Developing a methodology for social network sampling 765
Franks, James, Noble and Ruxton

Utility and experimental testability of the Gaia hypothesis 766
Free, McNeil, Mozley and Allen

CAT: a market design competition 767
Gerding, McBurney, Niu and Parsons

The gene-function relationship in the metabolism of yeast and digital organisms 768
Gerlee, Lundh, Zhang and Anderson

In silico evolution of chemotaxis 769
Goldstein and Soyer

Immune-inspired networked service delivery 770
Gowans, Marrow and Tateson

From market to non-market: an autonomous agent approach to central planning 771
Greenwood

How epigenetic evolution can guide genetic evolution 772
Groom, Mills and Watson

Chemical basis for minimal cognition 773
Hanczyc and Ikegami

Homeostasis via chaos: implementing the uniselector as a dynamical system 774
Harvey

Animatronic model of a human tongue 775
Hofe and Moore

Simulating active touch with a simple embodied agent 776
Iizuka and Ikegami

Adaptive fault tolerance in wireless sensor networks 777
Irons, Saffre and Cannings

Solving the division of labour problem using stigmergy and evolved heterogeneity 778
James, Noble and Watson

Nonconsumable resources facilitate complex evolution 779
Kampis, Gulyas and de Back

Intentional and causal urges: when to ascribe life or mind to artificial systems? 780
Keijzer

The effects of local information and trading opportunities in a network constrained economy 781
Ladley and Bullock

Towards mental life as it could be: a robot with imagination 782
Marques, Holland, Knight and Newcombe

Dimensions of adaptivity 783
McGann

An artificial chemistry towards the identification of the transition to life 784
Mendoza, Muethe, Gómez and Ewert-Sarmiento

Adaptive units of selection can evolve complexes that are provably unevolvable under fixed units of selection 785
Mills and Watson

Considering the reconfiguration process of subjective temporal order by using recurrent neural networks 786
Nakajima and Ikegami

Emergent intracellular network states and cell fate decision: a dynamically integrated study of the epidermal growth factor receptor network with an agent based model 787
Nene and Nagl

Is the organism really a machine? 788
Nicholson

Intelligent locomotion of eukaryotic cells 789
Nishimura and Sasai

What can artificial life offer ecology? 790
Noble, Clarke and Mills

Information-theoretic characterization of relative and fluctuating system–environment distinction 791
Nozawa and Kondo

Personalities in fish without genetic differences: a model 792
Oosten and Hemelrijk

What can artificial life offer the development of methodologies in the field of socio-ecological sustainability? 793
Penn

Mechanisms for the initiation of multicellularity in bacterial biofilms 794
Penn, Watson, Powers, Webb, Kraaijeveld, Conibear and Bigg

Quantifying creative symbiosis: a lexical analysis of the evolution of technology as reflected in patent records 795
Pepper, Francis, Bedau, Chalmers and Head

The group selection debate and ALife: weak altruism, strong altruism, and inclusive fitness 796
Powers and Watson

A strong local dependency of developmental bias orients adaptive evolution in a model evo-devo system 797
Psujek and Beer

Evolutionary robotics models in the interdisciplinary study of embodied time perception 798
Rohde and Di Paolo

Selective attention in artificial organisms 799
Ruini and Parisi

Organizational requirements for 'open-ended evolution' 800
Ruiz-Mirazo, Umerez and Moreno

Self-organizing heterogeneous swarms designed through evolutionary methods 801
Sayama

Redefining robustness 802
Silverman and Ikegami

Dancing robots: form, environment and context in human-robot interaction 803
Silverman, Ikegami, Ogai and Aucouturier

The virtual rocky shore — linking A-life with ecological and pedagogical research 804
Stafford, Davies and Williams

A minimal approach to modular assembly 805
Studer and Harvey

Evolution of complexity in RNA-like replicator systems 806
Takeuchi and Hogeweg

How to know without having been there? — Investigating communication channels in the nectar collecting system of a honeybee colony 807
Thenius, Schmickl and Crailsheim

Towards language acquisition in autonomous robots 808
Tikhanoff, Cangelosi, Tani and Metta

Complex consumer preferences from rudiments of visual processing 809
Tosh, Krause and Ruxton

Self-organising synchronisation in a robotic swarm 810
Trianni and Nolfi

Mammalian decisions 811
Trimmer, McNamara, Houston, Marshall, Paul, Bogacz and Mendl

Modelling the social coordination of behavior with public symbols 812
Türkmen and Zugic

Using the RNA sequence-to-structure map for functional evolution of ribozyme catalyzed artificial metabolisms 813
Ullrich, Flamm and Endler

Modelling resilience of agro-ecosystems: the co-evolution of regimes 814
van Apeldoorn, Kok, Sonneveld and Veldkamp

Diet learning and the evolution of cultural inheritance 815
van der Post, Ursem and Hogeweg

Reaction-diffusion spots as a model for autopoiesis 816
Virgo and Harvey

Can we rebuild a cell? Bryopsis — an experimental model! 817
Vladimirescu

Can individual selection favour significant higher-level selection? 818
Watson, Mills, Penn and Powers

Autopoiesis, enactivism, and the extended mind 819
Wheeler

Environmental regulation by higher level selection in a simulated network of microbial ecosystems 820
Williams and Lenton

Timing of critical periods in development 821
Winks and Berthouze

Time and space in neuronal networks: the effects of spatial organisation on network behaviour 822
Womble and Cohen

Canalization and environmental engineering 823
Yamauchi

Adaptive Spam Detection Inspired by the Immune System

Alaa Abi-Haidar* and Luis M. Rocha

Department of Informatics, Indiana University, Bloomington IN 47401, USA
and
Instituto Gulbenkian de Ciência, Oeiras, Portugal
*aabihaid@indiana.edu

Abstract

This paper proposes a novel solution to spam detection inspired by a model of the adaptive immune system known as the cross-regulation model. We report on the testing of a preliminary algorithm on six e-mail corpora. We also compare our results with those obtained by the Naive Bayes classifier and another binary classification method we developed previously for biomedical text-mining applications. We obtained very encouraging results which can be further improved with development of this bio-inspired model. We show that the cross-regulation model is promising as a bio-inspired algorithm for spam detection in particular, and binary classification in general. Finally, we also present evidence that our bio-inspired model is relevant for understanding immune regulation itself.

Introduction

Spam detection is a binary classification problem in which e-mail is classified as either ham (legitimate e-mail) or spam (illegitimate or fraudulent e-mail). Spam is very dynamic in terms of advertising new products and finding new ways to defeat anti-spam filters. The challenge in spam detection is to find the appropriate threshold between ham and spam leading to the smallest number of misclassifications, especially of legitimate e-mail (false negatives). To avoid confusions, ham and spam will be labeled as positives and negatives respectively.

The vertebrate adaptive immune system, which is one of the most complex and intelligent biological systems, learns to distinguish harmless from harmful substances (known as pathogens) such as viruses and bacteria that intrude the body. These pathogens often evolve new mechanisms to attack the body and its immune system, which in turn adapts and evolves to deal with changes in the repertoire of pathogen attacks. A weakly responsive immune system is vulnerable to attacks while an aggressive one can be harmful to the organism itself, causing autoimmunity. Given the conceptual similarity between the problems of spam and immunity, we investigate the applicability of the cross-regulation model of T-cell dynamics (Carneiro et al., 2007) to spam detection.

Below we offer a short review of related work in spam detection, a brief introduction to the adaptive immune system, and the cross-regulation model (Carneiro et al., 2007). In the following section, the bio-inspired cross-regulation algorithm and its application to spam are discussed. In the Results section, the experiments and implementation of the model vis a vis the other binary classification models are discussed.

Spam Detection

Spam detection has recently become an important problem with the ubiquity of e-mail and the rewards of no-cost advertisement that can reach the largest audience possible. Spam detection can target e-mail headers (e.g. sender, receiver, relay servers...) and/or content (e.g. subject, body). Machine learning techniques such as support vector machines (Carreras and Marquez, 2001; Kolcz and Alspector, 2001), Naive Bayes classifiers (Sahami et al., 1998; Metsis et al., 2006) and other classification rules such as Case-Based Reasoning (Fdez-Riverola et al., 2007) have been very successful in detecting spam in the past. However, they generally lack the ability to detect spam drift since they rely on training on fixed corpora, features and rules. Research in this area is now focusing on concept drift in spam, with very promising results (Delany et al., 2006a; Méndez et al., 2006; Tsymbal, 2004; Kolter and Maloof, 2003). In addition, social-based spam detection models (Boykin and Roychowdhury, 2005; Chirita et al., 2005) have recently become relevant and competitive. Artificial Immune System (AIS) based algorithms (Oda, 2005; Bezerra and Barra, 2006; Yue et al., 2007) are another area of exciting development. The AIS models are inspired by diverse responses and theories of the natural immune system (Hofmeyr, 2001) such as negative selection, clonal selection, danger theory and the immune network theory. Our bio-inspired spam detection algorithm is based instead on the cross-regulation model (Carneiro et al., 2007), which is a novel development in AIS approaches to spam detection.

The Adaptive Immune System

The immune system, and more specifically, the vertebrate adaptive immune system, is a complex network of cells that distinguish between harmless and harmful substances or antigens—usually proteins or fragments of proteins and certain types of carbohydrate polymers that can be recognized by the immune system. When harmful antigens are discovered, an immune response to eliminate them is set in motion. Recognizing harmless self antigens, which obviously should not lead to an immune response to eliminate them, is resolved by a process known as positive and negative selection of T-cells which takes place in the thymus. It is in the thymus that T-cells develop and mature; only T-cells that have failed to bind to self antigens are released, while the rest of the T-cells is culled. The mature T-cells are allowed out of the thymus to detect harmful nonself antigens. They do this by binding to antigen presenting cells (typically B-cells, macrophages and dendritic cells) that collect and present antigens through MHC complexes after breaking them by lysosome. The specific T-cells that are able to bind to the presented antigens then stimulate B-cells that start a cascade of events leading to antibody production and the destruction of the pathogens or tumors linked to the antigens. However, it is possible that T-cells and B-cells, which are also trained in the thymus, could mature before being exposed to all self antigens. Even more problematic is the somatic hypermutation that ensues in lymph nodes after the activation of B-cells. At this stage, it is possible to generate many mutated B-Cell clones that could bind to harmless self antigens. Either situation can cause auto-immunity by generating T-cells capable of attacking self antigens. One way around this is by a process called costimulation which involves the co-verification of self antigens by both T-cells and B-cells before the antigen is identified as harmful pathogen and attacked. To further insure that the T-cells do not attack self, another type of T-cells known as T regulatory cells, are formed in the thymus where they mature to avoid recognizing self antigens. These regulatory T-cells have the responsibility of preventing autoimmunity by suppressing other T-cells that might bind and kill self antigens.

The Cross-regulation Model

The cross-regulation model, proposed by Carneiro et al. (2007), aims to model the process of discriminating between harmless and harmful antigen—typically harmless self/nonself and harmful nonself. The model consists of only three cell types: Effector T-Cells (E), Regulatory T-Cells (R) and Antigen Presenting Cells (A) whose populations interact dynamically, ultimately to detect harmful antigens. E and R are constantly produced, while A are capable of presenting a collection of antigens to the E and R. T-cell proliferation depends on the co-localization of E and R as they form conjugates (bind) with the antigens presented by A cells (this model assumes that A can form conjugates with a maximum of two E or R). The population dynamics rules of this model are defined by three differential equations, which can be, for every antigen being presented by an A, summarized by the following three laws of interaction:

1. When E bind to the A, they proliferate with a fixed rate.
2. When R bind to the A, they remain in the population.
3. if an R binds together with an E to the same A, the R proliferates with a certain rate and the E remains in the population but does not proliferate.

The E and R proliferation rates in this model are fixed to 200%, which is the exactly the process of duplication or production of one extra copy. Finally, the E and R die at a fixed death rate. Carneiro et al. (2007) showed that the dynamics of this system leads to a bistable system of two possible stable population concentration attractors: (i) the co-existence of both E and R types identifying harmless self antigens, or (ii) the progressive disappearance of R, identifying harmful antigens. An illustration of the three rules is shown in figure 1 and more details on the model are available in the original paper (Carneiro et al., 2007).

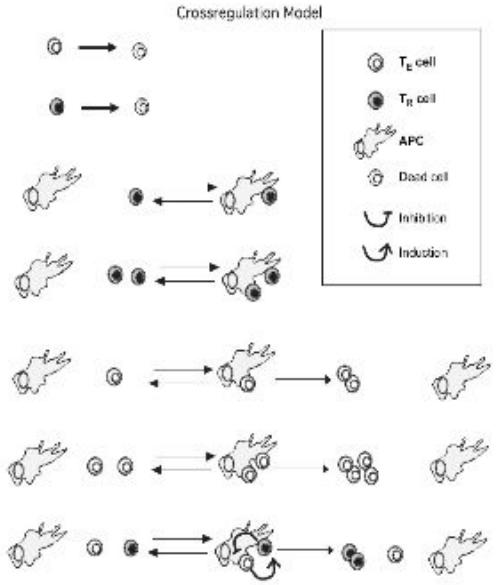


Figure 1: Figure is courtesy of Carneiro et al. (2007). The Cross-regulation Model. The diagram illustrates the interactions underlying the dynamics of A, E and R as assumed in the model in which A can only form conjugates with a maximum of two T cells.

The Cross-regulation Spam Algorithm

In order to adopt the cross-regulation algorithm for spam detection, which we named the Immune Cross-Regulation Model (ICRM), one has to think of e-mails as analogous to

the organic substances that upon entering the body are broken into constituent pieces by lysosome in A. In biology, these pieces are antigens (typically protein fragments) and in our analogous algorithm they are words extracted from e-mail messages and processed to become features[1]. Thus, in this model, antigens are words or potentially other features. For every antigen there exists a number of virtual E and R that interact with A which present a sample of the features of a given e-mail message. In other words, the A correspond to the e-mail. The general ICRM algorithm is designed to be first trained on N e-mails of "self" (a user's outbox) and harmless "nonself" (a user's inbox). However, in the results described here, it was not possible to directly obtain outbox data; we are currently working on collecting outbox data for future work. In addition, the ICRM is also trained on "harmful nonself" (spam arriving to a given user). Training on or exposure to ham e-mails, in analogy with Carneiro's et al model (Carneiro et al., 2007), is supposed to lead to a "healthy" dynamics denoted by the co-existence of both E and R with more of the latter. In contrast, training on or exposure to spam e-mails is supposed to result in much higher numbers of E than R. When e-mail features occur for the first time, a fixed initial number of E and R, for every feature, are generated. These initial values of E and R are different in the training and testing stages; more weight to R for ham features, and more weight to E for spam features is given in the labeled training stage. While we specify different values for initializing the proportions of E and R associated with e-mail features, depending on whether the algorithm is in the training or the testing stage, the ICRM is based on the exact same algorithm in both stages. An illustration comparing the artificial model to the biological one is shown in figure 2. The ICRM algorithm begins when an e-mail is received and cycles through three phases for every received e-mail:

In the **pre-processing phase**, HTML tags are not stripped off and are treated as other words, as often done in spam-detection (Metsis et al., 2006) . All words constituting the e-mail subject and body are lowercased and stemmed using Porter's algorithm (Porter, 1980) after filtering out common English stop words and words of length less than 3 characters. A maximum of n processed unique features (words, in this case) are randomly sampled and presented by the virtual A which corresponds to the e-mail. These virtual antigen presenting cells have n_A binding slots per feature, i.e. $n \times n_A$ slots per e-mail message. The breaking up of the e-mail message into constituent portions (features) is inspired by the natural process in Biology, but is further enhanced in this model to select the first and last $\frac{n}{2}$ features in the e-mail. The assumption is that the most indicative information is in the beginning (e.g. subject) and the end of the e-mail (e.g. signature), especially concerning ham e-mails. Nevertheless, the feature selection problem will be studied in more detail in future work.

[1]Naturally, features other than words are possible (e.g. bigrams, e-mail titles)

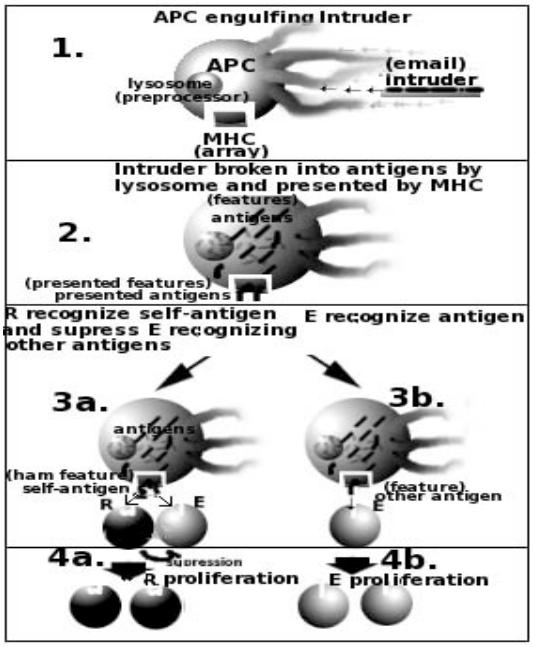


Figure 2: An illustration of the cross-regulation model (and its mapping to spam detection). In step 1, the intruder (received e-mail) is engulfed by an A (e-mail representer array). In step 2, the intruder is broken down by lysosome (preprocessor which strips html tags, filters out stop words and short words and porter stems a selection words) into antigens (features) which are then sampled and presented through MHC (an array residing in the memory) so that in step 3 specific E or R T-cells (virtual E and R residing in memory) can recognize it and bind to it. In step 3a, an R recognizing what probably is a self-antigen (ham feature) shares the A with an E recognizing a probably nonself-antigen (new or spam feature). In step 4a, the R suppresses the E which then excites the R to make it proliferate with a higher rate giving the antigen recognized by E more tolerance (making the novel feature more ham since it co-occurred with a ham feature). In step 3b on the other hand, the E is not suppressed by any R and thus it proliferates in step 4b making the system more immune to the antigen recognized by E (making the feature E recognize one more spam feature). After step 4, the whole intruder (e-mail) is judged based on its antigens (features) on whether it is bad or good (spam or ham) as explained in the decision phase of the algorithm.

In the **interaction phase**, feature-specific R_g and E_f are allowed to bind to the corresponding antigens presented by A, which are arbitrarily located on its array of feature slots. Every adjacent pair of A slots is dealt with separately: the E_f for a given feature f proliferate only if they do not find themselves sharing the same adjacent pair of A binding slots with R_g, in which case only the R_g, associated with feature g, proliferate. The model assumes that novel ham features k tend to have their E_k suppressed by R_g of other pre-occurring ham features g because they tend to co-occur in the same message. As for the algo-

rithm's parameters, let n_A be the number of A slots per feature. Let ($E_{0_{ham}}$, $R_{0_{ham}}$) and ($E_{0_{spam}}$, $R_{0_{spam}}$) be the initial values of E and R for features occurring for the first time in the training stage for spam and ham respectively. For the testing stage we have ($E_{0_{test}}$, $R_{0_{test}}$). Moreover, $E_{0_{ham}} << R_{0_{ham}}$, $E_{0_{spam}} > R_{0_{spam}}$ and $E_{0_{test}} > R_{0_{test}}$. Therefore, a feature f initially occurring in a ham e-mail would have $R_f >> E_f$ and vice versa for spam. In the ICRM implementation hereby presented, a major difference form Carneiro's et al model (Carneiro et al., 2007) was tried: the elimination of cell death. This is a rough attempt to provide the system with long term memory. Cell death can lead to the forgetfulness of spam or ham features if these features do not reoccur in a certain period of time as shown later on.

In the **decision phase**, the arriving e-mail is assessed based on the relative proportions of R and E for its n sampled features. Features with more R are assumed to correspond to ham while features with more E are more likely to correspond to spam. The proportions are then normalized to avoid decisions based on a few highly frequent features that could occur in both ham and spam classes. For every feature f, the feature score is computed as follows:

$$score_f = \frac{R_f - E_f}{\sqrt{R_f^2 + E_f^2}}, \quad (1)$$

indicating an unhealthy (spam) feature when $score_f \leq 0$ and a healthy (ham) one otherwise. $score_f$ varies between -1 and 1. For every e-mail message e, the e-mail immunity score is simply:

$$score_e = \sum_{\forall f \in e} score_f. \quad (2)$$

Note that a spam e-mail with no text such as as the cases of messages containing exclusively image and pdf files, which surpass many spam filters, would be classified as spam in this scheme—e-mail e is considered spam if $score_e = 0$. Similarly, e-mails with only a few features occurring for the first time, would share the same destiny, since the initial E is greater than R in the testing stage $E_{0_{test}} > R_{0_{test}}$ which would result in $score_e < 0$.

Results

E-mail Data

Given the assumption that personal e-mails (i.e. e-mails sent or received by one specific user) are more representative of a writing style, signature and themes, it would be preferable to test the ICRM on e-mails from a personal mailbox. Unfortunately, this is not offered by the most common spam corpus of *spamassasin*[2] and similarly for *ling-spam*[3]. In addition, the ICRM algorithm requires timestamped e-mails, since order of arrival affects final E/R populations. Timestamped data is also important for analyzing concept drifts over time, thus we cannot use the *PU1*[4] data described by Androutsopoulos et al. (2000b) . Delany's spam drift dataset[5], introduced by Delany et al. (2005), meets the requirements in terms of timestamped and personal ham and spam however its features are hashed and therefore it is not easy to make tangible conclusions based on their semantics. The *enron-spam*[6] preprocessed data perfectly meets the requirements as it has six personal mailboxes made public after the enron scandal. The ham mailboxes belong to the employees *farmer-d, kaminski-v, kitchen-l, williams-w3, beck-s and lokay-m*. Combinations of five spam datasets were added to the ham data from *spamassassin* (s), *HoneyProject* (h), *Bruce Guenter* (b) and *Georgios Paliousras'* (g) spam corpora and then all six datasets were tokenized (Metsis et al., 2006). In practice, some spam e-mails are personalized, which unfortunately cannot be captured in this dataset since the spam data comes from different sources. Only the first 1000 ham and 1000 spam e-mails of each of the corpora are used, as shown in table 1.

Table 1: Enron datasets

Dataset	ham + spam	ham:spam	[ham, spam] time range
Enron1	farmer-d + gp	1000:1000	[12/99, 06/00], [12/03, 01/05]
Enron2	kaminski-v + sh	1000:1000	[12/99, 05/00], [05/01, 07/05]
Enron3	kitchen-l + bg	1000:1000	[2/01, 06/01], [08/04, 03/05]
Enron4	williams-w3 + gp	1000:1000	[4/01, 01/02], [12/03, 06/04]
Enron5	beck-s + sh	1000:1000	[1/00, 11/00], [05/01, 03/05]
Enron6	lokay-m + bg	1000:1000	[6/00, 7/01], [08/04, 10/04]

ICRM Settings and Parameters

For each of the six enron sets, we ran each algorithm 10 times. Each run consisted of 200 training (50% spam) and 200 testing or validation (50% spam) e-mails that follow in timestamp order. From the 10 runs we computed variation statistics for the F-score[7], and Accuracy performance.

In the e-mail pre-processing phase, we used $n = 50$, $n_A = 10$, $E_{0_{ham}} = 6$, $R_{0_{ham}} = 12$, $E_{0_{spam}} = 6$, $R_{0_{spam}} = 5$, $E_{0_{test}} = 6$ and $R_{0_{test}} = 5$. These initial E

[2]http://spamassassin.apache.org/publiccorpus/

[3]http://www.aueb.gr/users/ion/publications.html

[4]http://www.iit.demokritos.gr/skel/i-config/downloads/enron-spam/

[5]http://www.comp.dit.ie/sjdelany/Dataset.htm

[6]http://www.iit.demokritos.gr/ ionandr/publications/

[7]The F1-measure (or *F-Score*) is defined as $F = \frac{2 \cdot Precision \cdot Recall}{Precision + Recall}$, where $Precision = \frac{TP}{(TP+FP)}$ and $Recall = \frac{TP}{(TP+FN)}$ and $Accuracy = \frac{(TP+TN)}{(TP+TN+FP+FN)}$ measures of the classification of each test set, where TP, TN, FP and FN denote true positives, true negatives, false positive and false negatives respectively (Feldman and Sanger, 2006)

and R populations for features occurring for the first time are chosen based on the initial ratios chosen by Carneiro et al. (2007) and were then empirically adjusted to achieve the best F-score and Accuracy results for the six enron datasets. Finally, the randomization seed was fixed in order to compare results to other algorithms and search for better parameters. The ICRM was compared with two other algorithms that are explained in the following two subsections. The ICRM was also tested on shuffled (not in order of date received) validation sets to study the importance of e-mail reception order. The results are shown in table 2. The mean and variance of the results are also plotted on the F-score vs Accuracy axes as shown in figure 3.

Naive Bayes

We have chosen to compare our results with the multinomial Naive Bayes (NB) with boolean attributes (Jensen et al., 1996) which has shown great success in a previous research (Metsis et al., 2006). In order to fairly compare NB with ICRM, we selected the first and last unique $n = 50$ features. The Naive Bayes classifies an e-mail as spam in the testing stage if it satisfies the following condition:

$$\frac{p(c_{spam}).\prod_{f\in e-mail} p(f|c_{spam})}{p(c_{spam}).\sum_{c\in\{c_{spam},c_{ham}\}}\prod_{f\in e-mail} p(f|c)} > 0.5, \quad (3)$$

where f is the feature sampled from an e-mail, and $p(f|c_{spam})$ and $p(f|c_{ham})$ are the probabilities that this feature f is sampled from a spam and ham e-mail respectively, while c is the union of spam and ham emails. The results are shown in table 2 and plotted in figure 3.

Variable Trigonometric Threshold (VTT)

We developed the VTT as a binary classification algorithm and implemented it as a protein-protein abstract classification tool[8] using bioliterature mining (Abi-Haidar et al., 2007, 2008). VTT is itself inspired by another case-based spam detection algorithm (Fdez-Riverola et al., 2007). Briefly, VTT's strategy is to make a selection of most significant preprocessed words ranked by a score $S(w) = |p_{ham}(w) - p_{spam}(w)|$ where $p_{ham}(w)$ and $p_{spam}(w)$ are the probabilities of a word w of occurring in the ham and spam training datasets which in our case are batches of 200 e-mails each. Naturally, a selection of 650 words would be fairly sufficient. The e-mails are then reduced to vectors of these 650 words. Then, the probabilities of co-occurring pairs of words (w_i, w_j) in these vectors are computed using $p_{ham}(w_i, w_j)$ and $p_{spam}(w_i, w_j)$. Then the trigonometric measures of the angle α, of this vector with the p_{ham} axis: $\cos(\alpha)$ is a measure of how strongly terms are exclusively associated with training ham e-mails, and similarly $sin(\alpha)$ with training spam ones. Then, for every e-mail e, we compute the sum of all pairs' measures to study the e-mail e's likelihood of being ham or positive $P(e)$ and spam or negative $N(e)$:

$$P(e) = \sum_{(w_i,w_j)\in e} \cos(\alpha(w_i, w_j)), \quad (4)$$

$$N(e) = \sum_{(w_i,w_j)\in e} \sin(\alpha(w_i, w_j)) \quad (5)$$

and finally the decision of whether an e-mail is ham or spam is made using the VTT equation:

$$\begin{cases} e \in ham, & \text{if } \frac{P(e)}{N(e)} \geq \lambda_0 + \frac{\beta - np(a)}{\beta} \\ e \in spam, & \text{otherwise} \end{cases} \quad (6)$$

where λ_0 is a constant threshold for deciding whether an e-mail is positive (spam) or negative (ham) obtained through exhaustive parameter search. For this experiment $\lambda_0 = 1.3$ produces the best results. Another parameter is β which was used in the abstract classification experiment to regulate $np(a)$ which counts the number of tagged protein in an abstract a but will be ignored in spam detection for the sake of simplicity. Therefore, equation 6 can be reduced to classify e as ham if $\frac{P(e)}{N(e)} \geq 1.3$ or as spam otherwise. The results are shown in table 2 and plotted in figure 3 then discussed in the discussion section.

Table 2: F-score and Accuracy mean +/- sdev of 10 runs for 50% spam enron data sets with the first two columns using ICRM (the first one applied on ordered e-mail, the second one on shuffled timestamps of testing data, and the last two using Naive Bayes and VTT.

		ICRM		Other Algorithms	
Dataset		**Ordered**	**Shuffled**	**Naive Bayes**	**VTT**
Enron1	F-score	0.9 ± 0.03	0.9 ± 0.03	0.89 ± 0.04	0.91 ± 0.04
	Accuracy	0.9 ± 0.03	0.9 ± 0.03	0.87 ± 0.05	0.9 ± 0.04
Enron2	F-score	0.86 ± 0.06	0.85 ± 0.06	0.92 ± 0.07	0.82 ± 0.23
	Accuracy	0.85 ± 0.06	0.83 ± 0.07	0.93 ± 0.05	0.86 ± 0.13
Enron3	F-score	0.88 ± 0.04	0.88 ± 0.04	0.93 ± 0.03	0.86 ± 0.08
	Accuracy	0.87 ± 0.05	0.87 ± 0.05	0.92 ± 0.04	0.85 ± 0.07
Enron4	F-score	0.92 ± 0.05	0.92 ± 0.04	0.92 ± 0.05	0.95 ± 0.03
	Accuracy	0.92 ± 0.05	0.92 ± 0.05	0.91 ± 0.06	0.95 ± 0.03
Enron5	F-score	0.92 ± 0.03	0.87 ± 0.06	0.94 ± 0.04	0.84 ± 0.13
	Accuracy	0.91 ± 0.03	0.87 ± 0.05	0.95 ± 0.03	0.87 ± 0.09
Enron6	F-score	0.89 ± 0.04	0.9 ± 0.04	0.91 ± 0.02	0.88 ± 0.05
	Accuracy	0.88 ± 0.05	0.89 ± 0.05	0.9 ± 0.03	0.87 ± 0.07
Total	**F-score**	0.9 ± 0.05	0.89 ± 0.05	0.92 ± 0.04	0.88 ± 0.12
	Accuracy	0.89 ± 0.05	0.88 ± 0.06	0.91 ± 0.05	0.88 ± 0.08

Discussion

As clearly shown in table 2 and figure 3, ICRM, NB and VTT are very competitive for most enron datasets, indeed the performance of ICRM is statistically indistinguishable from VTT (F-score and Accuracy p-values 0.15 and 0.63 for the paired t-test validating the null hypothesis of variation equivalence), though its slightly lower performance

[8]The Protein Interaction Abstract Relevance Evaluator (VTT) tool is available at http://casci.informatics.indiana.edu/VTT/

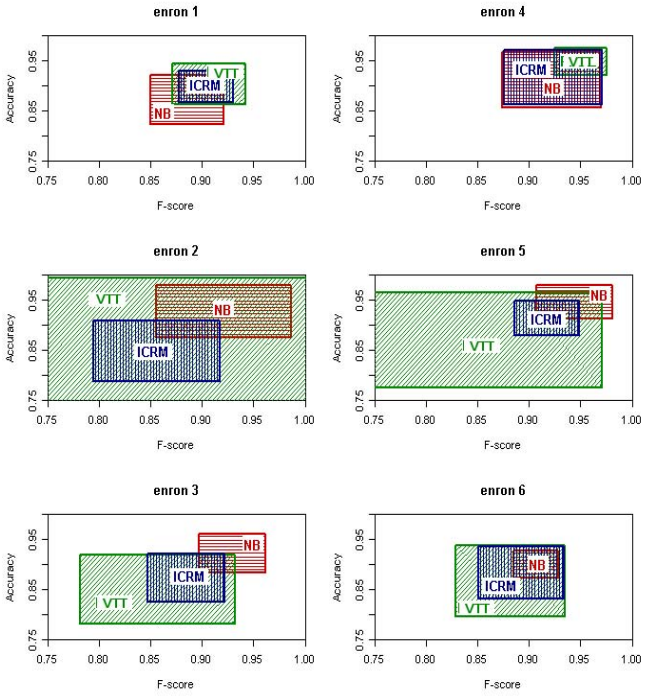


Figure 3: F-score vs Accuracy (mean and standard deviation) plot comparison between ICRM (vertical blue), NB (horizontal red) and VTT (diagonal green) for each of the six enron datasets. A visualization of table 2.

against NB is statistically significant (F-score and Accuracy p-values 0.01 and 0.02 for the paired t-test, rejecting the null hypothesis of variation equivalence with 0.05 level of significance).

More particularly, we investigate VTT's performance deviations between enron 2 and enron 4 and notice that the average number of top 650 features that are ham features is only 10.22 for enron 2 (having many spam and very few ham indicative features) while it is 75.02 for enron 4 (having relatively more ham and less spam indicative features) this giving us the maximum deviations off 43.40, which is the mean of ham features' constituency of the top 650 features for all enron sets. Enron 4's Inbox (williams-w3), contained 619 automatically generated notification e-mails of the exact same contents with a subtle variation in the filename id, as shown via *Enron Explorer*[9], an online visualization tool of the publicly available enron data. The peculiarity of enron 4 is also manifested in Metsis' Naive Bayes results (Metsis et al., 2006). We think that the huge proportion of spam indicative features for enron 2 (similarly but less so for enron 5) is due to the huge spam drift and diversity of spamassassin and HoneyProject spanning four years mostly in 2001, 2002 and 2005 which is not available in the barely six months lifespan of ham. This diversity gives VTT many highly indicative spam features that only occur in spam and much less, if at all, in ham. This leads to many ham misclassifications for the few indicative features (out of 650) that are selected for the training. A fix to this could be by either by increasing the threshold beyond 650 features or balancing the number of top 650 indicative ham and spam features as clearly is the case for enron 4, or by finding a synchronous spam and ham data. VTT's disadvantage of the features selection is paid off by its advantage of using feature co-occurrence of the top 650 features which is not the case in any of ICRM and NB. This might not be a fair comparison yet a modification to VTT would result in a modified VTT for another project and similarly, the use of co-occurrences with ICRM and NB will be pursued for a more advanced ICRM. From here onwards, we proceed with the comparison between ICRM and NB only.

[9]http://enron.trampolinesystems.com/focus/338815

Table 3: ICRM vs NB F-score and Accuracy mean +/- sdev for spam to ham ratio variations for mean of the six enron datasets.

		50% spam	30% spam	70% spam
ICRM	F-score	0.9 ± 0.05	0.91 ± 0.03	0.79 ± 0.12
	Accuracy	0.89 ± 0.05	0.86 ± 0.05	0.83 ± 0.08
NB	F-score	0.92 ± 0.04	0.86 ± 0.07	0.79 ± 0.07
	Accuracy	0.91 ± 0.05	0.84 ± 0.07	0.74 ± 0.01

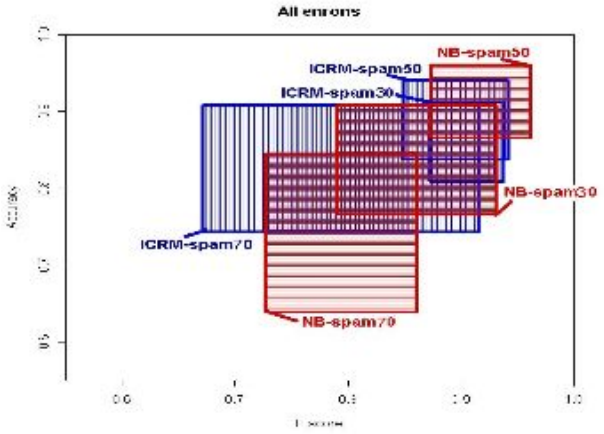


Figure 4: F-score vs Accuracy plot comparison between ICRM (vertical blue) and NB (horizontal red) with different spam to ham ratio variations 30:70 (spam30), 70:30 (spam70) and 50:50 (spam50) for the mean of the six enron datasets.

As shown in table 3 and figure 4, the ICRM can be more resilient to ham ratio variations[10]. While the performance of both algorithms was comparable for 50% spam (though significantly better for NB), the performance of NB drops for

[10]The 30% and 70% spam results were balanced for the evaluation by randomly sampling from the 70% class, reducing it to 30%

30% spam ratio (5% lower F-score than ICRM) and 70% spam ratio (9% less accurate than ICRM) while ICRM relatively maintains a good performance. The difference in performance is statistically significant, except for F-Score of the 70% spam experiment, as the p-values obtained for our performance measures clearly reject the null hypothesis of variation equivalence: F-Score and Accuracy p-values are 0 and 0.01 for 30% spam, and Accuracy p-value is 0.01 for 70% spam (p-value for F-Score is 0.5 for this case). While one could argue that NB's performance could well be increased, in the unbalanced spam/ham ratio experiments, by changing the right hand side of equation 3 to 0.3 or 0.7, this act would imply that, in real situations, one could know a priori the spam to ham ratio of a given user. The ICRM model, on the other hand, does not need to adjust any parameter for different spam ratios—it is automatically more reactive to whatever ratio it encounters. It has been shown that spam to ham ratios indeed vary widely Meyer and Whateley (2004); Delany et al. (2005), hence we conclude that the ICRM's ability to better handle unknown spam to ham ratio variations is more preferable for dynamic data classification in general and spam detection in particular.

In most Enron sets, the shuffled e-mails in the test set did very slightly worse than the ordered-by-reception-date ones. This observation was however statistically insignificant according to a t-test with p-value greater than 0.05 and thus it accepts null hypothesis of similarity between the two performances showing no importance of order for the ICRM dynamics. To further study the resilience of ICRM and its adaptive ability to catch concept drifts, we trained both ICRM and NB on the first 200 emails and then tested them on sequential overlapping slices of 200 emails. Our results showed very little decay in performance for both methods in most data sets (Abi-Haidar and Rocha, 2008). Therefore, we conclude that the data sets are not appropriate to study the effects of concept drift. In future work, we plan to test the ICRM on more appropriate data sets for the study of concept drift in spam (Delany et al., 2005, 2006b).

The three modifications to the original cross-regulation model, namely training on both ham and spam classes, feature selection and cell death elimination have quite improved the performance of the algorithm to make it rival with traditional binary classifier. The first modification's improvement was mostly manifested in enron 4 which cannot only rely on positive training for the majority of exact uninformative e-mails it has. Nonetheless, it is debatable whether the automatically generated messages in enron 4 should be classified as ham or not. The selection of the first and last features boosted the performance of both ICRM and NB about 2% in terms of F-score and Accuracy yet we are still working on making a better selection without totally disregarding the message body. The elimination of cell death also improved the overall performance of ICRM about 1%, especially in terms of long term memory. We are currently experimenting with a carrying capacity for the E and R concentrations that could be promising for future work.

Conclusion

The observations made based on the artificial immune system can help us guide or further deepen our understanding of the natural immune system. For instance, ICRM's resilience to spam to ham ratio show us how dynamic is our immune system and functional independently of the amount of pathogens attacking it. In addition, the three modifications made to the original model can be very insightful: The improvements made by training on both spam and ham (rather than only ham or self) reinforce the theories of both self and nonself antigen recognition by T-cells outside the thymus. The feature selection makes us wonder whether the actual T-cell to antigen binding is absolutely arbitrary. Finally, the elimination of cell death may reinforce the theories behind long lived cells as far as long term memory is concerned.

In this paper we have introduced a novel spam detection algorithm inspired by the cross-regulation model of the adaptive immune system. We have compared it with Naive Bayes and another binary classification tool called VTT. Our model has proved itself competitive with state of art spam binary classifiers in general and resilient to spam to ham ratio variations in particular through interestingly unique results that can be further improved by integration, hopefully in the near future. The overall results, even though not stellar, seem quite promising especially in the area of tracking concept drifts in spam detection. This original work should be regarded not only as a promising bio-inspired method that can be further developed and even integrated with other methods but also as a model that could help us better understand the behavior of the T-cell cross-regulation systems in particular, and the natural vertebrate immune system in general.

Acknowledgements

We thank Jorge Carneiro for his insights about applying ICRM on spam detection and his generous support and contribution for making this work possible. We also thank Florentino Fdez-Riverola for the very useful indications about spam datasets and work in the area of spam detection. We would also like to thank the FLAD Computational Biology Collaboratorium at the Gulbenkian Institute in Oeiras, Portugal, for hosting and providing facilities used to conduct part of this research.

References

Abi-Haidar, A., Kaur, J., Maguitman, A., Radivojac, P., Retchsteiner, A., Verspoor, K., Wang, Z., and Rocha, L. (2007). Uncovering protein-protein interactions in the bibliome. In *Proceedings of the Second BioCreative Challenge Evaluation Workshop*, volume ISBN 84-933255-6-2, pages 247–255.

Abi-Haidar, A., Kaur, J., Maguitman, A., Radivojac, P., Retchsteiner, A., Verspoor, K., Wang, Z., and Rocha, L. (2008). Uncovering protein-protein interactions in abstracts and text using linear models and word proximity networks. *Genome Biology*. inpress.

Abi-Haidar, A. and Rocha, L. (2008). Adaptive Spam Detection Inspired by a Cross-Regulation Model of Immune Dynamics. In *Proceedings of the 7th International Conference on Artificial Immune Systems (ICARIS 2008).* Lecture Notes on Computer Science, Springer-Verlag . inpress.

Androutsopoulos, I., Koutsias, J., Chandrinos, K., Paliouras, G., and Spyropoulos, C. (2000a). An evaluation of Naive Bayesian anti-spam filtering. *Proceedings of the workshop on Machine Learning in the New Information Age, 11th European Conference on Machine Learning, 2000*, pages 9–17.

Androutsopoulos, I., Koutsias, J., Chandrinos, K., and Spyropoulos, C. (2000b). *An experimental comparison of naive Bayesian and keyword-based anti-spam filtering with personal e-mail messages*. ACM Press New York, NY, USA.

Bezerra, G. and Barra, T. (2006). An Immunological Filter for Spam. *International Conference on Artificial Immune Systems (ICARIS 2006), LNCS*, pages 446–458.

Boykin, P. and Roychowdhury, V. (2005). Leveraging social networks to fight spam. *Computer*, 38(4):61–68.

Carneiro, J., Leon, K., Caramalho, Í., van den Dool, C., Gardner, R., Oliveira, V., Bergman, M., Sepúlveda, N., Paixão, T., Faro, J., et al. (2007). When three is not a crowd: a Cross-regulation Model of the dynamics and repertoire selection of regulatory CD4 T cells. *Immunological Reviews*, 216(1):48–68.

Carreras, X. and Marquez, L. (2001). Boosting Trees for Anti-Spam Email Filtering. *Proceedings of RANLP-2001, 2001* pages 58–64.

Chirita, P., Diederich, J., and Nejdl, W. (2005). MailRank: using ranking for spam detection. *Proceedings of the 14th ACM international conference on Information and knowledge management*, pages 373–380.

Delany, S. J., Cunningham, P., and Smyth, B. (2006a). Ecue: A spam filter that uses machine leaming to track concept drift. In Brewka, G., Coradeschi, S., Perini, A., and Traverso, P., editors, *ECAI 2006, 17th European Conference on Artificial Intelligence, August 29 - September 1, 2006, Riva del Garda, Italy, Including Prestigious Applications of Intelligent Systems (PAIS 2006), Proceedings*, pages 627–631. IOS Press.

Delany, S. J., Cunningham, P., and Tsymbal, A. (2006b). A comparison of ensemble and case-base maintenance techniques for handling concept drift in spam filtering. In Sutcliffe, G. and Goebel, R., editors, *Proceedings of the 19th International Conference on Artificial Intelligence (FLAIRS 2006)*, pages 340–345. AAAI Press.

Delany, S. J., Cunningham, P., Tsymbal, A., and Coyle, L. (2005). A case-based technique for tracking concept drift in spam filtering. *Knowledge-Based Systems*, 18(4–5):187–195.

Fdez-Riverola, F., Iglesias, E., Díaz, F., Méndez, J., and Corchado, J. (2007). SpamHunting: An instance-based reasoning system for spam labelling and filtering. *Decision Support Systems*, 43(3):722–736.

Feldman, R. and Sanger, J. (2006). *The Text Mining Handbook: advanced approaches in analyzing unstructured data*. Cambridge University Press.

Hofmeyr, S. (2001). An Interpretative Introduction to the Immune System. *Design Principles for the Immune System and Other Distributed Autonomous Systems*.

Jensen, F., Jensen, F., and Jensen, F. (1996). *Introduction to Bayesian Networks*. Springer-Verlag New York, Inc. Secaucus, NJ, USA.

Kolcz, A. and Alspector, J. (2001). SVM-based filtering of e-mail spam with content-specific misclassification costs. *Proceedings of the TextDM*, pages 1–14.

Kolter, J. and Maloof, M. (2003). Dynamic weighted majority: a new ensemble method for tracking concept drift. *ICDM 2003. Third IEEE International Conference on data mining 2003*, pages 123–130.

Mason, J. (2002). SpamAssassin corpus, 2002. *U RL http://spamassassin. apache. org/publiccorpus*.

Méndez, J., Fdez-Riverola, F., Iglesias, E., Díaz, F., and Corchado, J. (2006). Tracking Concept Drift at Feature Selection Stage in SpamHunting: an Anti-Spam Instance-Based Reasoning System. *Proceedings of the 8th European Conference on Case-Based Reasoning, ECCBR-06*, pages 504–518.

Metsis, V., Androutsopoulos, I., and Paliouras, G. (2006). Spam Filtering with Naive Bayes–Which Naive Bayes? *Third Conference on Email and Anti-Spam (CEAS)*, pages 125–134.

Meyer, T.A. and Whateley, B. (2004) SpamBayes: Effective open-source, Bayesian based, email classification system *Proceedings of the First Conference on Email and Anti-Spam (CEAS)* http://ceas.cc/papers-2004/136.pdf .

Oda, T. (2005). *A Spam-Detecting Artificial Immune System*. Masters thesis, Carleton University.

Porter, M. (1980). An algorithm for suffix stripping (1980). *Program*, 14:130–137.

Sahami, M., Dumais, S., Heckerman, D., and Horvitz, E. (1998). A Bayesian approach to filtering junk e-mail. *Learning for Text Categorization: Papers from the 1998 Workshop*, pages 55–62.

Tsymbal, A. (2004). The problem of concept drift: definitions and related work. *Informe técnico: TCD-CS-2004-15, Departament of Computer Science Trinity College, Dublin, https://www.cs.tcd.ie/publications/techreports/reports*, 4:15.

Yue, X., Abraham, A., Chi, Z., Hao, Y., and Mo, H. (2007). Artificial immune system inspired behavior-based anti-spam filter. *Soft Computing-A Fusion of Foundations, Methodologies and Applications*, 11(8):729–740.

Uncertainty and communication complexity in iterated cooperation games

Peter Andras[1]

[1]School of Computing Science, Newcastle University
peter.andras@ncl.ac.uk

Abstract

Iterated cooperation games (e.g. Prisoner's Dilemma) are used to analyze the emergence and evolution of cooperation among selfish individuals. Uncertainty of outcomes of games is an important factor that influences the level of cooperation. Communication of intentions also has a major impact on the outcome of situations that may lead to cooperation. Here we present an agent-based simulation that implements the uncertainty of outcomes together with the communication of intentions between agents. This simulation is used to analyze the relationship between uncertainty and the complexity of the language that the agents use to communicate about their intentions. The complexity of the language is measured in terms of variability of its usage among agents. The results show that more outcome uncertainty implies lower complexity of the agent language.

Introduction

Iterated cooperation games are commonly used to study the emergence and evolution of cooperative behaviour in communities of selfish individuals (Axelrod, 1997). Such games, especially ones that have a non-cooperative single play equilibrium (e.g. Prisoner's Dilemma game (Axelrod, 1997)) can simulate the selfish drive of individuals and offer a natural experimental environment to analyse the effects of various factors on cooperative behaviour.

The main theories about the emergence of cooperation consider as key factors the similarity (relatedness, kinship, joint interest) of individuals and the direct/indirect reciprocity of their behaviour (Axelrod and Hamilton, 1981; Leimar and Hammerstein, 2001; Nowak and Sigmund, 1998; Riolo et al., 2001; Rockenbach and Milinski, 2006; Roberts and Sherratt, 1998; Trivers, 1971). Other important factors include commitment inertia (Roberts and Sherratt, 1998) and segregation of cooperators (Pepper, 2007). Most of these theories assume repeated interactions between the same individuals and/or interactions between all possible pairs of individuals (Axelrod, 1997). These theories also assume well defined outcomes of the played games and usually pay little attention to communicative behaviour of individuals participating in game playing (Axelrod, 1997).

In real life situations the outcomes of cooperation or defection usually are uncertain and depend on many other factors outside of the control of the interacting individuals (Callaway et al., 2002; Pulford and Coleman, 2007; Seghers, 1974; Spinks et al., 2000). Communication between individuals is an integral part of the action selection and decision making process and consequently may matter very much during the interaction process (Dugatking, 1997; Dunbar, 1988). Individuals may try increase cooperation willingness in their partner through communication. They may also use communications to hide their true intentions. Earlier works show that indeed uncertainty of outcomes and communicated intentions may play an important role in determining the level of cooperation in communities of selfish individuals (Andras et al., 2007; Andras et al., 2006; Andras et al., 2003).

Anecdotal evidence suggests that experienced uncertainty, due to the environmental context, has the effect of reducing the complexity of the communication of intentions. For example, surgeons use very restricted language to communicate during operations, the restrictions of the language being aimed to reduce uncertainty and the possibility of misunderstandings in a very uncertain environment (i.e. there may be many unexpected complications during the surgical operation). Another example is the army, where again the communication of orders is done in a highly simplified language, again aimed to reduce uncertainty in the interpretation of orders in the context of a highly uncertain environment (where the soldiers may encounter many unexpected situations created by their enemy).

Here we describe a simulation study aimed to analyse and quantify the effects of uncertainty on communication complexity in the context of situation where cooperation emerges and is maintained in a community of agents. In the agent-based simulation study the agents played Prisoner's Dilemma games. The study confirms the expectation based on the anecdotal evidence, i.e. that more experienced uncertainty implies more reduction in the complexity of the communication language that agents use to communicate their intentions during their interactions. This result helps the understanding of the evolution of the language that is used as medium of interactions between individuals in the context of potentially cooperative social interactions.

The rest of the paper is structured as follows. First we discuss the concept of uncertainty in the context of cooperation games. Next we consider the role of communication of intentions between individuals playing cooperation games. This is followed by the discussion of communication complexity. Next we describe the agent-based simulation environment that we used for our study. This is followed by the presentation of the results of the simulation study. Finally, we end the paper with our conclusions.

Uncertainty in cooperation games

The usual setting of agent-based simulations with iterated cooperation games assumes that the possible outcomes of games are known and fixed across all games played during the simulation (Axelrod, 1997). This assumption is useful to keep the games and the simulations analytically tractable. However, this assumption is frequently not satisfied in real life situations that are modelled by such games and simulations.

In real life situations the more usual is that the outcomes of games vary around some expected outcome. The amount of variation differs from case to case. If the variation is small the situation and its possible outcomes are relatively certain, for example in case of interactions which lead to a contract defining obligations of the parties that determine the outcomes of the interaction (or game). If the variation of outcomes is high the situation and its outcomes are uncertain, for example when army troops are advancing on unknown enemy territory, interactions between soldiers may have widely varying effects.

Uncertainty of outcomes can be represented in a straightforward way in cooperation games by replacing the fixed outcomes by outcome distributions (Andras et al 2007). For example, Table 1 represents a fixed outcome cooperation game.

		Player 1	
		Cooperate	Defect
Player 2	Cooperate	R,r	s,t
	Defect	T,s	p,p

Table 1: A fixed income cooperation game

The letters r, t, s, p stand for: 'reward for cooperation', 'temptation to defect', 'punishment for joint defection' and 'sucker's payoff'. To include the representation of outcome uncertainty the values r, t, s, p are replaced by (normal or exponential) distributions R, T, S, P such that the mean value of these distributions is the corresponding fixed outcome, while the variance of the distributions represents the uncertainty of the game outcomes. When players play the game they pick first their distribution according to their game playing choice and then they pick their actual outcome from this distribution by taking a random sample from the distribution. In average the outcomes of many games will be close to the fixed outcome approximation of the game, but considering outcomes of individual games they will be distributed according to the adopted distributions with a variance corresponding to the uncertainty of the game.

It has been shown that uncertainty in the outcomes of games (due to environmental factors) influences the level of cooperation in the context of agent-based simulations of iterated cooperation games with uncertain outcomes (Andras et al., 2007; Andras et al., 2006; Andras et al., 2003). The stable level of cooperation within a population of agents increases as the uncertainty of the outcomes of the games played by agents increases (Andras et al., 2007). This is consistent with a range of observations of natural situations where the uncertainty imposed by the environments induces more cooperative behaviour among bacteria (Drenkard and Ausubel, 2002; Mehdiabadi et al., 2006), plants (Callaway et al., 2002), and animals (Seghers, 1974; Spinks et al., 2000; Kameda et al., 2002).

Communication of intentions

Commonly agent-based simulations of emergence and evolution of cooperation use cooperation games where the communication between agents is compressed into a single-shot communication expressed by the game playing choice of the agent (Axelrod, 1997). This excludes the communication of intentions or provision of cues about intentions that may influence the other player. However, in real world situations such communications play a critical role in the development of interactions between individuals (Drenkard and Ausubel, 2002; Dugatking, 1997; Dunbar, 1988). (Note, that while in the literally understood Prisoner's dilemma situation there is no possibility of communications, in many situations analysed using this model of interaction there is an important role for communication of intentions.)

It is crucial to include into agent-based simulations the communication of intentions to understand how real world cooperation works. Including communications about intentions may also allow the study of the role of trust and deception in the emergence and evolution of cooperation.

Representing communication of intentions is not trivial in the context of agent-based simulations. To do this the agents have to be equipped with some form of communication language which relates to intentions of the agents and allows the communication about these intentions (exposing or hiding them) in a consistent manner. One way to achieve this is to define the language of agents in the form of a probabilistic grammar with two parallel inputs (Andras, 2003). This grammar can be described using production rules of the form

$$u_{current}, u'_{current} \rightarrow \{u^1_{next} : p_1, \ldots, u^k_{next} : p_k\} \quad (1)$$

Where $u_{current}$ is the last communication symbol produced by an agent, $u'_{current}$ is last communication symbol produced by the interaction partner of the agent, $u^1_{next}, \ldots, u^k_{next}$ are the next communication symbols that can be produced by the agent, and $p_1, \ldots, p_k$ are the probabilities of production of these communication symbols,

$$\sum_{j=1}^{k} p_j = 1 \quad (2)$$

The grammar should include communication symbols representing the start of the communication and symbols representing the play choice of the agents (cooperate or defect). Other symbols may have various semantics depending on the intended semantic extent of the language (e.g. the aim may be inclusion of modeling of trust).

The rules of the language should be such that they are consistent with the practice of communication of intentions in the case of biological organisms. In particular, signs of positive intentions are usually followed by signs of similarly or more positive intentions. Such consistency rules should be implemented in the language of agent communications by imposing consistency constraints on the transition

probabilities. The positivity of a communication symbol is given by the level of pro-cooperation intention (positiveness) indicated by the symbol when it is produced during communication. To express the above rules more formally, if u_0, u_1, u_2, u_3 are communication symbols, such that their positivity ranking is

$$u_1 \leq u_2 \leq u_3 \tag{3}$$

and u_3 can be produced according to production rules

$$u_1, u_0 \rightarrow u_3 : p_1 \tag{4}$$

and

$$u_2, u_0 \rightarrow u_3 : p_2 \tag{5}$$

where p_1, p_2 are the probabilities of application of these rules then

$$p_1 \leq p_2 \tag{6}$$

In other words, more positive symbols are more likely to be followed by even more positive symbols than less positive symbols. Similarly, if the production rules are

$$u_0, u_1 \rightarrow u_3 : p_1 \tag{7}$$

and

$$u_0, u_2 \rightarrow u_3 : p_2 \tag{8}$$

and (3) holds, then again (6) holds, i.e. the same rule applies if the symbols with different positiveness are produced by the communication partner.

Communication complexity

Communication complexity can be defined using the concepts of Kolmogorov complexity (Li and Vitanyi, 1997). The complexity of a description is given by the length of the description. The complexity of a language can be considered in terms of the average length of non-interrupted communications in that language. Of course, this is a relatively rough measure of description and language complexity, but it can be used reliably, assuming no intention aimed to distort the measured complexity. A better approximation of description or language complexity, perhaps, is to consider the description length after the elimination of redundant and irrelevant components from the description or communications. Of course, this may inject some subjective bias into the measurement. In the case of an agent-based simulation with communication of intentions the above defined complexity of the agent language can be measured as the average length of communications between agents that last from the start of the interaction until the decision about the game choice.

An alternative way to measure the complexity of the language is to look at the variability of the language rules used by various individuals. If the rules contain high variability, i.e. there are relatively large differences in the way language rules are used, that indicates high complexity of the language (indirectly indicates high dependence of the rule use on the context of the use). Consistent regular application of language rules without much variation implies lower complexity of the language (i.e. less context-dependence). This alternative approach of measuring complexity of the language fits with the concept of Kolmogorov complexity in the sense that low variation of language rules means that the language can be described listing its rules and relatively few additional meta rules about the context-dependent application of the listed basic rules – i.e. the description of the language is relatively short. In the case of high variation of rule application, the language can be described by listing its basic rules and adding many meta rules about the context-dependent application of basic rules – i.e. the description of the language becomes relatively long. In the case of an agent-based simulation with communication language the measurement of complexity of the language according to this method involves the measurement of the variance of distributions of probabilities of grammar production rules. Higher variance in average across all rules means more variable application of the language and a more complex language. On the other end, lower variance in average means lower variability in language use and lower language complexity.

In the real world high uncertainty situations appear to be associated with low complexity communication languages (e.g. surgical theatre, army – see examples in the Introduction). Generally, the higher lexical complexity and higher complexity of application of rules of a language implies more uncertainty about what is communicated using the language. The uncertainty implied by the complexity of the language adds to the uncertainty imposed by the environment.

On the basis of observation of the link between experienced uncertainty and level of cooperation in communities of selfish individuals we expect that if the community experiences high uncertainty then the possible ways to deal with this is either to have high level of cooperation or to have low level complexity of the communication language, or some combination of these. Earlier work shows that the level of cooperation increases with the level of experienced uncertainty. Similarly we expect that in accordance with anecdotal evidence, the level of complexity of the language should decrease as the experienced level of uncertainty increases.

Simulation implementation

The agents 'live' in a two-dimensional rectangular world, which is wrapped on both pairs of edges (up and down, right and left). A position in the world may be occupied by more than one agent, and positions of agents can be arbitrarily close (i.e. the world is not divided into a grid of disjoint places). The dimensions of the world are set to be 100 x 100.

The agents in the simulation own resources, which are used to maintain themselves and to generate new resources alone or through interaction with another agent. In each time turn each agent tries to choose an interaction partner. The partner is chosen from those agents, which are located close enough (i.e. in the neighbourhood) to the agent which is looking for a partner. An agent may be chosen as a partner if the agent is not already partnered up with another agent. An agent may remain without a partner in a time turn if it cannot find any agent in its neighbourhood which could become its partner. The neighbourhood of an agent is defined as the set of ten closest agents, where the distance between agents is measured in the two-dimensional world populated by the agents.

After finding a partner the agents play a Prisoner's Dilemma type game with uncertain outcomes. The uncertainty

of outcomes represents all uncertainties that may influence the interaction between the agents. The outcome uncertainty is implemented as described in the section 'Uncertainty in cooperation games'. For the sake of simplicity we use normal distributions characterized by a mean value and a variance. Playing the game determines the mean value of the distribution, while the variance of the distribution (σ) is a set value that characterizes the outcome uncertainty of the game. Note that our simulation implements iterated game playing without the requirement that the repeated games should be between the same agents, and in fact it is more likely that agents will play with many other agents during their 'lifetime'.

The agents participate in the game with their available resources, which determine the mean value of the outcome distribution. The function determining this mean value is

$$f(R) = a \cdot \frac{1}{1+e^{-R+R_0}} \tag{9}$$

where a and R_0 are parameters and Ris the amount of available resources. The parameters are set such that the game operates on the convex diminishing return part of the function where

$$f(2x) \geq 2f(x) \tag{10}$$

In order to preserve the Prisoner's Dilemma conditions (i.e. t>r>p>s and 2r>t+s) the game matrix determining the mean values of outcome distributions are set as in section 'Uncertainty in cooperation games' with the values

$$r = f(R) + \frac{\Delta}{2} \tag{11}$$

$$t = f(R) + \Delta \tag{12}$$

$$s = \alpha \cdot f(R) \tag{13}$$

$$p = f(R) \tag{14}$$

where

$$\Delta = \left[f(R_1 + R_2) - f(R_1) - f(R_2) \right]_+ \tag{15}$$

(i.e., it takes only the positive values of the expression in brackets and it is zero if the value of the expression is negative), $0<\alpha<1$ is a parameter.

After determining the mean values of the outcome distributions for both agents, they pick an actual outcome value from the normal distribution determined by this mean value and the variance σ that characterizes the outcome uncertainty of the world of the agents. The actual outcome values will be the new amount of resources available for the agents. Note that the actual outcome value may be below or above the mean value given by the game matrix. If the outcome uncertainty is high (i.e. σ is large) the likelihood of getting much more or much less than the mean value is relatively large. If the outcome uncertainty is low (i.e. σ is small) in most cases the actual outcome value will be close to the mean value determined by the game matrix.

The agents communicate using a simple language. The aim of this communication is to decide how to play the game. The lexicon of the language consists of the symbols: '0','s','i','y','n','h' and 't'. These symbols have the following meanings: '0' – no intention of communication, 's' – start of communication, 'i' – maintaining the communication, 'y' – indication of the willingness to engage into possible cooperation, 'n' – indication of no further interest in communication, 'h' – cooperation (ready to share the benefits of joint use of resources), 't' – cheating (ready to steal the benefits of possible joint use of resources). The last two symbols, 'h' and 't' represent the actual cooperation and defection game choices. The first four symbols are ranked according to their positive contribution towards engagement in cooperation (the least positive is the '0' and the most positive is 'y', '0'≤'s' ≤'i'≤'y').

Each agent has its own realization of the language. The language is represented in the form of a two-input probabilistic production rules according to equations (1) and (2). The implemented simple language contains 22 production rules. The probabilities associated with the production rules may differ between agents, representing the individual realization of the language. For example a probabilistic production rule is

$$i,i' \rightarrow \{y:0.3, i:0.5, n:0.2\} \tag{16}$$

that means that after producing the symbol 'i', and receiving a symbol 'i' from the communication partner, the agent will produce the symbol 'y' with probability 0.3, the symbol 'i' with probability 0.5, and the symbol 'n' with probability 0.2. The probability of the symbol pair ('y','y') being followed by the generation of the symbol 'h' is given by the intention to cooperate of the agent – I_{coop}. The individual realizations of language rules always satisfy the consistency constraints defined by equations (3) – (8).

After selecting an interaction partner the agents may engage in a communication process. The communication process starts properly after both agents communicated the 's' symbol. We set a limit (L_1) for the preliminary communication (i.e. before communicating 's' from both sides). If two agents do not reach the proper start of the communication in a communication of length L_1 they stop further communication and decide to choose play defection in their current game.

The agents use their own realization of the common language to produce communication symbols. The communication process ends either with the communication of an 'n' symbol (i.e., signaling no further interest), or with the communication of the 'y' symbol by both partners (or by automatically stopping the communication according to the set rules). After this each agent decides whether to cooperate or defect by producing the symbol 'h' or 't'. We impose a communication length limit (L_2) on this second stage of communication. If the agents do not reach the communication of 'y' symbols in L_2 steps, they stop their communication and decide to play defection in the current game.

During each communication process, as an agent produces equally or more positive symbols their intention to cooperate increases. The intention to cooperate of the agent increases temporarily and the increased intention of cooperation is valid only for the current communication process. The upgrade equation of the intention to cooperate is

$$I_{coop}(t+1) = 1 - (1-\delta) \cdot (1 - I_{coop}(t)) \tag{17}$$

where $I_{coop}(0)=I_{coop}$, t is the counter of communication symbols produced by the agent so far within the current communication process, and δ is a parameter (δ=0.025).

At the end of each time turn the agents make a random move, i.e. their position is updated according to the equation

$$(x_{new}, y_{new}) = (x, y) + (\xi_x, \xi_y) \quad (18)$$

where (x,y) is the old position of the agent, (x_{new}, y_{new}) is the new position of the agent, and (ξ_1, ξ_2) are random values from a uniform distribution over [-5,5].

The agents 'live' at most for 60 time turns. The agents may die earlier if they run out of resources. When they reach the end of their life they may produce a number of offspring agents. The number of these depends on the amount of resources owned by the agent, more resources implying larger number of offspring. If a dying agent has R amount of resources, and the mean amount of the resources in the agent community at that moment is R_m, and the standard deviation of resources is R_S, then the number of offspring of the agent is calculated as

$$n = \alpha \cdot \frac{R - (R_m - \beta \cdot R_s)}{R_s} + n_0 \quad (19)$$

where α, β, n_0 are parameters of the simulation environment. If n is negative or R=0 this means that the agent has no offspring. If $n>n_{max}$, where n_{max} is the allowed upper limit of offspring, the number of offspring is set to be n_{max}. The offspring of an agent inherit its resources divided equally between them. The locations of the offspring are set by a small random modification of the position of the parent agent.

When agents reproduce at the end of their life, their offspring inherits the language of the parent agent, possibly with some small random modifications of the language rule probabilities. This means that the offspring of an agent will speak the agent language in a very similar manner (using production rules with similar probabilities), which may facilitate cooperation interactions between them.

We ran 20 simulations for each level of outcome uncertainty. Each simulation ran for 400 time turns each time. Each simulation was initialized with 1500 agents with randomly set positions, initial resource amounts, and language transition probabilities.

To summarize, in each time turn the agents search for an interaction partner, and if they find one, they communicate about their intentions and play the above described game to generate their new resource amount. If an agent cannot find a partner it generates its new amount of resources as if it would be playing a defection/ defection game with another agent (i.e. the mean value of the resource value distribution from which it picks its new resource amount is set to be f(R), where R is the amount of its current resources). Agents move randomly at the end of each time turn and deduct from their resource amount a fixed amount of living costs. Agents may die because they run out of resources, or because they reach the end of their life (at most 60 time turns). When an agent dies and still has available resources, it may generate offspring, which will inherit its language with small variation. The offspring initially form a cluster around the place of their parent and gradually move away by random movements. (For more details about the simulation see Andras et al. (2003) and Andras et al. (2006). A version of the simulation code is available as online supplementary information for Andras et al. (2006). For further details and simulation code please contact the author.)

Uncertainty and communication complexity

In earlier work (Andras et al., 2007; Andras et al., 2006; Andras et al., 2003) we have shown that higher outcome uncertainty implies higher level of cooperation in agent populations. This is because the agents share their experienced uncertainty through cooperation, averaging the effective uncertainty that applies to their outcome. This means that through cooperation the effective uncertainty experienced by agents within the agent community is reduced compared to the individually experienced uncertainty that would apply to them without involvement in cooperative interactions (Andras et al., 2006). In order for the agent population to reproduce there is a critical level of outcome uncertainty, above which the population shrinks until it goes extinct. If the outcome uncertainty imposed by the environment is high, high level of cooperation is required to bring down the effective uncertainty to or below the critical level (Andras et al., 2006). Consequently, higher outcome uncertainty implies higher level of cooperation that is required to keep the population away from extinction. The current simulation confirms this earlier finding (see Figure 1). Note that this relationship between outcome uncertainty and the level of cooperation is valid even if there is no communication of intentions (Andras et al., 2007).

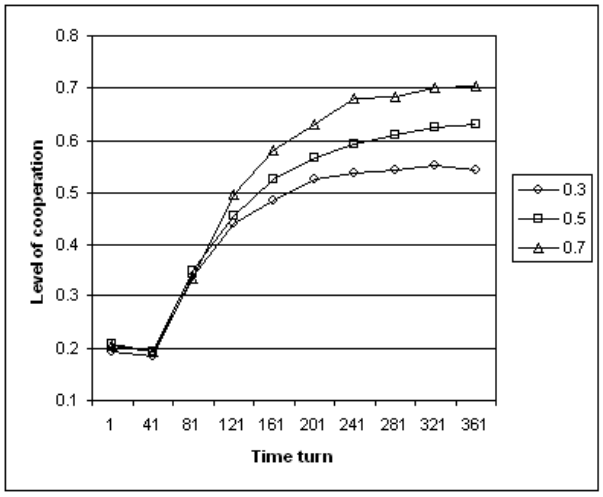


Figure 1: The relationship between outcome uncertainty and level of cooperation. The three lines show the evolution of the average level of cooperation across 20 populations of agents for three levels of outcome uncertainty (σ=0.3, σ=0.5, σ=0.7 – the box on the right indicates the corresponding lines). The level of cooperation is measured as the percentage of joint cooperation decisions among all game decisions made by agents in a given time turn. (Error bars are omitted as standard deviations are relatively small)

Here we investigate the relationship between the outcome uncertainty and the complexity of the language that the agents use. Our expectation is that higher outcome uncertainty implies lower language complexity. To measure the complexity of the language used by agents we adopt the approach introduced earlier based on the measurement of the

variation of the use of the language rules. In other words, we measure the complexity of the language used within an agent population as the average of the variances of probabilities characterizing the production rules of the agent language. For each language production rule

$$R_i : u^i_{current}, u'^i_{current} \rightarrow \{u^{i,1}_{next} : p^i_1, \ldots, u^{i,k}_{next} : p^i_{k_i}\} \quad (20)$$

we consider all realizations of this rule (i.e. each realization is a realization of the rule in a 'living' agent) and calculate the variance for each involved probability p_1^i, …, p_k^i. Let us denote these variances as

$$\sigma^i_1, \ldots, \sigma^i_{k_i} \quad (21)$$

then the complexity of the language is defined as

$$cx = \frac{1}{K} \sum_{i=1}^{L} \sum_{j=1}^{k_i} \sigma^i_j \quad (22)$$

where L is the number of language rules (in the simulation we have 22 such production rules), and

$$K = \sum_{i=1}^{L} k_i \quad (23)$$

Using this measurement of language complexity we found that indeed higher level of outcome uncertainty implies lower level of language complexity in the context of our simulated agent communities. This result is presented in Figure 2. This confirms our expectation.

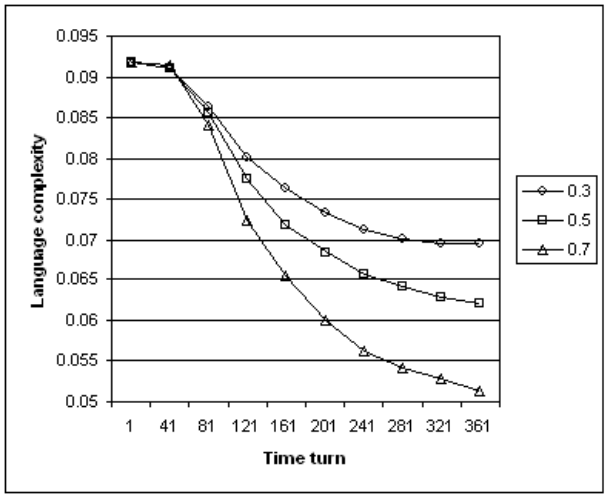


Figure 2: The relationship between outcome uncertainty and language complexity. The lines show the evolution of average language complexity across 20 populations of agents for three levels of outcome uncertainty (σ=0.3, σ=0.5, σ=0.7 – the box on the right indicates the corresponding lines)). The language complexity is measured according to equation (22). (Error bars are omitted as standard deviations are relatively small)

We also considered the alternative measure of the language complexity, i.e. the average length of communication processes that lead to the reaching of the cooperation / defection decisions. However this measure gives much less clear results, as the length of communication processes drops to around the same level (in average) at all considered levels of outcome uncertainty. The most likely reason for this is that the language is very simple and has very few communication symbols. Consequently, there is little variation that could exist in terms of communication process length between surviving agent communities 'living' in environments characterized by different outcome uncertainty. The language use variation based measure (the cx measure defined above) appears to be more sensitive to detect differences in language complexity between agent communities dealing with different levels of uncertainty.

The principial reason behind the observation of the lower language complexity in agent communities dealing with more outcome uncertainty is that lower language complexity adds less to the uncertainty of the world than higher language complexity, and consequently the lower language complexity is preferred in more uncertain environments. In practical terms, analyzing the evolution of simulated agent populations, we note that an important aspect is that surviving offspring of successful agents are clustered at the time of their creation. Having very similar language facilitates their continual success, especially if they inherited sufficiently cooperative inclinations from their parent. A more uncertain environment means stronger selection for successful individuals with relatively high cooperative inclination, which means that clusters of related agents have increased selection advantage in such environments. This is likely to contribute significantly to the reduction in the variability of the language usage that we adopted as a complexity measure of the language.

The presented results are about agent-based simulations. They confirm our expectation about the relationship between outcome uncertainty and language complexity and provide some explanation about why this is the case. However, to fully confirm our theoretical expectation about the effect of uncertainty on language complexity ideally we would need to consider real-world data. While it is not easy to find or collect relevant real-world data, we note that measurements of language diversity in naturally more and less uncertain areas of Africa (semi-desert in Northern Nigeria and rainforest in Burkina Faso) indicate that natural experimental confirmation of the presented results may be within reach (Nettle 1998; Nettle 1996). Nettle (1996, 1998) has shown that in the more arid and hostile semi-desert are the number of languages is much smaller than in comparable much less uncertain (in terms of availability of food) areas of rainforest. This appears to be in good agreement with our expectation and simulation results.

Measuring complexity of natural languages is not very obvious. We considered in this paper two measures and have shown that the one based on variability of language use seems to be more sensitive to measure complexity differences between agent languages. Applying similar measures to natural languages may lead to robust measures of language complexity. Our results indicate that language complexity is likely to be linked to the level of cooperativeness within a community. Consequently, analysis of complexity of language used for example in companies may help the understanding of the potential of the analyzed organization to deal with their experienced uncertain environment and to harness

organizational resources that can be mobilized through cooperation.

Finally, we underline that our analysis and simulation is focused on the lexical complexity of the language used to inform about intentions (i.e. complexity in the sense of the variability of use of lexical components – communication symbols in the context of the simulations). We did not consider grammatical complexity (i.e. the number and combinatorial variability of rules), as in our case the number of rules is always fixed. A more extensive analysis of language complexity and more complicated simulation would be needed to consider aspects of grammatical complexity. We expect that losses suffered in terms of lexical complexity, imposed by the necessity of dealing with an uncertain environment, are compensated by increased complexity at the level of grammar in the longer run. The reason of this is that having more complex grammar increases the computational capacity of the language which may be beneficial in a more uncertain environment. This increase in grammatical complexity is supported by the decrease in lexical complexity in the sense that less ambiguity in the lexicon reduces the likelihood of inappropriate application of grammatical rules. The investigation of this conjecture is not the subject of this paper.

Conclusions

We have shown in this paper that higher outcome uncertainty implies lower level of language complexity in the context of agent-based simulations of social interactions conceptualized as playing iterated Prisoner's Dilemma games. The complexity of the language was measured in terms of variability of the use of the language, and in particular in terms of variability of 'meanings' of lexical units of the language.

Considering that we modeled repeated social interactions through the iterated game playing, our result implies that in the case of social situations with high outcome uncertainty we expect a reduction in complexity of the language usage. More specifically, we expect a reduction in the range of possible/acceptable ways of usage ('meanings') of words, and possibly also an effective reduction of the size of the lexicon of used words. This matches well with the anecdotal evidence about very high outcome uncertainty environments like a surgical theatre or an army.

Data about variability of languages over larger geographical territories also suggests that our finding about the link between uncertainty induced by the environment and the (lexical) complexity of the language used by humans to live in these geographical areas is valid. Of course, this needs to be checked further and confirmed numerically on the basis of the data.

Our result indicates that environment induced uncertainty (represented as outcome uncertainty in our agent-based simulation study) plays an important role in the evolution of languages. This uncertainty implies complexity constraints on the language, which limit the lexical variability of the language. Such constraints may explain simplification of a language used in high uncertainty context and may also explain the variability of human languages in geographical areas characterized by high or low uncertainty implied by available resources (e.g. food, shelter, etc.).

Our analysis did not extend to cover the grammatical complexity of languages. This would need more complicated simulations allowing the change of the grammar (and symbol set of the lexicon) of the language used by agents. However, we conjecture that less lexical complexity may be compensated by more grammatical complexity in languages used in more uncertain environments.

Finally, our investigation of the link between outcome uncertainty and language complexity shows that the approach to measure language complexity as the average length of communication processes may not be sensitive enough to measure the effects of environment induced uncertainty. The proposed and used complexity measure which measures the variability of the usage of lexical elements is a more appropriate measure for this task, and possibly it is generally a more appropriate measure to measure lexical complexity of natural languages.

References

Andras, P., Lazarus, J., Roberts, G. (2007). Environmental adversity and uncertainty favour cooperation. *BMC Evolutionary Biology*, 7:240.

Andras, P., Lazarus, J., Roberts, G., and Lynden, S. J. (2006). Uncertainty and cooperation: Analytical results and a simulated agent society. *JASSS – Journal of Artificial Societies and Social Simulation*, 9:1/7.

Andras, P., Roberts, G., and Lazarus, J. (2003). Environmental risk, cooperation and communication complexity. In Alonso, E., Kudenko, D., and Kazakov, D., editors, *Adaptive Agents and Multi-Agent Systems*, pages 49-65. Springer-Verlag, Berlin.

Axelrod, R. (1997). *The Complexity of Cooperation: Agent-Based Models of Competition and Collaboration*. Princeton University Press, Princeton, NJ.

Axelrod, R., and Hamilton, W. D. (1981). The evolution of cooperation. *Science*, 211:1390-1396.Callaway, R. M. et al. (2002). Positive interactions among alpine plants increase with stress. *Nature*, 417:844-847.

Drenkard, E., and Ausubel, F. M. (2002). Pseudomonas biofilm formation and antibiotic resistance are linked to phenotypic variation. *Nature*, 416:740-743.

Dugatkin, L.A. (1997). *Cooperation Among Animals. An Evolutionary Perspective*. Oxford University Press, New York.

Dunbar, R. (1988). *Primate Social System*. Croom Helm, London.

Kameda, T., Takezawa, M., Tindale, R. S., and Smith, C. M. (2002). Social sharing and risk reduction – Exploring a computational algorithm for the psychology of windfall gains. *Evolution and Human Behavior*, 23:11-33.

Leimar, O. and Hammerstein, P. (2001). Evolution of cooperation through indirect reciprocity. *Proceedings of the Royal Society of London Series B: Biological Sciences*, 268:745-753.

Li, M. and Vitanyi, P. M. B. (1997). *An Introduction to Kolmogorov Complexity and its Applications*. Springer, New York.

Mehdiabadi, N. J., et al (2006). Social evolution: Kin preference in a social microbe. *Nature*, 442:881-882.

Nettle, D. (1998). Explaining global patterns of language diversity. *Journal of Anthropological Archaeology*, 17:354-374

Nettle, D. (1996). Language diversity in West Africa: An ecological approach. *Journal of Anthropological Archaeology*, 15:403-438

Nowak, M. A., and Sigmund, K. (1998). Evolution of indirect reciprocity by image scoring. *Nature*, 393:573-577.

Pepper, J. W. (2007). Simple models of assortment through environmental feedback. *Artificial Life*, 13:1-9.

Pulford, B. D. and Coleman, A. M. (2007). Ambiguous games: Evidence for strategic ambiguity aversion. *Quarterly Journal of Experimental Psychology*, 60:1083-1100.

Riolo, R. L., Cohen, M. D., and Axelrod, R. (2001). Evolution of cooperation without reciprocity. *Nature*, 414:441-443.
Rockenbach, E. and Milinski, M. (2006). The efficient interaction of indirect reciprocity and costly punishment. *Nature*, 444: 718-723.
Roberts, G. and Sherratt T. N. (1998). Development of cooperative relationships through increasing investment. *Nature*, 394:175-179.
Seghers, B. H. (1974). Schooling behaviour in the guppy (Poecilia reticulata): an evolutionary response to predation. *Evolution*, 28:486-489.
Spinks, A. C., Jarvis, J. U. M., and Bennett, N. C. (2000). Comparative patterns of philopatry and dispersal in two common mole-rat populations: implications for the evolution of mole-rat sociality. *Journal of Animal Ecology*, 69:224-234.
Trivers, R. L. (1971). The evolution of reciprocal altruism. *Quarterly Review of Biology*, 46:35-57.

Investigating Patterns for the Process-Oriented Modelling and Simulation of Space in Complex Systems

Paul S. Andrews[1], Adam T. Sampson[3],
John Markus Bjørndalen[4], Susan Stepney[1], Jon Timmis[1,2], Douglas N. Warren[3] and Peter H. Welch[3]

[1]Department of Computer Science, University of York, UK, YO10 5DD
[2]Department of Electronics, University of York, UK, YO10 5DD
[3]Computing Laboratory, University of Kent, Canterbury, UK, CT2 7NF
[4]Department of Computer Science, University of Tromsø, Norway
psa@cs.york.ac.uk, A.T.Sampson@kent.ac.uk

Abstract

Complex systems modelling and simulation is becoming increasingly important to numerous disciplines. The CoSMoS project aims to produce a unified infrastructure for modelling and simulating all sorts of complex systems, making use of design patterns and the process-oriented programming model. We provide a description of CoSMoS and present a case study into the modelling of space in complex systems. We describe how two models – absolute geometric space and relational network space – can be captured using process-oriented techniques, and how our models can be refactored to allow efficient, distributed simulation. We identify a number of design, implementation and refactoring patterns that can be applied to future complex systems modelling problems.

Introduction

Complex systems consist of populations of low-level simple agents that interact concurrently with each other and their environment to exhibit high-level emergent behaviours. The modelling and simulation of complex systems is becoming increasingly important in a number of scientific disciplines. Real-world experimentation is often expensive and time-consuming; accurate simulations provide a powerful tool for understanding complex systems, and their results can help to direct future experimental work. There is therefore significant interest in the development of more effective tools and methodologies for modelling and simulation.

Under the banner of CoSMoS[1] (Complex Systems Modelling and Simulation infrastructure), we aim to develop a modelling and simulation infrastructure to allow complex systems to be designed, analysed and explored within a uniform framework. When completed, the CoSMoS system will allow users, guided by our methodology, to design, develop and analyse their own complex systems.

Our modelling process aims to be applicable to *generic* complex systems, and will make use of *patterns* and *refactorings*. Our simulation environment will be *massively-concurrent* and *distributed* through the use of the process-oriented programming model. This is important as our final infrastructure will be supported on a number of processing platforms including FPGAs, general-purpose PCs and clusters. We are adopting a case-study-based approach, modelling and simulating many complex systems to identify the necessary generic components. As we develop the case studies, we are consciously documenting and analysing how we are developing the models and simulations to extract the CoSMoS process. Through each case study this process is refined and augmented as new situations arise.

A number of tools for complex systems modelling and simulation already exist; for example, environments such as Breve[2] and Repast[3] allow for the development of agent-based complex systems simulations. The use of design patterns to document reusable solutions to complex systems modelling problems has previously been advocated by Wiles et al. (2005). CoSMoS differs in that it will bring together the modelling, simulation and analysis of generic complex systems under a single unified framework. Additionally, the massively-concurrent simulation environment will enable us to get closer to the scale of real-world complex systems.

In the context of CoSMoS, this paper presents a rationale for our approach and describes some initial steps towards achieving our aims. We start by describing why a process-oriented approach is applicable to complex systems, followed by some of the techniques we are employing in the pursuit of engineering reusable elements. We then present an investigation into space representations in various complex systems, and show how space can be modelled and simulated using a process-oriented approach. Finally we look at what our case study has shown us in relation to the aims of CoSMoS by identifying the kinds of patterns and refactorings that might be applicable to general complex systems.

A Process-Oriented Approach

In the process-oriented programming model, concurrent *processes* interact using mechanisms such as *channels* and

[1]http://www.cosmos-research.org

[2]http://www.spiderland.org

[3]http://repast.sourceforge.net

barriers. Process-oriented programming has a formal basis in process algebras such as CSP (Hoare, 1985) and the π-calculus (Milner, 1999). As a result, the semantics of communication and process composition are well-defined, and the behaviour of process-oriented programs can be reasoned about in a structured way. See Welch et al. (2006) for one example of this.

As the real world consists of concurrent, interacting entities, communicating process calculi – and therefore process-oriented programs – provide a natural way to construct models of the real world: entities map directly to processes, and the interactions between them are modelled by communications. There has been much research into modelling complex systems using process calculi such as the π-calculus (Phillips and Cardelli, 2007); this suggests that process-oriented programming techniques can be profitably applied to complex systems modelling and simulation.

Concurrent programs are well-placed to take advantage of multicore processors and multiprocessor hosts. The same programming models can be used to construct distributed systems; a process-oriented program can usually be refactored into a form that runs efficiently on a cluster.

Concurrency is traditionally seen as hard for programmers to get right, but this need not be the case. In a process-oriented system, a compiler can guarantee that processes are isolated from each other, and must communicate explicitly rather than sharing data. Processes may communicate references to data, but sending a reference to another process causes it to be lost from the sender (*data mobility*). These constraints combine to prevent common concurrency problems such as aliasing errors and race hazards. Other concurrency problems such as deadlock can be dealt with using simple design rules, proved correct by reference to the underlying process calculi.

One such set of design rules is the *client-server* pattern, in which client processes are connected to server processes by two-way bundles of channels. After the client initiates a communication, an arbitrary two-way conversation can take place between the client and server. Processes may act as both client and server, but they may be a client to only one server at a time. If there are no cycles in the directed graph of client-server relationships between a network of processes, the network will be both deadlock- and livelock-free (Martin and Welch, 1997). Many common patterns of concurrency – such as pipelines – already obey the client-server rules, but the rules also allow far more complicated process networks to be constructed safely.

Initial work on CoSMoS has used the occam-π process-oriented programming language[4]. In an occam-π implementation, process overheads are typically very small; a commodity PC can support millions of concurrent processes. Process creation and deletion is cheap, and communication is very efficient. This allows the programmer to take advantage of concurrency to simplify their program without worrying about adversely affecting performance. occam-π provides *channel bundles* as a language binding for client-server relationships; the endpoints of channel bundles can be communicated around at runtime (*channel mobility*), allowing dynamically constructed and reconfigurable networks.

[4] http://occam-pi.org/

Engineering Reusable Complex Systems

To develop an engineering approach to the modelling and simulation of generic complex systems, we must focus on *reusable* problem-solving techniques. Reusable techniques reduce the amount of work required in the development of a complex system, and lessen the risk of mistakes during specification or implementation. Additionally, because our systems will be built using common building blocks, it will be easier to combine models and simulations to study interactions between complex systems.

Our main tool for achieving reusability is one of the most successful and popular approaches in software engineering: patterns. The original idea of patterns comes from architecture courtesy of Alexander et al. (1977), who describe a pattern as "a problem which occurs over and over again in our environment, and then describes the core of the solution to that problem, in such a way that you can use this solution a million times over, without ever doing it the same way twice". This idea was applied to object-oriented software design by Gamma et al. (1995), who identify four essential elements of a pattern:

Name: a brief phrase to summarise the pattern, and that can be used as part of a *pattern language* when discussing problems.

Problem: the situation in which the pattern may be applied.

Solution: the elements involved when solving the problem, and a guide to their implementation.

Consequences: the advantages and disadvantages of applying the pattern, allowing the designer to make a decision about the appropriateness of the pattern in their particular situation.

Most existing uses of patterns in software engineering use the object-oriented programming model, but patterns can be applied equally effectively to process-oriented programming and other models.

Although the original use of patterns in software engineering was at the design stage (hence "design patterns"), patterns have been developed for all stages of the software development process, from low-level coding right up to the design of development processes themselves. For example, antipatterns (Brown et al., 1998) can be used to document common mistakes and how they can be avoided and rectified. Other patterns include analysis patterns (Fowler, 1997),

coding patterns (Beck, 1997), and metapatterns that describe patterns themselves. Our modelling process aims to take advantage of patterns wherever possible, and in particular to develop pattern languages for: abstract computational representations of complex systems models; analysis of collective and emergent properties; and validity argument structures.

We are not alone in wanting to apply patterns in the field of complex systems. Wiles et al. (2005) suggest that attention to software engineering practice can benefit both modellers and biologists. We are in complete agreement with their assertion that the field of *in silico* modelling is reaching a point where common practices should be identified and formalised into patterns.

Refactoring is a particularly interesting concept from a patterns perspective. Refactoring is improving the structure of a model or program without changing its external behaviour. For our purposes, such refactorings might include improving the clarity of the model, or adapting simulations to take advantage of FPGAs or clusters. These approaches will be codified as refactoring patterns.

Space: a Case Study

As noted in the introduction, we have employed a case-study-based methodology in our investigation of the issues surrounding the modelling and simulation of complex systems. For the purposes of discussion in this section, we define a model to be an abstract logical representation of something we wish to better understand. A (computer) simulation is the execution of such a model over time, allowing us to analyse the model's behaviour.

The case study we describe here deals with the representation of space in a variety of different "textbook" complex systems. By "textbook", we mean well-understood examples of complex systems from the literature, such as boids (Reynolds, 1987), artificial ant behaviour (Amos and Don, 2007), L-systems (Prusinkiewicz and Lindenmayer, 1990) and scale-free networks (Barabasi and Bonabeau, 2003). We have chosen to study space initially because we feel that it is a universal property of complex systems models – and one that can be expressed in a wide variety of ways. (Future case studies will cover time and other commonalities.)

For millennia, philosophers have pondered the nature of space in the real world. At the turn of the eighteenth century, two very different views on space were held by Newton and Leibniz. Newton believed that space was *absolute* – independent of the objects that could exist within it. Leibniz defined space as *relational* – only existing in the relationships among the objects it contains (Garber, 1995; Giavotto and Michel, 2002).

In complex systems modelling, space is commonly represented using either of the Newton and Leibniz approaches. For example, simulations such as boids and artificial ants use an absolute geometric space, with a fixed area of space being defined in which all the agents reside. L-systems and scale-free networks, on the other hand, use a relative model of space: the model's idea of space comes solely from the relationships between L-system symbols or network nodes, and the geometry and size of space can change over time. Hybrid models of space can be built in which absolute geometric space is modelled by a sparse network of regions, created only when an agent or a behaviour requires their presence (Sampson et al., 2005).

The meaning of points in a model's representation of space may correspond to locations in physical space (with one, two or three dimensions), or to something more abstract. For example, a point in absolute "shape-space" (Perelson and Oster, 1979) could represent a set of parameters describing the shape of a molecule in an immunological model (Hart and Ross, 2004).

Space models such as absolute geometric space may be continuous, or quantised into a grid of discrete positions. Some models may be built on either space representation; for example, interactions between cells in the bloodstream may be modelled using distance calculations in continuous space, or by looking at neighbouring cells in discrete space. The choice of space representation – and, if a discrete model is used, the fineness of the grid – will often affect the dynamics of the model.

Previous Work

The TUNA project[5] was the feasibility study that led to CoSMoS, investigating tools for engineering emergent behaviour in nanite systems. The primary case study was the simulation of artificial blood platelets, which would staunch wounds in a blood vessel as an emergent behaviour. A number of different models were considered.

Initial efforts focused on cellular automata in one, two and three dimensions. The first models used simple, deterministic rules, and were built using the CSP process calculus and analysed using FDR and Probe[6]. Later models were extended to include platelet activation and diffusion of chemical factors, and were implemented using occam-π. The completed blood clotting simulation (Ritson and Welch, 2007) runs in three dimensions, using VTK[7] for volumetric visualisation, and allows the user to interactively create wounds in the blood vessel. The program can be distributed across a cluster of commodity PCs, enabling simulations with tens of millions of agents to run at acceptable speeds. The resulting simulation demonstrates several behaviours seen in real-world haemostasis, and can be used to perform simple *in silico* experiments.

Design patterns were developed for the efficient simulation of grid-based space using process-oriented techniques, in which space cells are represented as processes (Sampson et al., 2005). The construction of space processes can

[5] http://www.cs.york.ac.uk/nature/tuna/

[6] http://www.fsel.com/software.html

[7] http://www.vtk.org/

be delayed until agents move into them, allowing a sparse, lazy representation of space. Agents can "sleep" by not engaging in synchronisation when their state is unlikely to change, saving processor time. Concurrent access to shared resources can be managed safely using barrier synchronisation and *phases* (Barnes et al., 2005). Agents can migrate transparently between different hosts in a cluster.

Modelling Continuous Space

The TUNA project examined grid-based models of space. These are easy to reason about, but they are insufficiently accurate for the purposes of many interesting models. Continuous space is generally more useful when modelling real-world systems. In continuous space, it immediately becomes harder for agents to find nearby agents with which to interact; they cannot simply look in the neighbouring locations, but must consider the distances between them. In a trivial implementation of continuous space, all agents have knowledge of all other agents, but this is inefficient (as well as a poor model of the real world); we need a representation of the world with an idea of locality. We would also like to be able to take advantage of the space-modelling efficiency patterns that were developed for TUNA.

As a first case study, we implemented the simulated bird flocking model *boids* (Reynolds, 1987). At each time step, boids adjust their velocity based on the following rules:

Collision Avoidance: avoid collisions with nearby objects

Velocity Matching: try to match velocity with nearby boids

Flock Centring: try to stay close to nearby boids

This results in the emergent flocking behaviour. We used a two-dimensional model of space, although boids (and our space model) would work equally well in three dimensions.

The approach we took was to divide the space up into regions, with each region represented by a *location* process. Each location contains an arbitrary number of *agent* processes (boids and obstacles), and keeps track of a local position for each, relative to the centre of the region. Locations are connected much as cells are in a grid-based model; each location process has a shared channel bundle which its neighbours have access to, and provides a server interface that allows clients to enter a cell, move around, and retrieve a list of agents along with their positions.

The first thing that each boid must do on each timestep is to "look around" for other agents in its neighbourhood. To do this, the boid needs to gather the contents of all the cells that intersect with the region it can see. We have restricted our agents to seeing a circular region with a diameter of at most one location, which means that it is sufficient to look at the location the agent is in and the eight surrounding locations. Figure 1 shows a boid's field of vision in the partitioned continuous space model.

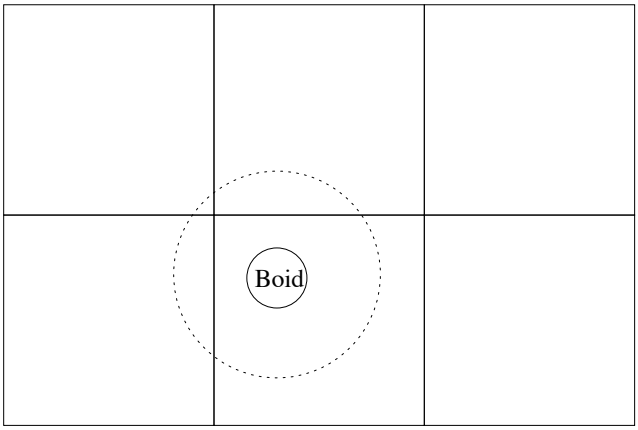


Figure 1: A boid's field of vision (dotted line) in the partitioned continuous space

Since all agents in each location need to look at the same set of nine locations, we can save some effort by delegating this task to a shared *viewer* process. Each location has a viewer process permanently attached to it, and on each time step the viewer updates its view of the surrounding world. The viewer process then provides a server interface to the agents in the corresponding cell which allows the agents to obtain their local view.

In order to guarantee that the agents see a consistent view of the world, we must make sure that all the viewers are updated after the agents have finished moving, but before they look again at the start of the next time step. We therefore divide each timestep into multiple phases, with a global barrier synchronisation between each phase:

- In phase 1, the viewers request the contents of the surrounding cells.
- In phase 2, the agents request their view from the viewers, compute their new velocity, and send movement messages to their locations.

Once a boid has looked around, it decides in which direction to move by sending a movement vector to its location. The location responds by updating the boid's position. If the boid remains within the same location, no further action is necessary. However, if the boid has moved outside the bounds of the location, it must be moved into the next location in the correct direction. This is achieved by the response to a boid's movement request being "you must move into this location". This approach makes it possible to move across multiple locations in one movement step: upon entry, the first new location can respond immediately with another "you must move into this location" message, thus the agent reaches the correct target location by an iterative process.

In order to avoid complicating every agent with code to handle movement, we inserted an additional *agent manager*

process that hides the details of this from the agent itself. The manager provides a simplified interface to the agent, supporting only "move" and "look" requests. Adding this level of indirection simplified later work: it made it possible to have arbitrary behaviour inside the space model that is not visible to the agent itself.

Distributed Continuous Space

We used *pony* (Schweigler, 2006) to distribute the boids simulation across a cluster of hosts, with each host simulating a rectangular region of space modelled by several location processes. *pony* provides networked channel bundles for occam-π programs, with exactly the same semantics as local channel bundles; the only visible difference is the significantly increased latency compared to local communications. To obtain good performance, it is generally best to engineer a distributed application in such a way as to minimise the number of cross-host communications, and perform cross-host communications in parallel as far as possible.

To start with, we just modified the existing simulation to set up the same network of processes across a distributed application. The resulting simulation worked exactly as before, but ran very slowly; furthermore, it got even slower as boids migrated between hosts. There were two major sources of inefficiency:

- Neighbouring viewer processes must request the same view information from a location on the other side of a network link. For local communications this is not a problem, since only a reference is transferred; for network communications the data must be copied.

- More seriously, agent processes continue to run on the host they were started on, so once moved to a new host, every communication they do is across a network link.

To solve the viewer problem, we applied the *remote proxy* distributed computing pattern (Roth, 2002) in the form of *ghost* processes, which cache the contents of a location on the other side of a network link. Viewer and agent processes that would ordinarily communicate with a remote location are instead given a channel bundle to the corresponding local ghost (which provides the same server interface as the remote location). Since ghost processes must update their cached contents before viewer processes try to read it, we needed to introduce an additional phase to the simulation:

- In phase 1, the ghosts request the contents of their corresponding locations.

- In phase 2, the viewers request the contents of the surrounding cells.

- In phase 3, the agents request their view from the viewers and send movement messages to their locations.

To solve the agent problem, we introduced the idea of *agent migration*. In response to moving to a location on a different host, an agent can be told to suspend itself: pack up its internal state and terminate on the originating host. The state is moved to the destination host, where a new agent process is started using the existing state. This is straightforward to implement: when an agent attempts to move into a ghost (rather than a real location), the ghost replies to the agent with a "suspend" message, and then signals the real target location to spawn a new process.

A sample process network at a host boundary in the final model is shown in Figure 2. The cycle time of the resulting simulation is approximately equal to that of the single-host simulation plus the network latency. We ran the simulation across a cluster of networked PCs: the cycle time remained approximately constant as the simulation was scaled from two to eight hosts. This is as expected, since each host only needs to communicate with its immediate neighbours.

In order to increase performance further, we have experimented with more efficient strategies for inter-host communication. Relaxing the normal CSP channel semantics for networked channels to permit *asynchronous* delivery of messages means that channel communications no longer need to be acknowledged by the receiving host, approximately halving the network latency and thus reducing the simulation cycle time. Since network channels are only used by the ghost processes, we can simply adjust the ghost protocol so that it still behaves correctly with asynchronous communication. In the future, we plan to experiment further with batching of messages in order to reduce TCP overheads and permit message compression.

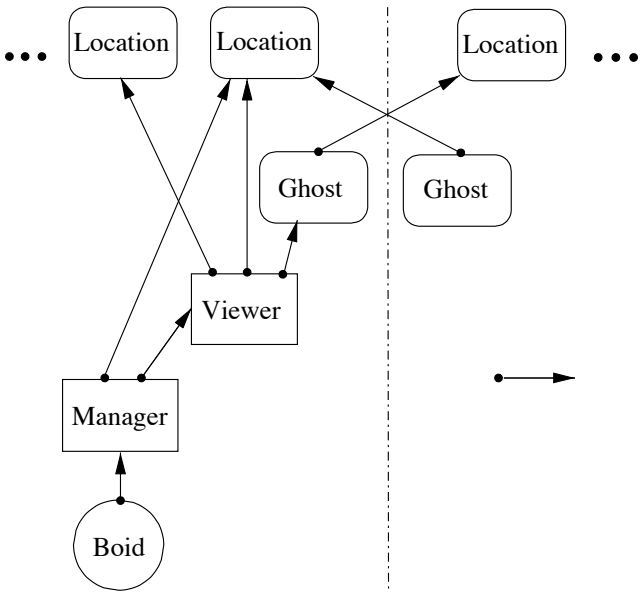


Figure 2: Process network at a host boundary in distributed boids

Different Model, Same Space

To demonstrate the reuse of our space model, we implemented a different complex system simulation on top of our continuous space model. The complex system we chose was ant-based annular sorting (Amos and Don, 2007), in which ants sort eggs into rings by size by picking up poorly-placed eggs and dropping them when they find a better location.

As with boids, we modelled ants and eggs as agent processes. To allow ants to carry eggs, we extended our system so that agents could pick up other agents (removing them from their locations), and put them down elsewhere. No further changes were necessary to the space model, suggesting that it had potential for reuse in other similar simulations.

Our model of continuous space is generic enough that it supports different kinds of agent with differing behaviours. They might sense different aspects of the environment and have different goals. The location and viewer processes report the locations of all nearby agents, whereas the agent manager and agent processes provide the specific agent behaviour. Additional sensory modes and/or noise could be added to the underlying location and viewer architecture, which agent processes can filter accordingly.

Constructing Network Space: Edges as Channels

Network space is an example of relational space in which the network nodes are the space-defining objects. Networks can be modelled very straightforwardly in a process-oriented way: nodes are processes, and edges are channels. As an example, we implemented L-systems (Prusinkiewicz and Lindenmayer, 1990): rewriting systems based on a formal grammar (a set of rules and symbols) that can be used to model growth processes such as plant development and organism morphology. For example, a very simple grammar might be defined as follows:

Symbols: $A, B, +, -$

Start symbol: A

Rules: $(A \rightarrow B - A), (B \rightarrow A + B)$

Here we have two symbols A and B which are transformed by the corresponding rules, and two symbols $+$ and $-$ which do not change. By specifying a start symbol, we can iteratively apply the L-system rules in parallel so that our symbol string grows with each iteration as follows:

Iteration 0: A

Iteration 1: $B - A$

Iteration 2: $A + B - B - A$

Iteration 3: $B - A + A + B - A + B - B - A$

L-systems are often visualised by translation into "turtle graphics" instructions. For example, if we define variables to mean "move straight ahead one unit", $+$ to mean "turn right $60°$" and $-$ to mean "turn left $60°$", we end up with the visualisation shown in Figure 3.



Figure 3: Example L-system after 4 iterations

We model an L-system as a process network in which the L-system symbols are represented as separate processes connected by channels. Here, the channels provide the ordering of the L-system string. At each iteration of the L-system, each process holding a variable symbol applies its corresponding transition rule, and replaces itself with the processes and channels corresponding to the expansion of the symbol.

Figure 4 shows a step-by-step application of the $B \rightarrow A + B$ transition rule. In the first step a new process is spawned that contains the last symbol in the transition rule. This process is then given the end of the right hand channel, and a new channel is created to connect this new process to the original B process that is being transformed. We work from the right-hand side of the rule so that the process network re-configures from the inside, with the right hand channel to the rest of the process network only having to be rewired once. After the first new process is connected, we work through the transition rule creating new processes and channels and reconfiguring existing channels where necessary. So, in our example, the next step involves inserting a process containing a $+$ symbol. As a last step, the original process that has undergone the transition rule has its symbol changed to the leftmost symbol in the rule – in this case, B is changed to an A. New processes are created on demand by a *factory* process.

The simulation is visualised on each iteration using a display process, which is connected to both ends of the chain of symbol processes, forming a ring. To visualise the network, the display process sends a channel end to the first symbol process, which outputs its symbol down the channel, then passes the channel end on to the next process. This repeats until the channel end makes it all the way around the ring and returns to the display process, which then knows it has gathered the complete state of the network, and draws it to the display using turtle graphics rules.

Constructing Network Space: Edges as Processes

A scale-free network is one in which some nodes are highly connected, whilst most have few connections. There is no notion of a typical node in the network: its properties are independent of the number of nodes. Examples include

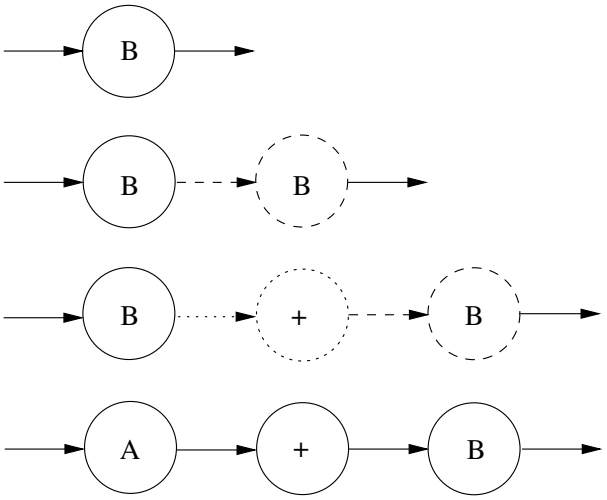


Figure 4: Network reconfiguration during a rule application

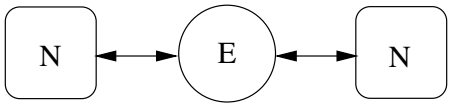


Figure 5: An implementation of an undirected edge, where N denotes a node process and E an edge process

the World-Wide Web: nodes are pages, and edges are hyperlinks; research collaborations: nodes are scientists, and edges are co-authorships; and protein regulatory networks: nodes are proteins, and edges are interactions amongst proteins (Barabasi and Bonabeau, 2003). They are scale-free because their properties are similar regardless of how many nodes are present in the network; for example, the distribution of path lengths between pairs of arbitrary nodes will not change as more nodes are added. Barabasi et al. (2000) have shown that by utilising a scheme known as preferential attachment – in which nodes prefer to connect to other nodes that are already well-connected – you can grow a generic network that is scale-free.

In the L-systems example above, edges were represented as channels. Channels in occam-π are directed, and are used directly in the L-systems network to match the left-to-right ordering of symbols. In a scale-free network, edges may be undirecte with no natural ordering . We modelled this by representing edges as processes with separate channels connecting them to the nodes on each side; an example is shown in Figure 5. This is a more flexible model, since there is no need for an explicit ordering, and edges may have their own behaviours if necessary. For example, if an edge needs to be reconnected between a different pair of nodes, it can take part in the decision and reconnection process itself.

To implement a growing scale-free network with preferential attachment, we start by creating two node processes and linking them by an edge processes. Next a controller process iteratively forks a new node process and connects it to a pre-defined number of new edge processes. For each new edge process, a randomly selected pre-existing node is selected and connected to the edge process. This random selection is biased towards highly connected nodes, thus implementing preferential attachment.

In the same way that we can apply a continuous space model to both boids and ant-based annular sorting, we can easily adapt the scale-free network with preferential attachment model to implement a small-world network (Watts and Strogatz, 1998) instead. This reuses the same node and edge processes, but changes the way they are connected together.

Space: the Results

Our space models produced several useful, reusable components. Both space models were successfully applied to more than one complex system example with minimal work. In addition, we have identified a number of initial design patterns, which we can can categorise into four groups: *modelling*, *implementation*, *optimisation* and *refactoring* patterns. Patterns we identified included:

Distributed Continuous Space (*modelling*): by dividing continuous space into regions, we can efficiently implement local vision in a distributed simulation.

Agent Process (*modelling*): agents are modelled as concurrent processes that interact within a space. The space may be modelled explicitly using additional processes, or may be implicit in the relationships between the agents.

Factory Process (*implementation*): factory processes spawn new processes at runtime in response to requests from other processes. They provide a common *context* for the newly-created processes, and hide details of creating, configuring and connecting up new processes behind an interface. (This is the process-oriented equivalent of the *abstract factory* pattern (Gamma et al., 1995).)

Ghost Location (*optimisation*): when refactoring a simulation to run in a distributed manner, a ghost process can cache the contents of a remote location to avoid repeated network communication. (This is an application of the existing *remote proxy* pattern.)

Agent Migration (*optimisation*): in a distributed simulation, an agent can be suspended and moved to a different host, in order to minimise the number of network communications it must do.

Reification (*refactoring*): creating a process (a "thing") to represent a relationship between two other processes. For example, a directed link between two processes can simply be a channel, but an undirected or buffered link can be better modelled as a process.

In addition, we found possible patterns related to visualisation and the modelling of time, which we are investigating.

Conclusion

In this paper we have outlined CoSMoS, a planned modelling and simulation infrastructure for the investigation of generic complex systems. We are using the process-oriented programming model, owing to the natural analogies between processes and complex system agents and the ability to construct massively-concurrent and distributed simulations. CoSMoS will promote reusable modelling techniques through the development of pattern languages.

We have studied the modelling and simulation of space in complex systems in the context of reusable modelling techniques. We have shown how two very different spaces, a geometric continuous space and a arbitrary network space, can be modelled and simulated in a reusable way, and have identified a number of design and refactoring patterns.

The next step for CoSMoS will be to start modelling and simulating some more detailed complex systems based on real-world observations and data. This will help identify further generic complex system components, and aid the development and validation of our method and toolset.

Acknowledgements

This work is part of the CoSMoS project, funded by EPSRC grant EP/E053505/1 and a Microsoft Research Europe PhD studentship.

References

Alexander, C., Ishikawa, S., Silverstein, M., Jacobson, M., Fiksdahl-King, I., and Angel, S. (1977). *A Pattern Language*. Oxford University Press.

Amos, M. and Don, O. (2007). An ant-based algorithm for annular sorting. In *Proceedings of the 2007 IEEE Congress on Evolutionary Computation (CEC)*, pages 142–148. IEEE Press.

Barabasi, A., Albert, R., and Jeong, H. (2000). Scale-free characteristics of random networks: the topology of the world-wide web. *Physica A*, 281:69–77.

Barabasi, A. and Bonabeau, E. (2003). Scale-free networks. *Scientific American*, 288:60–69.

Barnes, F. R. M., Welch, P. H., and Sampson, A. T. (2005). Barrier synchronisation for occam-pi. In *2005 International Conference on Parallel and Distributed Processing Techniques and Applications (PDPTA)*, pages 173–179. CSREA Press.

Beck, K. (1997). *Smalltalk Best Practice Patterns*. Prentice Hall.

Brown, W. J., Malveau, R. C., McCormick III, H. W., and Mowbray, T. J. (1998). *AntiPatterns: refactoring software, architectures, and projects in crisis*. Wiley.

Fowler, M. (1997). *Analysis Patterns: reusable object models*. Addison Wesley.

Gamma, E., Helm, R., Johnson, R., and Vlissides, J. (1995). *Design Patterns: Elements of Reusable Object-Oriented Software*. Addison-Wesley.

Garber, D. (1995). Leibniz: Physics and philosophy. In Jolley, D., editor, *The Cambridge Companion to Leibniz*. Cambridge University Press.

Giavotto, J. and Michel, O. (2002). Data structures as topological spaces. In *3rd International Conference on Unconventional Models of Computation (UMC02)*, pages 137–150. Springer.

Hart, E. and Ross, P. (2004). Studies on the implications of shape-space models for idiotypic networks. In *3rd international Conference on Artificial Immune Systems (ICARIS04)*, pages 413–426. Springer.

Hoare, C. A. R. (1985). *Communicating Sequential Processes*. Prentice Hall.

Martin, J. M. R. and Welch, P. H. (1997). A Design Strategy for Deadlock-Free Concurrent Systems. *Transputer Communications*, 3(4).

Milner, R. (1999). *Communicating and Mobile Systems: The π-calculus*. Cambridge University Press.

Perelson, A. S. and Oster, G. F. (1979). Theoretical studies of clonal selection: Minimal antibody repertoire size and reliability of self–non-self discrimination. *Journal of Theoretical Biology*, 81(4):645–670.

Phillips, A. and Cardelli, L. (2007). Efficient, correct simulation of biological processes in the stochastic pi-calculus. In *Computational Methods in Systems Biology (CMSB07)*, pages 184–199. Springer.

Prusinkiewicz, P. and Lindenmayer, A. (1990). *The Algorithmic Beauty of Plants*. Springer-Verlag.

Reynolds, C. W. (1987). Flocks, herds, and schools: A distributed behavioral model. In *14th Annual Conference on Computer Graphics and Interactive Technologies (SIGGRAPH87)*, pages 25–34. ACM.

Ritson, C. G. and Welch, P. H. (2007). A process-oriented architecture for complex system modelling. In *Communicating Process Architectures 2007*, pages 249–266. IOS Press.

Roth, J. (2002). Patterns of mobile interaction. *Personal Ubiquitous Computing*, 6(4):282–289.

Sampson, A. T., Welch, P. H., and Barnes, F. R. M. (2005). Lazy Cellular Automata with Communicating Processes. In *Communicating Process Architectures 2005*, pages 165–175. IOS Press.

Schweigler, M. (2006). *A Unified Model for Inter- and Intra-processor Concurrency*. PhD thesis, Computing Laboratory, University of Kent, Canterbury, UK.

Watts, D. J. and Strogatz, S. H. (1998). Collective dynamics of 'small-world' networks. *Nature*, 393:440–442.

Welch, P. H., Barnes, F. R. M., and Polack, F. A. C. (2006). Communicating complex systems. In *11th IEEE International Conference on Engineering of Complex Computer Systems (ICECCS06)*. IEEE Press.

Wiles, J., Geard, N., Watson, J., Willadsen, K., Mattick, J., Bradlet, D., and Hallinan, J. (2005). There's more to a model than code: understanding and formalizing *in silico* modeling experience. In *2005 Workshops on Genetic and Evolutionary Computation (GECCO)*, pages 281–288. ACM.

On Preferred States of Agents: how Global Structure is reflected in Local Structure

Tom Anthony[1], Daniel Polani[1,2] and Chrystopher L. Nehaniv[1,2]

[1]Adaptive Systems Research Group
[2]Algorithms Research Group
School of Computer Science, University of Hertfordshire
College Lane, Hatfield, Herts, AL10 9AB, UK
{T.C.Anthony,D.Polani,C.L.Nehaniv}@herts.ac.uk

Abstract

We investigate the correlation between the information theoretic measure of *empowerment* and the graph theoretic measure of *closeness centrality*, to better understand the structural conditions that must exist in a world for learning and adaptation. We examine both measures in both a simple grid-world scenario, represented as a graph, and on a scale-free graph. We show a strong correlation between the two measures, and discuss the strengths and weaknesses of both. We go on to show how the local measurement of empowerment can in many cases predict a measure for the global measurement of closeness centrality.

Motivation

"Nature uses only the longest threads to weave her patterns, so that each small piece of her fabric reveals the organization of the entire tapestry." - Richard Feynman

Learning and adaptation are central themes to artificial life, and it is our hypothesis that a better understanding of the conditions that must exist to make learning, adaptation and evolution possible will help to guide future research. It is plausible to assume that an arbitrary or random world would be extremely difficult, if at all possible, to learn. We know there is significant structure in the world, and believe that learning takes advantage of this structure. In this paper we begin to investigate what conditions, embedded within a world through some underlying structure, are necessary for certain types of adaptation problems.

It has been hypothesised that embodied agents receive an adaptive and evolutionary advantage by optimising their sensoric and neural configurations for their environment. Specific attention has been paid to processing and optimising of Shannon-type information they receive from their environment (Attneave, 1954; Barlow, 1959, 2001; Atick, 1992). Similar work includes the concept of *homeokinesis*, proposed by Der et al. (1999), where a homeokinetic system, or agent, learns to improve the predictive capabilities of its future perceptions.

A specific flavour of this view suggests that such informational predictive principles could provide organisms/agents with intrinsic motivation. Examples include that by Prokopenko et al. (2006), Bialek et al. (2001) and Ay et al. (2008), which use similar approaches based on *excess entropy / predictive information*.

In this paper we have chosen to use *empowerment* (Klyubin et al., 2005b,a), an information theoretic measure for the efficiency of a *perception-action loop*. Essentially empowerment uses the channel capacity for the external component of a perception-action loop to identify areas that are advantageous for an agent embodied within an environment.

It assumes situations with a high efficiency of the perception-action loop should be favoured by an agent. Based entirely on the sensors and actuators of an agent, empowerment intrinsically encapsulates an evolutionary perspective; namely that evolution has selected which sensors and actuators a successful agent should have, which in turn suggests which states should be visited.

This hypothesis was tested in a variety of different scenarios (Klyubin et al., 2005b,a; Capdepuy et al., 2007), and notwithstanding the quite different scenarios it coincided surprisingly well with an intuitive understanding of favourable behaviours or of natural solutions to particular challenges of adaptation. Furthermore, it correlated well against measures that had been hand crafted to evaluate certain scenarios.

Notwithstanding the successful performance, we do not currently have a strong understanding of why this may be. What are the properties of the world that make empowerment such a universal measure? Why should it work at all? These are the questions we are going to begin to study in this paper.

Locating Structure

We hypothesise that an agent that optimises its sensorimotor apparatus improves its ability to detect the underlying structure of the world, and that this is an important aspect of such optimisation. We further hypothesise that a better understanding of this structure would improve such optimisation, and thus allow for better adaptation and learning.

To investigate this we set out to start identifying the basic

properties of the world, and how they are detected by empowerment. We selected to go about this by investigating a representation for an environment that manifests its structure in an easily observable manner, is well understood, and has established methods for measuring preferable states.

We chose to represent the state space using graphs, which fit all these criteria; they are well understood through graph theory and social network analysis, and they have accessible methods for identifying certain aspects of their structure. As a measure to identify preferred states we chose to use *centrality*, a measure of a node's importance from graph theory, which is a well established method (Wasserman and Faust, 1994). There are varying measures for centrality; in this paper we use *closeness centrality*, which most closely corresponds with the spirit of empowerment.

Most stationary worlds, containing an embodied agent, can be viewed of as the current state of the world connected to neighbouring states by the actions the agent would need to take to arrive at them; this can be modelled as a graph. This same representation of the world was used by Şimşek and Barto (2007) in investigating skill development among agents.

We can now analyse empowerment, and some aspects of what it captures about the world, by comparing it with centrality measurements in the same scenarios.

Quantifying Preference

Empowerment, a local measure, quantifies the changes that an embodied agent can make on its environment, and observe the effects of, in a given time period. Here we reduced ourselves to a simple representation of the world which is entirely deterministic, creating a special case for empowerment. However, it can work in both entirely deterministic and probabilistic environments, which may even be non-stationary (Capdepuy et al., 2007).

The closeness centrality of a node in a graph is calculated by adding the distance of the shortest paths from that node to every other node in the network, and then inverting this value so that a shorter total path to all other nodes has a higher value. To calculate the closeness centrality of a signal node requires viewing the whole graph; it is a global measure. Klyubin et al. (2005a) showed an example where a similar measure, the average shortest distance in a maze, correlated well with empowerment.

We will examine two scenarios, and will employ both empowerment and closeness centrality in each for identifying and measuring states that an embodied agent would find 'interesting' or 'preferential' to be in. When we use the word 'state' we refer to the state of the whole system, including both the environment and the agent.

Information Theory

The notion of empowerment is based on information theory, introduced by Shannon (1948). To introduce this, the first important measure is *entropy*, which is a measure of uncertainty:

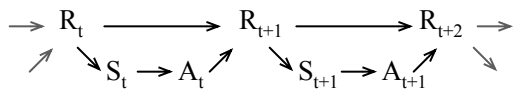


Figure 1: Bayesian network representation of the perception-action loop.

$$H(X) = -\sum p(x) \log p(x). \quad (1)$$

Where X is a discrete random variable with values $x \in X$ and $p(x)$ is the probability mass function such that $p(x) = Pr\{X = x\}$. The logarithm can be taken to any chosen base; in our paper we consistently use 2, and accordingly the units of measurement are then called *bits*. If Y is another random variable jointly distributed with X the *conditional entropy* is:

$$H(Y|X) = -\sum_x p(x) \sum_y p(y|x) \log p(y|x). \quad (2)$$

This measures the remaining uncertainty about the value of Y, if we know the value of X. Finally, this also allows us to measure the *mutual information* between to random variables:

$$I(X;Y) = H(Y) - H(Y|X). \quad (3)$$

Mutual information can be thought of as the reduction in uncertainty about the variable X or Y, given that we know the value of the other. The mutual information is symmetric, so we could also use $I(X;Y) = H(X) - H(X|Y)$ (Cover and Thomas, 1991).

Empowerment

Empowerment is based on the information theoretic perception-action loop formalism introduced by Klyubin et al. (2005a, 2004), as a way to model embodied agents and their environments. The model views the world as a communication channel; when the agent performs an action, it is injecting Shannon information into the environment, which may or may not be modified, and subsequently the agent reacquires part of this information from the environment via its sensors.

In Fig.1 we can see the perception-action loop represented by a Bayesian network, where the random variable R_t represents the state of the environment, S_t the state of the sensors, and A_t the actuation selected by the agent at time t. It can be seen that R_{t+1} depends only on the state of the environment at time t, and the action just carried out by the agent.

By modelling this as a communication channel, we can employ information-theoretic methods, which are the basis

for empowerment. First, we must introduce channel capacity (Shannon, 1948; Cover and Thomas, 1991) for a discrete memoryless channel:

$$C(p(y|x)) = \max_{p(x)} I(X;Y). \quad (4)$$

The random variable X represents the distribution of messages being sent over the channel, and Y the distribution of received signals. Clearly, the higher the mutual information between the two variables, the higher the capacity of the channel. The channel capacity is measured as the maximum mutual information taken over all possible input distributions, $p(x)$, and depends only on $p(y|x)$, which is fixed. One algorithm that can be used to find this maximum is the iterative Blahut-Arimoto algorithm (Blahut, 1972).

Empowerment can be intuitively thought of as a measure of how many observable adjustments an embodied agent can make to his environment, either immediately, or in the case of n-step empowerment, over a given period of time. An alternative way to view empowerment is that it guides agents to places in the world where they get the most benefit from their sensors and actuators. Using the above perception-action loop formalism and the Blahut-Arimoto algorithm, this can be directly quantified. We remind the reader that sensors and actuators implicitly encode evolutionary knowledge of the type of information to perceive and 'create'.

In the case of n-step empowerment, we first construct a compound random variable of the last n actuations, labelled A_t^n. We now need to maximise the mutual information between this variable and the sensor readings at time $t+n$, represented by S_{t+n}. Here we consider empowerment as the channel capacity between these:

$$\mathfrak{E} = C(p(s_{t+n}|a_t^n)) = \max_{p(a_t^n)} I(A_t^n; S_{t+n}). \quad (5)$$

An agent that maximises its empowerment will position itself in the environment in a way as to maximise its options for influencing its relationship with the environment (Klyubin et al., 2005a).

Note that in this paper we are use empowerment in an exclusively deterministic scenario, within a discrete world, but that empowerment is defined in full generality for non-deterministic probabilistic environments and does not assume perfect information.

In this paper we can use a shorthand method for calculating empowerment; we are able to do this for several reasons. All the scenarios we examine are deterministic and feature no non-stationary elements, and so do not require the probabilistic elements of empowerment. Additionally, as they are all represented as a graph, we are able to further simplify the formula. We can calculate n-step empowerment for a node v_i on the graph thus:

$$\mathfrak{E}_n(v_i) = \log \left[\sum_{\substack{j=1 \\ d(v_i,v_j)\leq n}}^{g} 1 \right] \quad (6)$$

Where $d(v_i, v_j)$ is the geodesic distance between the nodes v_i and v_j. Note that this is a shorthand method we are able to use as we have complete knowledge of the scenarios and the representation; Eq. (5) reduces to Eq. (6), and would work identically in the same scenarios, using the perception-action loop formalism.

Closeness Centrality

Graph Theory and Network Analysis have long had a requirement for identifying important nodes in a graph (Wasserman and Faust, 1994). The simplest methods for this have been to count the edges leaving or entering a node, known as outdegree and indegree respectively. This is very simplistic and is normally inadequate for complex graphs. Therefore, the primary method for measuring node importance is a group of various measures collectively known as centrality. There have been several methods of centrality suggested over time, but one of the most popular is closeness centrality, which can be presented in various ways. As mentioned in Wasserman and Faust (1994), and reviewed by Freeman (1979), the simplest formula for closeness centrality is that suggested by Sabidussi (1966):

$$C_C(v_i) = \left[\sum_{\substack{j=1 \\ j\neq i}}^{g} d(v_i, v_j) \right]^{-1}. \quad (7)$$

For a given node v_i, in a graph with g nodes, this gives a measurement of the sum of the shortest paths to all other nodes, which is then inverted to give a higher centrality to those with shorter total paths to the rest of the graph. Intuitively, this can be closely linked to the average distance from all other cells that empowerment was anti-correlated with, from the maze scenario used in Klyubin et al. (2005a).

To calculate the closeness centrality on the graphs encountered throughout this paper, we used the network analysis software Pajek (Batagelj and Mrvar, 1998). Pajek uses a modified version of closeness centrality, suggested in Beauchamp (1965):

$$C'_C(v_i) = \frac{(g-1)}{\left[\sum_{j=1}^{g} d(v_i, v_j)\right]} = (g-1)C_C(v_i). \quad (8)$$

This formula is used simply to normalise the closeness centrality figures to the graphs size in order to allow comparison of the figures between graphs of different sizes.

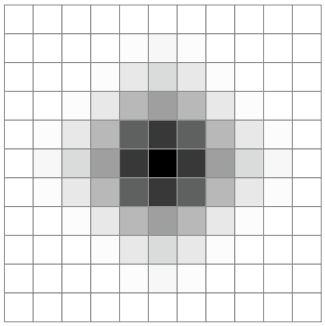


Figure 2: View of the empowerment distribution for the gridworld scenario, with the box positioned at the center. A darker shade means higher empowerment. Empowerment scales from 5.92 to 7.79 bits.

Scenarios

In order to compare these two measurements we apply them to the same two agent scenarios to identify the correlation between them, and any areas of disparity. In order to construct the first scenario, it is necessary to observe that most state spaces encompassing an agent in a stationary world can be naturally represented as a graph of nodes, with transitions leading between them corresponding to the actions of an agent within that world.

Box Pushing

Consider the box pushing scenario from Klyubin et al. (2005a) as a graph. The scenario consists of a gridworld of infinite size, within which there exists an agent and a box, each of which occupy a single cell. The box is visible to the agent; his view of the world consists of his position and the position of the box. The agent has 5 actions available to it at any time; it can stand still, or move to one of the four neighbouring cells. If the agent moves into a cell that is occupied by the box then the box is pushed, in the same direction, into the adjacent cell.

In Klyubin et al. (2005a) it was shown that for any n-step empowerment, the agent prefers being near the box, which gives it more influence on the state of the world. It most 'enjoyed' beginning on top of the box, where moving in and of the 4 directions would allow it to fall down next to the box, from where it could start pushing it like normal; this could be used as a starting position but was a position impossible for it to return to.

In translating this world into a graph representation, we needed to limit our originally infinite world to a finite graph. We investigate the influence of this finiteness by examining the growth of centrality. We show that beyond a certain horizon it can be seen that the centrality increases in a continuous fashion and that the centrality for the nodes represented in previous approximations grows proportionately. Whilst we do not offer a proof of this fact, in Fig.3 we demonstrate

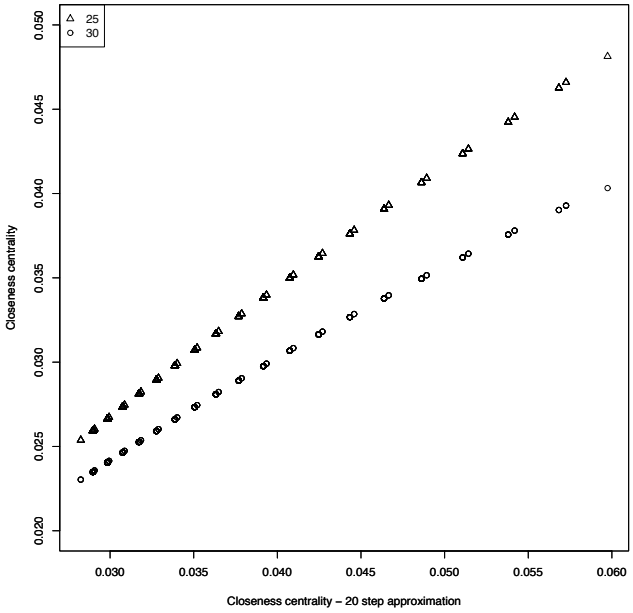


Figure 3: Correlation of closeness centrality for 25-step and 30-step graph approximations against a 20-step approximation.

the point by showing the correlation between graph representations of increasing diameters.

Results

Klyubin et al. (2005a) had previously shown how empowerment worked in the box pushing gridworld experiment, and so it made for a good environment in which to run our initial experiments. We generated a unweighted directed graph to represent the world. Note that we are using a non-classical view of graphs; rather than viewing them as comprised of units, with connecting links between them, we are viewing each node as a possible state of the world, including the agent itself, (of which, only one can be the real state at any moment) and the edges as transitions between these states.

To do this, we initialised the world with the box in the center, and the agent standing upon the box, as described earlier. We then let the agent run through every possible trajectory of 30 actuations, generating a graph of states and actions; the final graph had 419,121 nodes. Using Pajek, we calculated the closeness centrality for all nodes in the graph.

We next measured empowerment for every state with the box positioned in the center of the world, and the agent positioned at each location that it could reach within 30 timesteps from the center. This was sufficient as the dynamics of the world comes from the agent's initial position relative to the box, and thus moving the box was unnecessary. Our empowerment measurements were run to measure 3-step, 5-step and 7-step empowerment.

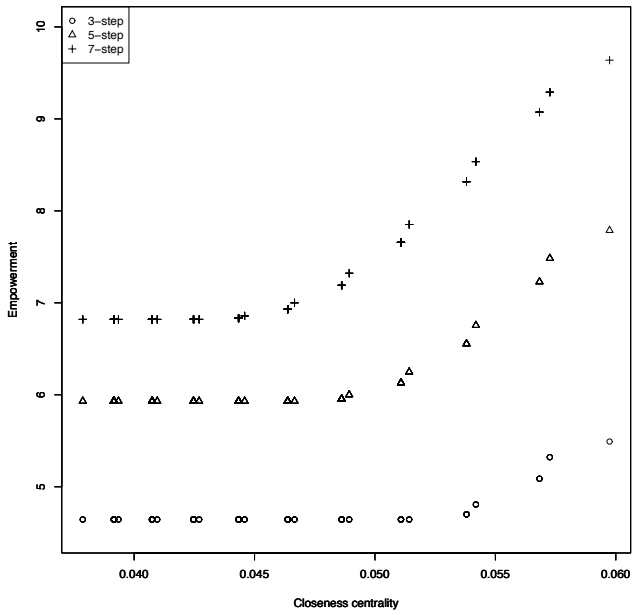


Figure 4: Correlation plot between Empowerment and Closeness Centrality. The horizon effect of empowerment can be seen clearly.

In order to correlate empowerment and centrality, we collated the results, removing the centrality results for nodes where the box was not positioned in the center of the world; this gave us a state for state comparison of each measure against the other for different initial positions of the agent.

We additionally ran the same experiment for graphs produced for both 20 and 25 timesteps, to identify the influence of representing the infinite gridworld as a finite graph did not skew the results. We found that the correlation of centrality for the overlapping nodes of these varying size graphs indicates a close to linear relationship and finite graphs work as a good approximation.

Note that closeness centrality is a global property, calculated it for any given node requires seeing all other nodes in the graph, while empowerment is local and looks only at neighbouring nodes within a given distance.

Local Structure

As hypothesised, we found a very strong correlation between the closeness centrality and empowerment, which can be seen in Fig.4. The graph shows clearly the horizon effect of empowerment; it can be seen to be constant whenever the box is outside of the agent's reach. For n-step value with larger values of n the horizon can be seen to extend further from the box. Once the box is within it's reach, according to n, the empowerment grows as the agent increases its influence over the world by getting closer to the box.

The horizon effect emphasises that empowerment is a local measure; it cannot see the whole world. However, when the agent is within an area where it can improve it's ability to manipulate the state of the world, this local measure correlates with the global measure of the world given by closeness centrality.

This highlights that in an infinite, or an unexplored, world where centrality cannot be employed, empowerment provides a measure that can be used. Whilst empowerment is limited by the horizon effect, exploring the world (which would be necessary to use closeness centrality) would allow our agent to also overcome the horizon.

In addition, this correlation also confirms our hypothesis that empowerment, within its horizon, does see global aspects of a system at a local level within this world. What structure or prerequisites that must exist for this effect to take place are yet to be determined.

It is important to note that the results from empowerment can be computed by the formula in Eq. (6), or equally by that in Eq. (5), without modelling the world as a graph at all.

Scale-free Graphs

The second scenario uses scale-free networks (graphs); a very important subclass of graphs, in which there are a few nodes with a high degree, and most nodes have a far lower degree. Their typical structure is independent of the graph's size; with fewer or more nodes, the graph would still exhibit similar properties. The exact distribution of edges per node follows a power law distribution (Barabasi and Albert, 1999):

$$P(k) \sim k^{-\gamma}. \tag{9}$$

Here $P(k)$ is the probability that a node connects with k other nodes, and decreases exponentially according to the coefficient γ.

As discussed in Barabasi (2003), scale-free graphs can be seen in many real world situations, including protein interaction networks (Jeong et al., 2001), social networks, and even the world wide web (Barabasi and Albert, 1999).

We hypothesise that the scale-free property of graphs can work to synthesise an underlying structure that may be found in real world task spaces, and can be used as a good platform for initial investigation of such structure.

Results

Using preferential attachment algorithm introduced by Barabasi and Albert (1999) we constructed a scale-free undirected graph with 400,000 nodes to run our measures on. Our graph was built using an initial complete graph of 3 nodes, and adding additional nodes one at a time. Each new node would create 3 new edges connected to 3 different nodes on the existing graph, chosen using a probability according to their current degree.

For all nodes in the graph we calculate both the n-step empowerment (for a range of values of n) and the closeness

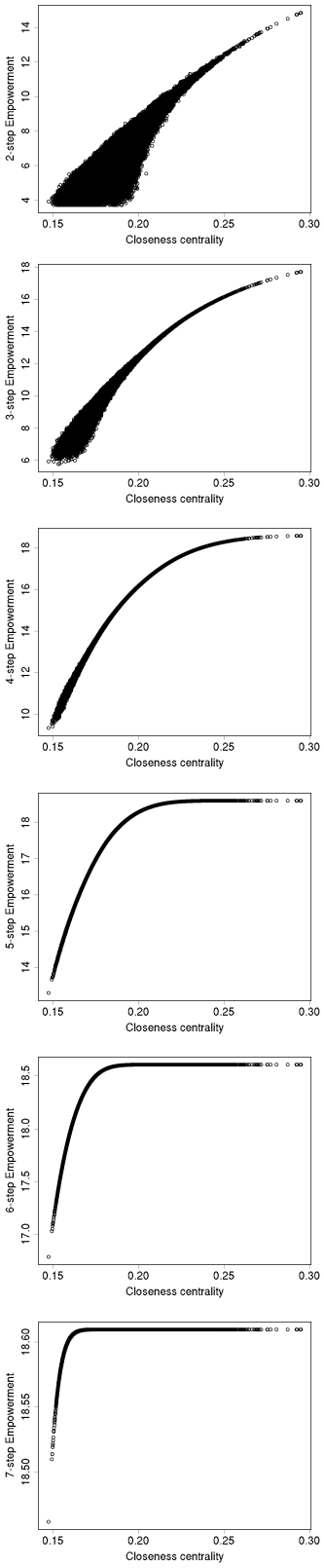


Figure 5: Correlation between closeness centrality and 2-step to 7-step empowerment.

centrality. To calculate the closeness centrality, we again use the Pajek analysis software.

Our results here corroborate those from our first experiment with regard to the correlation between closeness centrality and empowerment. Here, we see the inverse of the horizon effect; given too much time, empowerment can reach any part of the graph (analogous to being able to do anything within a world) and assigns almost all nodes equal value. This is an interesting point for empowerment; given too high of a 'budget', where an agent can do everything possible within the world (or reach every node in a graph) then it does not differentiate between them. This is the type of world we would describe as 'boring'; one where and agent can do anything it wants from any position of the world.

Again though, empowerment sees at a local level aspects of the global property of the world. In this scenario, this is maybe not surprising given the nature of a scale-free graph; but it is important to see that empowerment was not told anything of the structure of the world, and that still this fact comes through.

In Fig.5 we show the correlation between closeness centrality and n-step empowerment for n=2 to n=7. Note that even 2-step empowerment has a strong correlation at the higher centrality nodes, and 3-step even more so. As n increases it can be seen that the small-world property of the graph results in an empowerment ceiling being reached which results in a reduced correlation for high centrality nodes.

Discussion

Both of our experiments highlight the strong correlation between empowerment and closeness centrality, and that even n-step empowerment with a low value for n will normally serve a a strong predictor for centrality. This is significant given that individual node centrality is a global property of a graph, but we can use a local measure to give similar relative values to nodes. Note that empowerment doesn't see any more than centrality, but in the 'interesting' parts of the world it does see, the two measures agree.

In both scenarios the correlation is strong provided that the n chosen for n-step empowerment is suitable. We believe a simple method for overcoming this in an unknown world is for an agent to select the lowest n value possible; if the horizon of this n does not allow the agent to observe any degrees of freedom it can then increase n incrementally to overcome this (or embark on a random exploration).

With empowerment, selection of a suitable n is interesting in another regard; a low value of n can mean encountering the horizon effect, and possibly not seeing 'interesting' parts of the world, whilst a too high value of n can result in the agent being able to do anything and not needing to distinguish between different states. The result of this is a particular world having an n value with the correct balance

between these two effects, which we hypothesise may reflect one aspect of the underlying structure which is important for learning and adaptation.

Closeness centrality is limited to deterministic task spaces that can be completely represented by either a directed or undirected graph, which constricts the space of problems it can be used to measure. In the space of problems in which both measures can be used, these results indicate not only empowerment correlates well with centrality, but it does so without complete knowledge of the world. Furthermore, it can work in non-deterministic, non-stationary, environments which cannot be represented as a graph, including infinite worlds.

A comparison could have been drawn between the global measure of closeness centrality, and some local version of centrality that worked on a local subset of the graph, and we expect a similar correlation would have been found. However, any such localised version of centrality would suffer from many of the same restrictions that centrality does compared to empowerment. We studied empowerment specifically as part of a much more general picture which includes an evolutionary aspect and which in addition will allow us to extend the research into non-deterministic environments in future work. Essentially we are using centrality as a 'sanity check' that empowerment does something sensible in these scenarios.

Overall, we believe that these results show a strong indication of certain global aspects of various worlds being 'coded' at a local level, and an appropriate sensory configuration can not only detect this information, but can also use it. Such uses could include learning and adaptation, and uses for evolution between generations. There are indications that understanding which aspects of global structure are visible at a local level would allow improved adaptation and learning for agents embodied within the corresponding world.

Future work

Vergassola et al. (2007) drew a parallel between the behaviour of biological organisms and search methods that use local informational cues to draw conclusions about the global structure of the world. It is our belief that further study of this area will allow us to not only draw further parallels with the learning and adaptation methods employed by biological organisms, but will also allow a better understanding of these processes leading to improved methods.

Further work needs to be done to extend these results into other worlds and task spaces, and to better understand in which scenarios they hold true. This should include worlds with various elements providing opportunities for agents to manipulate their environment, and even non-stationary worlds.

Attention needs to be paid to how to choose an initial strategy when presented with a completely unknown task space (such as choosing an initial n for empowerment) and conversely, how much of this information is embedded with an agent or organisms embodiment.

Acknowledgements

The authors would like to thank the reviewers for their feedback and suggestions which helped bring about various improvements to the paper.

References

Atick, J. J. (1992). Could information theory provide an ecological theory of sensory processing. *Network: Computation in Neural Systems*, 3(2):213–251.

Attneave, F. (1954). Some informational aspects of visual perception. *Psychological Review*, 61(3):183–193.

Ay, N., Bertschinger, N., R. Der, F. G., and Olbrich, E. (2008). Predictive information and explorative behavior of autonomous robots. *European Physical Journal B*. (Accepted).

Barabasi, A.-L. (2003). *Linked: How Everything Is Connected to Everything Else and What It Means for Business, Science, and Everyday Life*. Plume Books.

Barabasi, A. L. and Albert, R. (1999). Emergence of scaling in random networks. *Science*, 286(5439):509–512.

Barlow, H. B. (1959). Possible principles underlying the transformations of sensory messages. In Rosenblith, W. A., editor, *Sensory Communication: Contributions to the Symposium on Principles of Sensory Communication*, pages 217–234. The M.I.T. Press.

Barlow, H. B. (2001). Redundancy reduction revisited. *Network: Computation in Neural Systems*, 12(3):241–253.

Batagelj, V. and Mrvar, A. (1998). Pajek – program for large network analysis.

Beauchamp, M. A. (1965). An improved index of centrality. *Behavioral Science*, 2:161–163.

Bialek, W., Nemenman, I., and Tishby, N. (2001). Predictability, complexity, and learning. *Neural Comp.*, 13(11):2409–2463.

Blahut, R. (1972). Computation of channel capacity and rate distortion functions. *IEEE Transactions on Information Theory*, 18(4):460–473.

Capdepuy, P., Polani, D., and Nehaniv, C. L. (2007). Maximization of potential information flow as a universal utility for collective behaviour. In *Proceedings of the First IEEE Symposium on Artificial Life*.

Cover, T. M. and Thomas, J. A. (1991). *Elements of information theory*. Wiley-Interscience, New York, NY, USA.

Der, R., Steinmetz, U., and Pasemann, F. (1999). Homeokinesis - a new principle to back up evolution with learning. In Mohammadian, M., editor, *Computational Intelligence for Modelling, Control, and Automation*, volume 55 of *Concurrent Systems Engineering Series*, pages 43–47. IOS Press.

Freeman, L. C. (1979). Centrality in social networks: Conceptual clarification. *Social Networks*, 1(3):215–239.

Jeong, H., Mason, S. P., Barabasi, A. L., and Oltvai, Z. N. (2001). Lethality and centrality in protein networks. *Nature*, 411(6833):41–42.

Klyubin, A. S., Polani, D., and Nehaniv, C. L. (2004). Organization of the information flow in the perception-action loop of evolved agents. In Zebulum, R. S., Gwaltney, D., Hornby, G., Keymeulen, D., Lohn, J., and Stoica, A., editors, *Proceedings of 2004 NASA/DoD Conference on Evolvable Hardware*, pages 177–180. IEEE Computer Society.

Klyubin, A. S., Polani, D., and Nehaniv, C. L. (2005a). All else being equal be empowered. In Capcarrère, M. S., Freitas, A. A., Bentley, P. J., Johnson, C. G., and Timmis, J., editors, *Advances in Artificial Life: Proceedings of the 8th European Conference on Artificial Life*, volume 3630 of *Lecture Notes in Artificial Intelligence*, pages 744–753. Springer.

Klyubin, A. S., Polani, D., and Nehaniv, C. L. (2005b). Empowerment: A universal agent-centric measure of control. In *Proceedings of the 2005 IEEE Congress on Evolutionary Computation*, volume 1, pages 128–135. IEEE Press.

Prokopenko, M., Gerasimov, V., and Tanev, I. (2006). Evolving spatiotemporal coordination in a modular robotic system. In Nolfi, S., Baldassarre, G., Calabretta, R., Hallam, J., Marocco, D., Meyer, J.-A., and Parisi, D., editors, *From Animals to Animats 9: 9th International Conference on the Simulation of Adaptive Behavior (SAB 2006), Rome, Italy, September 25-29 2006*, volume 4095 of *Lecture Notes in Computer Science*, pages 558–569. Springer.

Sabidussi, G. (1966). The centrality index of a graph. *Psychometrika*, 31(4):581–603.

Shannon, C. E. (1948). A mathematical theory of communication. *Bell System Technical Journal*, 27:379–423.

Vergassola, M., Villermaux, E., and Shraiman, B. I. (2007). 'infotaxis' as a strategy for searching without gradients. *Nature*, 445(7126):406–409.

Wasserman, S. and Faust, K. (1994). *Social Network Analysis*. Cambridge University Press.

Şimşek, Ö . and Barto, A. (2007). Betweenness centrality as a basis for forming skills. Technical report, University of Massachusetts, Department of Computer Science.

Information-Theoretic Aspects of Control in a Bio-Hybrid Robot Device

Stefan Artmann[1], Soichiro Tsuda[2] and Klaus-Peter Zauner[2]

[1]Frege Centre for Structural Sciences, Friedrich-Schiller-University,
Zwätzengasse 9, 07737 Jena, Germany
[2]School of Electronics and Computer Science
University of Southampton, SO17 1BJ, United Kingdom
stefan.artmann@uni-jena.de

Abstract

Information processing in natural systems radically differs from current information technology. This difference is particularly apparent in the area of robotics, where both organisms and artificial devices face a similar challenge: the need to act in real time in a complex environment and to do so with computing resources severely limited by their size and power consumption. The formidable gap between artificial and natural systems in terms of information processing capability motivates research into the biological modes of information processing. Such undertakings, however, are hampered by the fact that nature directly exploits the manifold physical characteristics of its computing substrates, while available theoretical tools in general ignore the underlying implementation. Here we sketch the concept of bounded computability in an attempt towards reconciling the information-theoretic perspective with the need to take the material basis of information processing into account. We do so in the context of *Physarum polycephalum* as a naturally evolved information processor and the use of this organism as an integral component of a robot controller.

Introduction

Technological progress makes ontological distinctions between classes of entities, like those between the natural and the artificial, or between the living and the non-living, more and more porous. Unconventional computing devices contribute to this process. Hybrid artifacts, for example, try to overcome the theoretic and physical limits of information processing in solid-state realisations of digital von Neumann machines by exploiting the self-organisation of naturally evolved systems in engineered environments (Zauner, 2005). Biological systems evolved enviable computing capabilities to cope with noisy and harsh environments and compete with rivalling life forms. Information processing in biological systems, from single-cell organisms to brains, directly utilises the physical and chemical processes of cellular and intracellular dynamics. Arguably, therefore, if one aims at narrowing the still formidable performance gap between artificial and biological systems, the material basis of their information processing cannot be ignored. An information-theoretic analysis of hybrid computational systems must, hence, take the physical properties of material substrates used for computation into account. By 'information theory' is meant here, not only Shannon's statistical theory of communication (Shannon and Weaver, 1949), but the structural science that constructs mathematical models of the form, meaning, and use of information and applies them to empirical phenomena. The empirical diversification of information theory would allow the engineering of unconventional computers to utilise empirical knowledge of naturally evolved systems more efficiently since the requirements for particular computational tasks could then be stated directly in terms of physical specifications of computational media (Tsuda et al., 2006a).

To explore the border zone between information theory and the physics of self-organising systems, it is necessary to elaborate a *theory of bounded computability* that relates generic traits of information-processing systems, not with general time and space bounds (as in the theory of computational complexity (Papadimitriou, 1994)), but with specific physico-chemical constraints on the realisation of such systems in different classes of computational media (analogously to the theory of bounded rationality (Simon, 1997)). In a theory about possible relations between material media and computational functions of physical information-processing systems, the concept of information must integrate the distinction between the behavioural structure of a system, its functional structure, and the structure of its material medium. Otherwise, the complex structural interplay of matter, function, and behaviour that constitutes the very nature of information, could not be adequately analysed and the concepts of information theory would add nothing new to physics and chemistry. For example, the more medium and function are considered being inseparable, the less it is reasonable to use information theory and its fundamental distinction between form and 'in-formed' media. Not only must one and the same information be regarded as representable by different physical entities; various material substrates must also be considered being possible media for the same information-processing function.

If computers will develop along an increasing number of technological ramifications, a theory of bounded com-

putability must become more and more empirically diversified. On the other hand, the theory has also to define its basic concepts more and more generally to let a unified approach to the analysis and construction of computers appear still promising. This paper first introduces some information-theoretic ideas that might be useful for constructing a general architecture of unconventional computing systems from elements already tried and tested in the architecture of conventional ones. Such a rather reformist approach of step-by-step generalisation is, of course, not the only one possible; alternatively, the theory might start from scratch by introducing a most general mathematical framework in which more revolutionary concepts that try to capture essential aspects of self-organising systems can be defined and compared exactly (Tsuda et al., 2004). The utility of an information-theoretic framework is to be tested by using it in the analysis and synthesis of real information-processing systems. Thus, a particular unconventional computing system based on the true slime mould *Physarum polycephalum* and used as a bio-hybrid robot controller will be presented. Finally, information-theoretic aspects of the bio-hybrid controller will be described on a coarse-grained level using the ideas introduced first. By showing that the processes in such a controller can be systematically categorised from a general information-theoretic perspective, this description is meant to be a preliminary step towards a theory of bounded computability.

Syntactic, Semantic, and Pragmatic Representation of Information

A full-fledged concept of information integrates the distinction between the behavioural structure of a system, its functional structure, and the structure of its material medium. Information is not a concrete entity that can be localised in a particular part of a system; it is an abstract structure that covers the complex systemic interplay of matter, function, and behaviour. Basic information-theoretic concepts that are general enough to describe this interplay for a spectrum of systems as broad as possible, are the concepts of syntax, semantics, and pragmatics (Artmann, 2008). In the following, they are considered denoting different ways of representing physical systems from a unified information-theoretic point of view.

First, information can be represented *syntactically* by the material structure of a physical system. The spatio-temporal organisation of the material components of a system is then regarded as an actualisation of the syntactic structure of information in a physical medium. The material structure of the medium actually stands for the syntactic structure of information that is constituted by the set of relations of its elements. The dynamics of self-organisation of the physical medium drives the processing of the syntactic representation of information, but does not require a specific information-theoretic explanation. An important criterion for classifying computational media from a syntactic perspective is how efficient media are in processing syntactic representations of information to perform specific computational tasks.

Second, information can be represented *semantically* by the functional structure of a physical system. The causal order between the material components of a system is then regarded as an implementation of the semantic structure of information by a physical medium. This semantic structure is constituted by codes. A code connects two syntactic structures with each other. 'Code' is the information-theoretic name of a mapping that relates, in case of encoding, each of the possible syntactic elements of a message to a possible element of a signal and, in case of decoding, each of the possible syntactic elements of a signal to a possible element of a message (Cover and Thomas, 2006). The dynamics of self-organisation of the physical medium implements the semantic structure of information by encoding and decoding syntactic structures in physical processes. From the perspective of semantics, it is necessary to interpret the present state of (a part of) a system as encoding the future state (of another part) of it. An important semantic criterion for classifying computational media is how general the codes used in a medium to implement semantic representations of information are, i.e., to which degree the codes are able to differentiate between possible messages under given boundary conditions.

Third, information can be represented *pragmatically* by the behavioural structure of a physical system. The pattern of interaction between the system and its environment is then regarded as an effectuation of the pragmatic structure of information through the agency of a physical medium. The pragmatic structure is constituted by transformations of boundary conditions on coding. When is a message selected for being encoded, when is a signal decoded, and how does the code originate? Generalising the idea that information is constituted pragmatically by the effect of a signal on its receiver (MacKay, 1969), the definition of the pragmatic structure of information involves at least two syntactic orders and one semantic mapping. The dynamics of self-organisation of the physical medium changes internal and external conditions of information processing in the system. From the perspective of pragmatics, it is necessary to interact with the present behaviour of a system in order to let its dynamics lead it to a particular future behaviour. An important criterion for classifying computational media from a pragmatic perspective is how versatile media are in effectuating transformations of the system's behaviour under changing boundary conditions, i.e., to which degree the behaviour of the system is able to adapt itself to different environments.

A theory of bounded computability, which relates generic traits of information-processing systems with specific physico-chemical constraints on the realisation of such systems in different classes of computational media, must deal with the interplay of syntactic efficiency, semantic gen-

erality, and pragmatic versatility. To get a first idea of this interplay, a real computing system whose further development requires information-theoretic backing, should be analysed. For this purpose, the following section introduces a naturally evolved information processor, and the section after the next describes how it is used in a bio-hybrid robot controller.

Physarum polycephalum as Information Processor

The plasmodium of the true slime mould, *Physarum polycephalum*, is an amoeba-like unicellular organism, whose body size ranges from several hundred microns to a radius of more than one meter (Fig. 1). Despite its large size, the single cell acts as an integrated organism and is known for its distributed information processing.

A plasmodial cell of *Physarum polycephalum* consists of an ectoplasm tube that encloses an endoplasmic core. The former is a gel membrane layer, while the latter is a more fluid state of the protoplasm (Wohlfarth-Bottermann, 1979). In the ectoplasm tube, cytoplasmic actomyosin periodically aggregates to form sheet-like structures and then unravels into fibrils. These structural changes create a hydrostatic pressure gradient within the cell, and eventually give rise to a flow of ectoplasm shuttling from one location in the cell to other parts of the cell and back. If a cell is not stimulated, this contractile rhythm is synchronised throughout the cell. However, when a local part of a cell is exposed to an external stimulus such as food or white light, it leads to desynchronisation of the rhythm. The frequency of the oscillating rhythm at the stimulated part increases if it is an attractive stimulus, or decreases if it is repulsive. Such local frequency change affects oscillations of other parts through protoplasmic streaming and forms a spatial phase pattern in the cell (Hejnowicz and Wohlfarth-Bottermann, 1980; Matsumoto et al., 1986; Tanaka et al., 1987). The emerging global phase pattern eventually determines the direction of migration, i.e., the behaviour of the organism (Matsumoto et al., 1988).

This mode of information processing affords scalability to the plasmodium. As long as the plasmodium is able to form the phase gradient of the contractile oscillation rhythm within its single-cell body, it reacts to various external stimuli in the same fashion no matter how large it grows. Central to this size-invariant behaviour is the spatial phase pattern of the oscillation rhythm formed within a cell. It emerges from the interaction of the intracellular dynamics of the plasmodium and the environment triggered by a contact with an external stimulus and the plasmodium. Several theoretical models have been proposed to explain the behaviour (Miura and Yano, 1998; Miyake et al., 1996) based on the theory of positional information (Gierer and Meinhardt, 1972). It is interesting to note, that the information processing in the cell can access information about past states. Nakagaki and his colleagues found if it is exposed to periodic environmental changes the plasmodium is able to anticipate the next change by changing its behaviour at the time a periodic change is next due to occur; the memory persists over several hours (Saigusa et al., 2008).

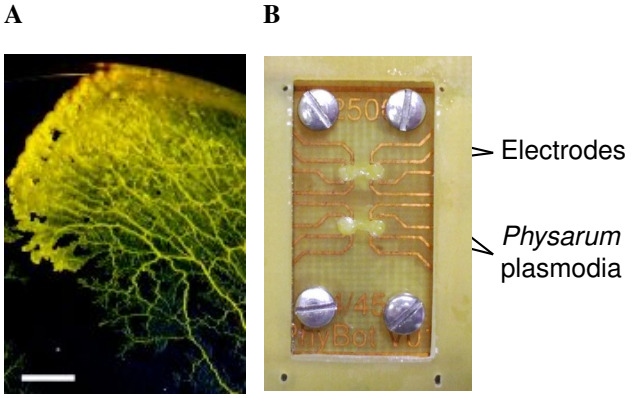


Figure 1: The plasmodium of the true slime mould *Physarum polycephalum* growing on the 1.5 % agar gel plate (A) and growing in the *Physarum* chip (B). A white bar on each panel is 5 mm.

Cellular Robot Control

The information-processing abilities of the plasmodium together with its humble requirements, suggest to use this simple organism in hybrid systems that fulfil some function for the generation of adaptive behaviour in engineered systems. For this purpose, the contractile oscillation dynamics of the cell was employed to control a robot.

Previously we worked on a bio-hybrid robot system controlled by the plasmodium, in which a robot and the plasmodium are connected with a bi-directional optical interface (cf. Tsuda et al., 2006a,b). However, the optical interface design sets the limit for the complete integration of the cell into bio-hybrid robot devices because the robot was remotely controlled by the plasmodium, which was located under a microscope in a humidity-controlled chamber.

Our recent work addresses this issue. As seen in other robotic systems using uncomventional computing devices (Adamatzky et al., 2004), we focused on an on-board robot controller design to integrate the plasmodium into an autonomous robot. For implementing the design, a new interface between the robot and the cell is required. The optical measurement of the plasmodium's oscillations required bulky equipment and it was therefore desirable to explore other technologies for monitoring the activity of the plasmodium. A custom circuit board for electrical impedance spectroscopy (EIS) has been developed (Macey, 2007) and mounted on a small wheeled robot platform (Jones, 2006).

Figure 2 shows the new setup of the bio-hybrid robotic system. The system consists of four components: a

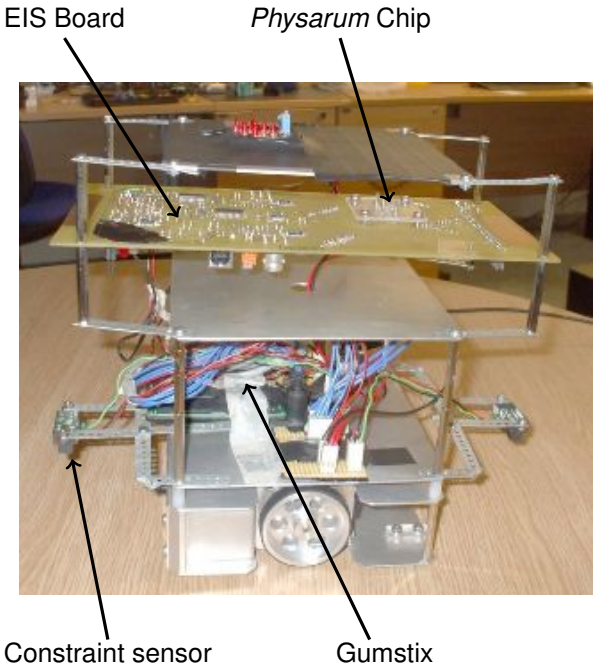


Figure 2: The complete setup of the robotic system driven by the *Physarum* plasmodium.

Physarum chip (Fig. 1B), the EIS board, a small gumstix computer, and a wheeled robot base. In this configuration the cell's oscillations are read through impedance measurements and mapped onto the wheel motion of the robot base. The current implementation of the un-tethered robot still lacks the interface between the cell and sensors on the robot, i.e., the robot is driven by the cell's oscillating pattern without any feedback to the cell.

The *Physarum* chip is a small printed circuit board (PCB) containing two plasmodial cells and mounted with a plexi glass frame to the EIS circuit board. Each plasmodium is confined in a dumbbell-shaped cut-out of the PCB sheet, as shown in Fig. 1B. The dumbbell-shape design follows (Takamatsu et al., 2000a,b) who studied the oscillation patterns of the plasmodium confined to this shape.

The impedance measurement circuitry (EIS board) allows for non-invasive monitoring of the plasmodium's oscillation activity. Fig. 3A and B show signals from two consecutive time periods of a single experiment. The two curves show the magnitude of the impedance from the left and right circular areas (wells) of the dumbbell shaped plasmodium after a noise filter has been applied. Measured impedance data from the EIS board is converted into commands for the robot control and stored for subsequent analysis by the on-board gumstix computer.

It is known that the plasmodium if confined to the dumbbell shape show in-phase and out-of-phase oscillation patterns between the two wells (cf. Takamatsu et al., 2000b). Based on this observation, we introduced a simple mapping from oscillations to robot movement. The mapping is inspired by the motor-control in bacterial chemotaxis (Adler and Tso, 1974; Scharf et al., 1998): If the signal from the left well and the signal from the right well are in synchrony, the robot pivots either left or right randomly, otherwise the robot moves straight. The update cycle from impedance measurement to change of robot behaviour is once per second; for details see the materials and methods section at the end.

The trajectories of the robot that results from this mapping are shown in Fig. 3C and D. During the time period shown in panel A of Fig. 3, the oscillations of the two parts of the plasmodium cell are predominately synchronised and accordingly the robots trajectory shows many random pivot turns (Fig. 3C). On the other hand, in the period shown in panels B and D, the robot runs straight more often because the oscillation pattern switched to an out-of-phase mode about midway through the period shown in (Fig. 3B).

Although the current implementation of the bio-hybrid robot has only a one-directional interface from the plasmodium to the robot, the preliminary experiments indicate the feasibility of integrating a living cell into the controller of an autonomous robot. The next required step is the implementation of the converse interface from the robot to the cell, i.e., inputs to the plasmodium. This may be achieved by illuminating the cell with white or blue light from LEDs according to signals from sensors on the robot. This part of the interface is still under investigation, however we expect it to be much simpler to realise than the cell-to-actuator interface described above.

To close the interaction loop of artificial control, natural organism, and environment so that it can be used for the construction of an adaptively behaving robot, this loop can be analysed in terms of information theory. In the following, it will be described on a coarse-grained level that allows for a general differentiation between the syntactic, semantic, and pragmatic representation of information. The focus is on the plasmodium as a medium that does bounded computation under specific internal and external physico-chemical constraints on information processing.

Application of Information-Theoretic Framework to the Bio-Hybrid Robot

The following general information-theoretic description of the interaction loop of artificial control, natural organism, and environment assumes a bidirectional interface between the organism and the robot's sensor and actuators as already implemented in the earlier, tethered robot (Tsuda et al., 2007). As mentioned above, a bi-directional interface for the robot with the integrated cell is still under development. For the following discussion, however, it is not crucial whether the plasmodial cell is located in the robot; for practical applications of course it is.

Where is information represented semantically in the interaction loop? First and foremost, in the code-based func-

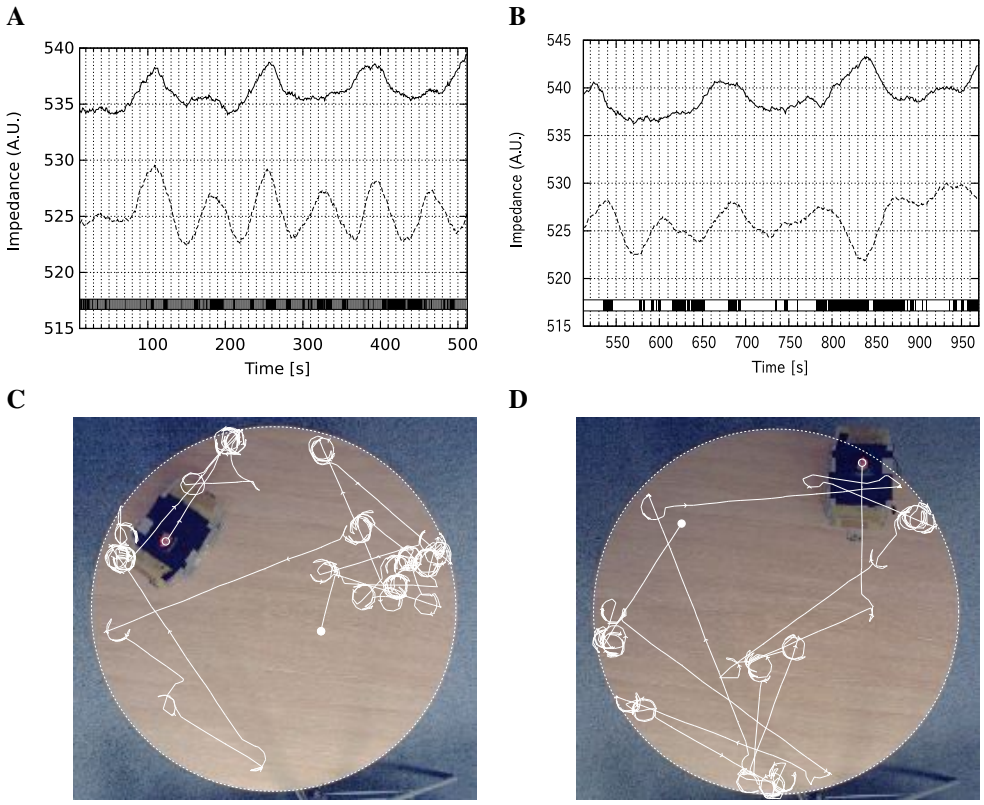


Figure 3: Oscillation of a plasmodium in a *Physarum* chip and the corresponding trajectories of the robot. The moving-averages of the magnitude of the impedance at 100k Hz are plotted for 30–500 s (A) and 500–970 s (B). The solid and dotted curves in the plots correspond to oscillations of the plasmodium from right and left wells, respectively. The phase relationship between two wells are shown in the bottom of the plots as black (in-phase) and white (out-of-phase) vertical lines. The behaviour of the robot is determined according to the phase relationship as traced in panel (C) for 30–500 s and panel (D) 500–970 s. A solid circle indicates the start and an open circle the end of the trajectory.

tional structure of the artificial control. Encoding happens when stimuli from the robotic light sensors are transduced to white light signals for the cell. Decoding occurs when amplitude signals calculated from measured data of the plasmodium oscillators are processed by software to alter the motion of the robot actuators. The control software acts as a decoding device that semantically relates a syntactic representation of information (plasmodium oscillation signals) to another syntactic representation of information (robot motor signals). All in all, four syntactic representations of information (external light signals, white light signals for the plasmodium, oscillation signals from the plasmodium, and robot motor signals) are related by two semantic representations of information (namely, the code used for encoding the light sensor data into white light signals and the code used for decoding the oscillation data into motor signals). In-between, the plasmodium connects the two semantic representations by bounded computation (see Fig. 4).

Where is information represented pragmatically in the interaction loop? First, the behaviour of the robot that results from decoding plasmodium oscillation signals into robot motor signals, changes the boundary conditions on encoding since the effects of the robot's activity on the environment are perceived by the robot's light sensors whose data is then encoded into white light signals for the cell. Second, the plasmodium behaves according to its own dynamics in its direct environment, i.e. in the artificial control. This environment receives the behaviour of the plasmodium in form of oscillation data that is decoded into robot motor signals. The pragmatic interaction of the plasmodium with its engineered environment is, thus, semantically represented in the very same environment and then pragmatically represented by the behaviour of the robot in its real-world environment. Connecting the relation of the robot to its real-world environment with the relation of the plasmodium to its artificial environment, it results that the semantic structure of the robot control device (in short, its *control semantics*) is given by the two codes mentioned above. They map two different pragmatic representations of information to each other, namely the behaviour of the plasmodium and the behaviour

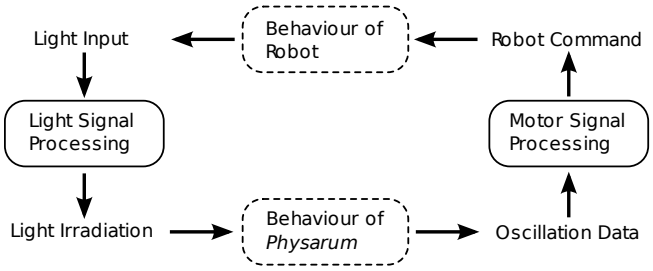


Figure 4: Interaction loop of *Physarum*-controlled robot. Each part of the diagram corresponds to either syntax (no box), semantics (solid box), or pragmatics (dotted box).

of the robot (see Fig. 4).

Given a particular control semantics and a specific environment, the behaviour of the robot can be anticipated by an external observer. This does not necessarily mean the bio-hybrid robot system would always behave as its designer or observer expects. The key issue here is how we can exploit the cell's self-organising dynamics to achieve a fully autonomous robot. Arguably, biological cells outperform conventional autonomous robots in many features by exploiting pragmatic versatility, i.e. the high degree to which the cells are able to adapt their behaviour to different environments.

In fact, several researchers have observed that the plasmodium is able to spontaneously change its behaviour pattern against external stimuli to overcome unfavourable conditions (Aono and Hara, 2007; Nomura, 2001; Takamatsu et al., 2004). For example, Takamatsu found that if the plasmodium is entrained to oscillate at a fixed frequency by external periodically-changing stimulus, it spontaneously deviates from the frequency after a certain period even if the stimulus is maintained. She speculated that such spontaneous change might stem from multistability or chaotic behaviour of the plasmodium's dynamics and may contribute to the diversity of behavioural modes of the plasmodium, such as food-searching mode and feeding mode (Takamatsu et al., 2004).

Although physiological mechanisms underlying such behaviour are yet to be investigated, these observations point to richness of the cell's internal dynamics. However, they also point to the lack of theory about relations between the dynamics of physico-chemical material structures and their use as computational media. The conventional computing paradigm assumes perfection of each part in a system. It is, therefore, inadequate when we want to harness the pragmatic versatility of the plasmodium, which results from the richness of its self-organised processing of syntactic representations of information, by a control semantics that allows the robot to adapt its behaviour to a real-world environment.

The development of a control semantics for devices like the plasmodium-based robot controller is an important engineering contribution to the construction of a general architecture for unconventional computers. This architecture could be described in the conceptual framework of a theory of bounded computability that relates generic traits of information-processing systems with physico-chemical constraints on the realisation of such systems in different classes of computational media.

The information-theoretic sketch of some processes in the bio-hybrid controller given above hints at general features that seem to be fundamental to the architecture of such devices, and perhaps of other unconventional computing systems, too. However, the largely varying physico-chemical properties of the computational media used in those systems make it difficult to propose bold yet reasonable generalisations. The following remarks that address syntactic efficiency, semantic generality, and pragmatic versatility try to make a virtue out of necessity by drawing some consequences from the significance of specific physico-chemical properties of unconventional computational media.

First, the material features of computational media like the plasmodium appear, from the perspective of the controller, as constraints on the processing of syntactic representations of information. Information processing by the computational medium has, of course, also internal semantic and pragmatic aspects. There exist, e.g., semantic representations of information in the cell like the organic code that structures the expression of genetic information in the plasmodium (Barbieri, 2003). Yet from the perspective of the controller those cell-internal aspects are to be considered just as constraints on the efficiency of processing syntactic representations of information that are actualised in the cell's environment. Therefore, the syntactic effectiveness of the computational medium shows itself in its pragmatic versatility, i.e. the degree to which its behaviour is able to adapt itself to changing boundary conditions that bear information syntactically.

Second, the semantic substructures of the controller, the codes used for encoding external stimuli and for decoding internally collected measurement data, are pragmatically connected by the behaviour of the computational medium, i.e. by how it measurably reacts to the encoded stimuli. To encode information means in pragmatic respect that the control device sets the boundary conditions on how the computational medium processes syntactic representations of information. To decode information means that the control device semantically represents the pragmatic results of the information processing by the computational medium. This functionally differentiated interplay of semantics and pragmatics in the controller is as important as the syntactic efficiency of the isolated organic computational medium, since it is the means by which the pragmatic versatility of the cell is also detectable in the behaviour of the robot.

Third, from the perspective of the computational medium, the control device is a behaviour amplifier. The microscopic behaviour of a cell is amplified to the macroscopic behaviour

of a robot by semantic means. This suggests to think about the generality of codes, i.e., about the degree to which they are able to differentiate between possible messages under given pragmatic boundary conditions, in terms of the graininess of behaviours between which the implemented codes can differentiate. The finer the behavioural differences of the cell that a code can semantically represent are, the more general the code is in respect to this particular control setting.

These remarks highlight some features of the interplay between syntactic, semantic, and pragmatic representations of information in unconventional computing systems. They indicate to which information-theoretic problems, not only the further development of the plasmodium-based robot controller, but also the construction of a general architecture of unconventional computers should concentrate its attention.

Materials and Methods

Plasmodia of *Physarum polycephalum* were cultured on 1.5 % agar gel plate and fed with oat-flakes. They were starved for more than 24 hours before the experiment. When plasmodia are transferred into the chip, tip portions of the cell taken from the anterior region of a thin-spread culture are used. The chip, shown in Fig. 1B, can host two independent plasmodia for monitoring. Each plasmodium cell is confined in a dumbbell-shaped cut-out of a thin sheet of printed circuit board (470 μm). To maintain the moisture required by the cell, the PCB is covered on one side with an approximately 0.5 mm thick layer of 1.5 % agar gel, and on the other side with an approximately 50 μm thick sheet of the gas-permeable elastomer polydimethylsiloxane (PDMS). The copper side of the PCB with its patterned electrodes faces the agar gel and is insulated from it with laminate. The stack of PDMS–PCB–Agar is clamped with a plexiglass frame. This assembly, referred to as "*Physarum* chip", completely encloses the plasmodial cell and provides the necessary humidity and adequate oxygen supply to keep the cell active for more than 5 hours.

The dumbbell-shaped design, two 1.6 mm diameter circular holes at a centre-distance of 2.5 mm connected by a 0.4 mm wide channel, is modelled on the design reported in Takamatsu et al. (2000b). A prepared *Physarum* chip is mounted on the EIS board and left to stand in a dark place for 2–3 hours or more until the cell starts steadily oscillating in the chip. The PCB sheet is equipped with a total of eight pairs of electrodes for the two plasmodia samples, two electrode pairs for each well (Fig. 1B). During the incubating period, electrical impedances of plasmodia at 100 kHz AC frequency are constantly monitored via these electrodes to trace the oscillatory activity of the cells. Based on the strength of the oscillation signal recorded, one of two plasmodia is selected to be used for the control of the robot.

In the robot control experiment, impedances of the plasmodia at the eight points are measured once per second and saved in the flash memory of the gumstix computer. Although the data from all eight electrode pairs are recorded throughout the experiment, only one of two electrode pairs available at each well is selected for the control of the robot. As in the case of selecting a plasmodium, the criterion is the strength of the oscillation signal received from the electrode pair.

The robot carries a computer $8\times3.5\times2$ cm^3 in size on which a customised Linux kernel has been installed (`www.gumstix.com`). This computer serves for signal processing and as a data logger, recording the impedance measurements in flash memory for off-line analysis. To this end the computer configures the EIS board over an I^{2}C bus, configures the impedance circuitry, controls the analog-switches that multiplex among the electrodes and retrieves the impedance measurements. After the signal processing described below, the computer sends commands to a microcontroller in the wheeled robot base.

The wheeled robot base is a minimalist design based on the Braitenberg vehicles (Braitenberg, 1984). It allows for the *Physarum* chip, the EIS board, and the gumstix computer to be mounted, and accommodates the necessary power supplies. The base has its own microcontroller that translates simple commands (forward, left, right) to the drive level of the stepper motors. The microcontroller also monitors two infra-red proximity switches and ignores forward commands if one of these switches detects a cliff. This effectively constrains the robot to the area of the table that serves as arena for experiments.

A two-step process converts the measured impedance signals into drive commands for the robot base: Signal processing to recover the oscillation state of the cell from the impedance measurements, and mapping of the cells oscillation state into actuator commands.

First, the moving average over 15 samples ($\approx$ 15 seconds) of recorded data is calculated to reduce noise in the impedance measurements. At present the circuitry on the robot has not been optimised to reduce noise, but the signals are strong enough that the simple moving average filter works sufficiently well for our purpose. The curves in Fig. 3A and B show the signals after the noise filtering step. Next a differential of the averaged signal is computed by subtracting the 15 s delayed signal to remove the long-term trend of the signals (Nakagaki et al., 1996).

After the signal processing, a command to drive the robot is determined according to the phase relationship between the differential signals from both wells: If the two wells are in phase (synchronised mode), the robot takes a random turn. If they are out of phase (phase delayed mode), then it moves straight. The phase relationship is classified by the following simple rule: If the signs of the two differential signals are equal (both oscillations are increasing or both are decreasing) the oscillation state is classified as synchronised mode. If the signs of the differential signals differ the oscillation state is classified as phase delayed mode. The former is mapped into a random choice of either a "left" or a "right" command, the latter is mapped into a "forward" command. The whole conversion cycle is performed once per second and commands for the robot's actuators are issued accordingly.

The robot system was tested on a 1 m diameter round table, the robot being constraint to this area by the cliff sensing described above. Position and direction of the robot are tracked by an Ethernet camera mounted above the table using an illuminated target pattern on top of the robot.

References

Adamatzky, A., de Lacy Costello, B., Melluish, C., and Ratcliffe, N. (2004). Experimental implementation of mobile robot taxis with onboard Belousov-Zhabotinsky chemical medium. *Materials Science & Engineering C*, 24(4):541–548.

Adler, J. and Tso, W.-W. (1974). "Decision"-making in bacteria: Chemotactic response of Escherichia coli to conflicting stimuli. *Science*, 184:1292–1294.

Aono, M. and Hara, M. (2007). Amoeba-based nonequilibrium neurocomputer utilizing fluctuations and instability. In *6th International Conference, UC 2007*, volume 4618 of *LNCS*, pages 41–54, Kingston, Canada. Springer.

Artmann, S. (2008). Biological information. In Sarkar, S. and Plutynski, A., editors, *A Companion to the Philosophy of Biology*, pages 22–39. Malden/MA: Blackwell.

Barbieri, M. (2003). *Organic Codes: An Introduction to Semantic Biology*. Cambridge University Press, Cambridge/England.

Braitenberg, V. (1984). *Vehicles: Experiments in Synthetic Psychology*. MIT Press, Cambridge, MA.

Cover, T. M. and Thomas, J. A. (2006). *Elements of Information Theory*. Wiley-Interscience, 2nd edition.

Gierer, A. and Meinhardt, H. (1972). Theory of biological pattern formation. *Kybernetik*, 12:30–39.

Hejnowicz, Z. and Wohlfarth-Bottermann, K. E. (1980). Propagated waves induced by gradients of physiological factors within plasmodia of *Physarum polycephalum*. *Planta*, 150:144–152.

Jones, G. (2006). Robotic platform for molecular controlled robots. Part III Project Report, School of Electronics and Computer Science, University of Southampton.

Macey, P. (2007). Impedance spectroscopy based interfacing with a living cell for biosensors and bio-coporcessors. Part III Project Report, School of Electronics and Computer Science, University of Southampton.

MacKay, D. (1969). *Information, Mechanism and Meaning*. Cambridge, MA: MIT Press.

Matsumoto, K., Ueda, T., and Kobatake, Y. (1986). Propagation of phase wave in relation to tactic responses by the plasmodium of *Physarum polycephalum*. *Journal of Theoretical Biology*, 122:339–345.

Matsumoto, K., Ueda, T., and Kobatake, Y. (1988). Reversal of thermotaxis with oscillatory stimulation in the plasmodium of *Physarum polycephalum*. *Journal of Theoretical Biology*, 131:175–182.

Miura, H. and Yano, M. (1998). A model of organization of size invariant positional information in taxis of *Physarum* plasmodium. *Progress of Theoretical Physics*, 100(2):235–251.

Miyake, Y., Tabata, S., Murakami, H., Yano, M., and Shimizu, H. (1996). Environmental-dependent self-organization of positional information field in chemotaxis of *Physarum* plasmodium. *Journal of Theoretical Biology*, 178:341–353.

Nakagaki, T., Uemura, S., Kakiuchi, Y., and Ueda, T. (1996). Action spectrum for sporulation and photoavoidance in the plasmodium of Physarum polycephalum, as modified differentially by temperature and starvation. *Photochemistry and Photobiology*, 64(5):859–862.

Nomura, S. (2001). *Symbolization of an object and its freedom in biological systems*. PhD thesis, Kobe University.

Papadimitriou, C. M. (1994). *Computational complexity*. Addison-Wesley, Reading, Massachusetts.

Saigusa, T., Tero, A., Nakagaki, T., and Kuramoto, Y. (2008). Amoebae anticipate periodic events. *Physical Review Letters*, 100(1).

Scharf, B. E., Fahrner, K. A., Turner, L., and Berg, H. C. (1998). Control of direction of flagellar rotation in bacterial chemotaxis. *Proc Natl Acad Sci USA*, 95(1):201–206.

Shannon, C. and Weaver, W. (1949). *Mathematical Theory of Communication*. University of Illinois Press.

Simon, H. (1982/1997). *Models of Bounded Rationality, three volumes*. MIT Press, Cambridge, Massachusetts.

Takamatsu, A., Fujii, T., and Endo, I. (2000a). Control of interaction strength in a network of the true slime mold by a microfabricated structure. *BioSystems*, 55:33–38.

Takamatsu, A., Fujii, T., Yokota, H., Hosokawa, K., Higuchi, T., and Endo, I. (2000b). Controlling the geometry and the coupling strength of the oscillator system in plasmodium of *Physarum polycephalum* by microfabricated structure. *Protoplasma*, 210:164–171.

Takamatsu, A., Yamamoto, T., and Fujii, T. (2004). Spontaneous switching of frequency-locking by periodic stimulus in oscillators of plasmodium of the true slime mold. *BioSystems*, 76:133–140.

Tanaka, H., Yoshimura, H., Miyake, Y., Imaizumi, J., Nagayama, K., and Shimizu, H. (1987). Information processing of Physarum polycephalum studied by micro-thermography. *Protoplasma*, 138:98–104.

Tsuda, S., Aono, M., and Gunji, Y.-P. (2004). Robust and emergent *Physarum* logical-computing. *BioSystems*, 73:45–55.

Tsuda, S., Zauner, K.-P., and Gunji, Y.-P. (2006a). Computing substrates and life. In Artmann, S. and Dittrich, P., editors, *Explorations in the Complexity of Possible Life: Abstracting and Synthesizing the Principles of Living Systems*, Proceedings of the 7th German Workshop on Artificial Life, pages 39–49, Jena, Germany. IOS Press.

Tsuda, S., Zauner, K.-P., and Gunji, Y.-P. (2006b). Robot control: From silicon circuitry to cells. In Ijspeert, A. J., Masuzawa, T., and Kusumoto, S., editors, *BioADIT 2006*, volume 3853 of *Lecture Notes in Computer Science*, pages 20–32. Springer.

Tsuda, S., Zauner, K.-P., and Gunji, Y.-P. (2007). Robot control with biological cells. *BioSystems*, 87:215–223.

Wohlfarth-Bottermann, K. E. (1979). Oscillatory contraction activity in physarum. *The Journal of Experimental Biology*, 81:15–32.

Zauner, K.-P. (2005). Molecular information technology. *Critical Reviews in Solid State and Material Sciences*, 30(1):33–69.

Movement Strategies for Learning in Visual Recognition

Edgar J. Bermudez*, Andrew Philippides and Anil K. Seth

University of Sussex. UK
*E.J.Bermudez-Contreras@sussex.ac.uk

Abstract

In this paper we study the role of movement strategies during learning in object recognition models. We show that a simple model, the RBF, can outperform a more complex hierarchical model, the HMAX, when rotation and scale invariance are provided by the training phase. Moreover, we assess the exploitation of temporal information by the RBF using optic flow. The results show that the RBF model can only exploit the temporal information using optic flow when the training and testing trajectories are the same. This work exemplifies the idea that the complexity of the neural mechanisms in object recognition can be understood not only in the brain but also in the interaction between brain, body and environment.

Introduction

Object recognition is a very complex computational task that has been widely studied. Whereas visual systems in nature solve this task with exceptional reliability and speed, the performance of artificial visual systems is still far from their counterpart in nature. We are interested in the exploration of ways active vision can help biologically inspired models of object recognition in autonomous agents. In order to understand the visual processes in the brain and design artificial visual systems, various models have been proposed for object recognition based on different perception theories. These models can be classified as object-based or view-based. The former category describes models that "extract" structural features or parts of the object that are view-invariant in a 3D coordinate system centred on the object. In contrast, view-based models represent objects as a combination or set of features extracted directly from the image.For a review of models and theories of perception see (Riesenhuber and Poggio, 2000; Peters, 2000). Most state-of-the-art models are view-based, which in turn, are divided by the way they extract the view-based features. Some computer-vision based models use statistical regularities extracted from the images, mainly using, template or histogram systems (bag-of-features, nearest-neighbour, etc.) (Wang et al., 2006; Zhang et al., 2006; Lazebnik et al., 2006). Others are biologically inspired, resembling the hierarchical nature of the visual cortex (Riesenhuber and Poggio, 1999; Poggio and Edelman, 1990; Serre et al., 2005; Mutch and Lowe, 2006).

Template-based models perform very well on object recognition of single object category (e.g. faces, cars, etc.). However, these methods show limitations when the object is subject to appearance modifications, suffering from high specificity and therefore, lacking invariance to object transformations. Histogram-based models show a large amount of invariance to transformations but their performance drops for general object recognition tasks (i.e. with multiple object categories) (Serre et al., 2005). Biologically inspired models for object recognition have been gaining interest because they perform very well for general purpose object recognition tasks (Pinto et al., 2008).(Serre et al., 2005), presented a modified hierarchical model based on (Riesenhuber and Poggio, 1999) and reported it to be at least comparable to the best computer vision-based systems.

A common baseline of these systems is that they do not acquire the incoming visual information by themselves, the way the visual information is presented to them is restricted by the experimenter. In some cases, these imposed restrictions can play an important role in the recognition process and hence, in the performance and evaluation of models. For example, in (Bermudez-Contreras et al., 2007), it was shown that a simple model of the primary visual cortex [(RBF), (Howell and Buxton, 1995; Poggio and Edelman, 1990)]can perform just as well as a complex hierarchical model [(HMAX), (Riesenhuber and Poggio, 1999)] when natural conditions are present and the former is augmented by a simple 'attentional mechanism'. In addition, in (Pinto et al., 2008), a comparison between state-of-the-art object recognition systems and a simple V1-like model is carried out. They show that by imposing conditions on the way the visual information is presented to the systems (taken from databases of natural images), the simple biological model outperforms all of the state-of-the-art systems presented.

Given this importance, an additional fact to be considered when studying or modelling visual systems is that, in nature, visual systems are active. In active vision, the control of acquisition of visual information is part of the system.

It is well known that the restrictions imposed by the interaction between body and environment can facilitate visual processing (Aloimonos, 1993). Active vision strategies are important both in recognition and in visual learning. For instance, insects utilise specific movement strategies in order to learn how to perform various visually guided tasks including homing, navigation, and finding conspecifics (Lehrer and Bianco, 2000; Collett and Rees, 1997; Carwright and Collett, 1983). Therefore, in the study and modelling of visual systems, it is important to consider the way incoming visual information is acquired.

There have been relatively only a few studies that analyse the role of motion and object recognition. Arbel and Ferrie (2002, 2001) propose a paradigm to facilitate object recognition of a system. Most of the research work on visual systems in autonomous robots is oriented to navigation, with some exceptions. For example, in (Gvozdjak and Li, 1998), the importance of active vision in an agent for recognition tasks is highlighted using a hierarchical template-based model. In (Andreasson and Duckett, 2003), an exploratory study of object recognition using a mobile robot with an omni-directional camera is presented. The robot tracks extracted low-level features and constructs higher level features for object identification. While these works are exploratory, they show the potential of active vision in object recognition tasks. Furthermore, given the success of object recognition models that reflect the hierarchical nature of the visual cortex, we evaluate the importance of the role of visual information acquisition processes in these models.

In this work, we analyse how different movement strategies during training affect the performance of a version of the HMAX model and the RBF model. We employ a mobile agent in a simulated world with a simple object recognition task. We find that movement strategies are exploited differently for both models. When a movement strategy does not provide the opportunity to develop rotation and translation invariance, the HMAX model performs better than the RBF. However, when such opportunities are provided, the RBF model outperforms the HMAX model. These results suggest that exploiting the dynamics of agent-environment interaction can, in certain circumstances, obviate the need for complex models of visual object recognition. We also consider whether the RBF model performance can be further enhanced by training and testing using dynamic visual signals generated during each movement strategy. We find that such time dependent information is only exploited by the RBF model when the training and testing movement strategies are the same.

Methods

The following experiments involve a simulated agent performing a simple object recognition task. The agent-environment system comprises a simple wheeled agent in a flat planar environment containing two objects (a 'kettle' and a 'bolt'), simulated using the OpenGL library. It is important to mention that the goal of this exploratory study is to investigate how the way of acquiring visual information can affect the recognition process in two models of object recognition rather than comparing their performance. The visual object recognition system of the agent comprises three parts: a 'blob detection mechanism' (BDM), an 'analysis module' consisting of either the HMAX or the RBF model, and a 'classifier module' which classifies the output of the analysis module into one of two categories (kettle' or 'bolt'). Each experiment consisted of two phases. First, a learning phase in which the agent followed one of four different movement strategies (see figure 3) while collecting training views which are used to train either the HMAX or the RBF model. Second, a testing phase, during which the agent follows a separate movement strategy (see figure 2A) while collecting views used to test object recognition performance.

Blob detection mechanism. The BDM selects the area of the visual field containing the object. It is the 'attentional mechanism' referred to in the introduction. Cropped regions returned by the BDM are normalised to 60×80 pixels (a blob) before being processed by the analysis module. The BDM therefore provides some robustness to changes in the size of objects. The order in which blobs are processed by the visual system is determined by the area of the blob detected. The larger the area of the blob in the visual field, the higher the priority of being processed by the visual system [see (Bermudez and Seth, 2007) for a more detailed explanation of the visual system of the agent].

Analysis module. The analysis module processes visual information coming from the BDM (current views). These views are processed by either the HMAX or the RBF model. The RBF model emulates simple cells in the primary visual cortex, V1, based on the function of receptive fields implemented by using Derivative of Gaussian filters with different orientations and sizes. The RBF model uses four different sizes of square filters with sides of 7, 11, 15 and 21 units and 0, 45, 90, and 135 degrees of orientation. There are therefore 16 different filters in total with outputs responding to oriented 'edges' at different spatial scales. Therefore, this model responds only to a collection of simple primary features. In contrast, the HMAX model proposed in (Riesenhuber and Poggio, 1999) is a hierarchical model resembling the ventral pathway in the visual cortex. The HMAX model consists of four layers (S1, C1, S2 and C2) resembling simple and complex cells in the ventral pathway. Units in S1 would correspond to simple features detected by the different filters of the RBF model. The next layer C1, responds to the most salient features in S1 at each orientation and spatial scale. It achieves this by applying max pooling operations (extracting the most salient features across the different orientations and spatial scales) over the selected features in S1. The next layer, S2 combines the output of C1 into a higher order features sets which are passed into C2 where the out-

puts are again max pooled to produce a vector of the dominant features detected along the hierarchy (see details of the original model in (Riesenhuber and Poggio, 1999) and details of this implementation in (Bermudez-Contreras et al., 2007)). By virtue of its hierarchical structure, this model shows a degree of translation and scale invariance.

Classifier module.

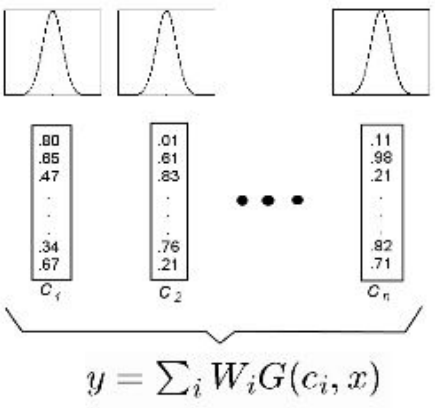


Figure 1: View Tuned Unit (VTU): each view vector c_i is the centre of a Gaussian function. The more similar a vector x is to a centre, the stronger the response of the unit.

The classifier module is based on the work of (Edelman and Duvdevani-Bar, 1997; Poggio and Edelman, 1990). It uses view tuned units (VTU) to recognise objects. There is one VTU for each object. Each VTU is trained to respond so that it responds strongly to test views that are similar to the training views of the object. Each VTU (see figure 1) corresponds to a set of radial basis functions (RBF unit). A RBF unit is a Gaussian function G centered on each view c_i collected during the training phase. The response of each RBF is given by $G(c_i, v) = e^{-\|c_i - v\|/\sigma_i^2}$ where v is the vector that is being classified.

The response y of each VTU for a test vector x is given by $y = \Sigma_i W_i G(v_i, x)$, that is, y is a linear combination of weights W_i and $G(v_i, x)$. The weights W_i are computed using an inversion matrix procedure (the details are described in the appendix section in (Bermudez-Contreras et al., 2007)).

Movement strategies. The training views were collected when the agent was navigating around or approaching the object, following one of four different trajectories (these trajectories are called movement strategies throughout the rest of this paper). The training views were processed by the analysis module (using the RBF model or the HMAX model) and learned and classified by the classifier module.

The properties of the set of training views changed depending upon the movement strategy used during their collection. These strategies were designed in order to provide different properties in the training views (see figure 3). movement strategy 1 allows the agent to exploit the different training distances while using the same point of view. Therefore, the training views using this strategy only provide variance in scale. Strategy 2 provides a small degree of variance in perceived rotation (points of view) and a small degree of variance in scale as well, since the agent is passing in front of the target object. The point of view changes slightly as the distance between the agent and the object changes. Strategy 3 provides only variance in points of view since the distance between the agent and the object is always the same, while the point of view changes for each training view. Strategy 4 provides a combination of variance in scale and point of view since the distance and the perspective of the agent to the object are changing continuously. For each strategy, 16 training views are taken for each object at regular time intervals. Therefore, training phases varied in length from 160 to 200 time steps depending on the movement strategy used.

In the testing phase, the agents followed a trajectory (testing trajectory) that differs from the movement strategies used in the learning phase. The testing trajectory was designed so it would resemble a plausible situation in the real world where the objects are approached in a natural way that provides views of the objects from multiple angles and scales (see figure 2). The testing phase lasted for 200 time steps. During the first 55 steps (period 1) object 1 was present in the visual field and during 125-180 (period 2) object 2 was present in the visual field (see testing trajectory in figure 2).

Optic flow. An important consequence of actively exploring the world is the visual motion that this evokes. Optic flow is defined as this type of motion. In our study, we calculated a simple approximation of optic flow by taking the absolute difference between consecutive views i and j, $F = 1/2 \cdot \|RBF(i) - RBF(j)\|$ after being processed by the RBF model. For the rest of the paper, F is referred to as RBF optic flow.

Experiment 1: Movement strategies

To assess the role of active vision in the object recognition models, we tested the RBF and HMAX using the different movement strategies shown in figure 3 during the learning phase. The models are then tested while the agent traverses the testing trajectory shown in figure 2A. The results are shown in figure 3.

For strategy 1, HMAX outperforms the RBF model. Since this movement strategy presents the objects from a single point of view, the models can only acquire scale invariance. For a simple model like the RBF, this strategy would only work if the objects were viewed from a similar perspective to training during the test phase. Since this is not the case (the point of view is changing and is different from training), the RBF model cannot closely match test views to training. However, the HMAX model is able to generalise when a limited point of view is provided during training. This is

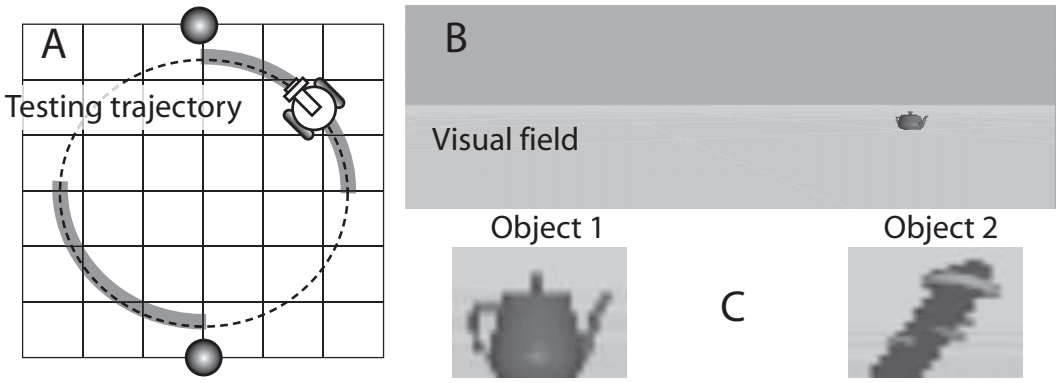


Figure 2: A. Testing trajectory: the grey segments represent the periods where objects were present in the visual field. B. Visual field of the agent: shows object 1 in the field of view. C. Sample views of object 1 and object 2: object 1 is a rounded object so it does not have a significant variability to rotation, in contrast, object 2 has a significantly higher variability to rotation due to its vertical inclination.

because the features extracted by HMAX from a single perspective capture higher order properties of the objects which are in some sense independent of the angle it is viewed at. For strategy 2, the results are very similar to the previous case as the training views are again taken from a limited set of angular positions. However, when the point of view is varied significantly during the training phase in strategy 3, the RBF model's performance increases greatly. Since the number of points of view is significantly increased, the RBF can achieve a close match between the training and the test views. In contrast, HMAX's performance decreases, demonstrating that its discriminability can be reduced when the variability of the training views is increased. Similar results are obtained for strategy 4 where both point of view and scale are changed during training.

The reason the models' performance changes with different movement strategies has to do with the way the objects change with the movement of the agent and also with the features detected by each model. In particular, the variability of the objects to rotation is significantly different. As object 1 is quite round (see figure 2), its image does not change significantly when the agent rotates around it (especially at large distances). In contrast, object 2 has a vertical orientation which makes it very variable when the point of view is changed.

Since the RBF model responds mainly to oriented edges (as it simply comprises a set of differently oriented filters at different spatial scales), its response depends on a close match between the test and the training views and we would expect it to fail when a close match is not possible. When the points of view are limited (strategy 1 and strategy 2), since the features detected for object 1 do not change significantly with the change of perspective (along the testing trajectory), the RBF model has a relatively close match between training and test views. Object 2 is difficult to discriminate, however, as it changes significantly along the testing trajectory. Thus the overall performance on these strategies is around 50%. Because the HMAX model acts on a combination of the dominant features detected by the RBF (since its first layer is the RBF), it responds to a more generalised pattern of features, rather than a close match. Since object 1 does not change significantly, the dominant features will be the ones responding to the main orientation of the object (horizontal). For object 2, if the object is seen from a single point of view, the dominant features will be the ones corresponding to the main orientation of the object, roughly 30 degrees from the vertical in the case of strategy 1. These features (which form the HMAX template for object 2), will be different to the dominant features detected for object 1 (which form the HMAX template for object 1), so the discriminability of the HMAX model is high in this case.

In contrast, when the point of view is varied significantly during the training phase (strategies 3 and 4), the RBF achieves a close match. Since there are more points of views in the training set, the model can cope with object rotation. In the case of the HMAX model, since object 2 changes its orientation during training, the model extracts dominant features in many orientations, which form a very general template and thus decrease object discriminability. This scenario is depicted in figure 4 which shows the models' output after training with strategy 3. The objects are within the field of view in different periods (grey segments in figure 2) during the 200 time step trial. In period 1 (1-55 time steps) object 1 is within the field of view, and in period 2 (125-180 time steps) object 2 is within the field of view. For the RBF model, the agent can correctly discriminate both objects. Note the peak in output that corresponds to a close match between test and training view (around time step 37). In the case of the HMAX model, while there is no problem with period 1, in period 2 discriminability is reduced significantly.

Similarity maps further explain the discrimination ability of the models (figure 5). A similarity map is a diagram representing the similarity between the current view (the one

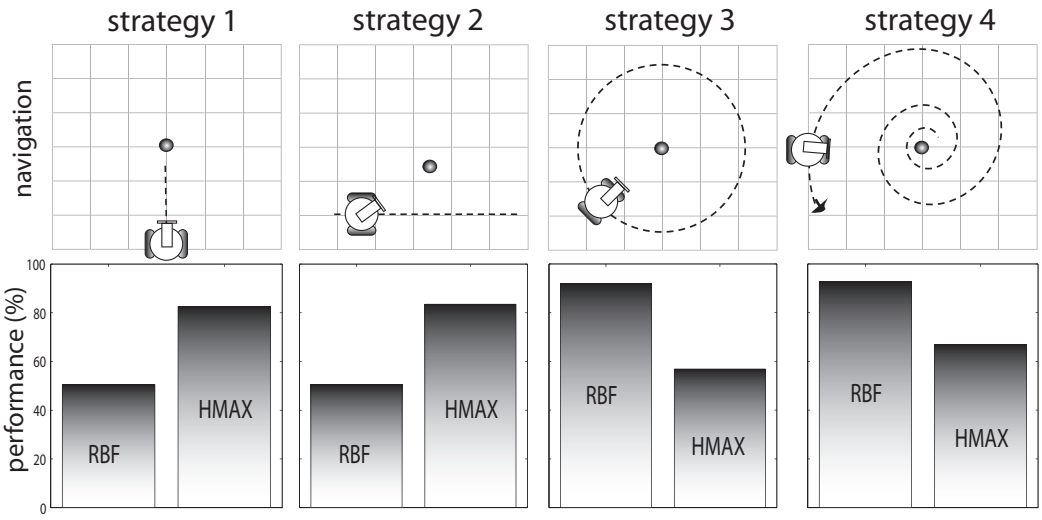


Figure 3: Movement strategies and models' performance. The performance of the models refers to the number of times the model has a correct guess over the test phase. During the following trajectories, the agent takes snapshots at uniform intervals. Strategy 1: the agent approaches the object in a straight line. Strategy 2: the agent passes the object following a straight line. Strategy 3: the agent circles the object with a fixed radius. Strategy 4: the agent spirals the object. The performance of the RBF model increases when the movement strategies allow it to exploit the rotational information during training. In contrast, the HMAX model performance decreases when the model is exposed to multiple rotational views during training in strategies 3 and 4.

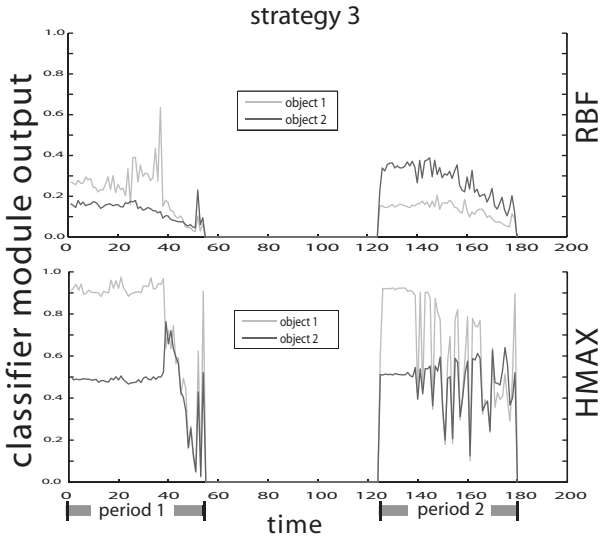


Figure 4: RBF and HMAX models activity during the test phase using strategy 3. When the movement strategy provides multiple points of view during the learning phase, the RBF can have a close match between the training and the test views. In contrast, the HMAX model decreases its discriminability when more points of view are considered. Period 1 represents the time when object 1 is within the visual field. Period 2 is the time when object 2 is within the visual field.

extracted from the visual field) and the training views of the objects (Y axis) at every time step (X axis). Every point in the map has a grey-scale value dependent on the distance between the current view and the training view after processing by the analysis module. The darker a point, the smaller the distance between the views, where distance is the sum of the absolute difference between the views. Each map is divided in two periods which correspond to points where the objects are in the agents' visual field (see figure 2). In the first 55 time steps (period 1), object 1 is present in the visual field and during period 2 (from 125-180), object 2 is in the visual field.

The upper part of figure 5 shows the similarity between views for the RBF, while the lower shows the similarity map for the HMAX model (HMAX views). If a model was responding correctly, we would expect darker areas in the lower region of period 1 and in the upper region of period 2. The similarity map for the RBF has these general features as it has acquired a degree of both rotation and scale invariance from the training trajectory. The responses of the HMAX model however, show that the higher level features extracted for each object are too similar for the two objects to be discriminated reliably.

Thus we see that the performance advantage of the complex HMAX model over the RBF can be achieved by an active vision strategy which uses its motion to provide generalised rotational information. Moreover, we note that HMAX can fail if the views provided to it are too dissimilar.

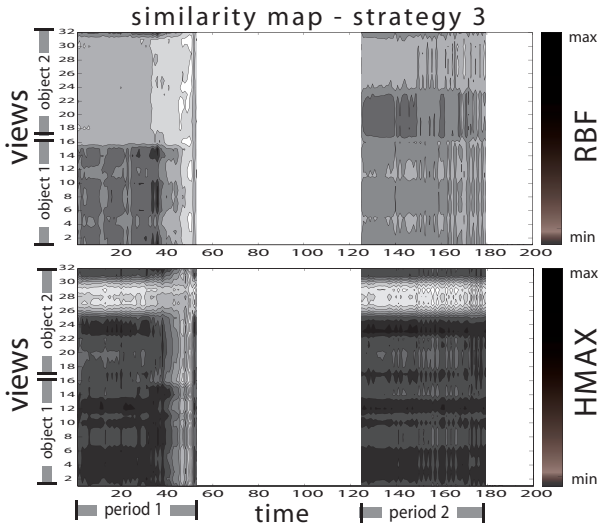


Figure 5: Similarity maps of the models using strategy 3. The darker the regions in each map, the more similar the corresponding views. For the RBF map there is an obvious darker region in the left lower area (corresponding to the views of object 1) for the first period, and a smaller darker region in the right upper area (corresponding to the views of the object 2). In contrast, for the HMAX similarity map dark areas appear during both periods for views associated with both objects.

Experiment 2: RBF Optic flow

Above, we have seen how the RBF exploits multiple viewpoints in training to achieve reliable object discrimination. However, embodied visual systems gather information by moving not only in space (defining the perceived properties of the world) but also in time. In these experiments we explore the role of time dependency in the presentation of the training views during learning. To do this we have assessed the performance of the RBF model when we provided it with optic flow type information (see Methods). The performance of the RBF model with and without optic flow are shown in table 1. The results are broadly similar showing that optic flow information can be exploited by the RBF and provides the same invariances to rotation and scale as when the model was trained on a series of static images (Figure 4).

strategy	non-optic	optic
1	50	51
2	51	56
3	92	73
4	94	95

Table 1: Comparison of the performance (%) of the RBF model with and without optic flow when using the 4 movement strategies. The performance refers to the number of times the models guess correctly over the number of time steps in the test phase.

Time dependency in the recognition process. One of the important properties of optic flow is the time dependency imposed in the recognition process. While the experiment above shows that the model is able to take advantage of differences between successive images, it does not tell us whether it is using this temporal structure. That is, it does not tell us whether the order in which the views are presented during the learning phase is important. To test this, we trained the RBF model using optic flow with views taken using strategy 3 as usual or with the order of training views randomised. To further emphasise the effects of temporal structure the test trajectory used was the same as the training trajectory.

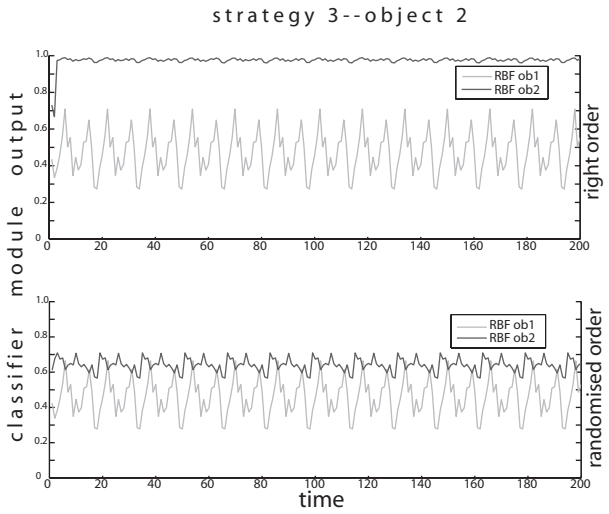


Figure 6: RBF model activity trained using strategy 3 and tested in the same trajectory with randomised ordered training views. (A) model activity for normal conditions (B) model activity of random ordered training views.

Given that object 1 is a rounded object and so appears similar from any perspective while object 2 is more rotationally variable, object 2 was used for this experiment. Figure 6 shows the model activity in using both non-randomized (top) and randomized (bottom) view sequences when circling object 2. While the object can be discriminated in both cases, in the randomised training case outputs from both VTUs are very similar. Results (not shown) confirm that for object 1, variations in the order of the presentation during the training phase are not as relevant as for object 2. These results show that if the same movement strategy is used during training and testing phase, the RBF model with optic flow can exploit the time dependency imposed in the strategy. We next consider what happens when the trajectory is not the same: Is the optic flow-based model robust to changes in the trajectory?

Using a different test trajectory. Robustness in the recognition signals is an important issue when using movement strategies. It is desirable to have some degree of robustness in the movement strategies when testing an object recognition model. In this section, we test the optic flow

strategy in the visual system to certain perturbations in the testing trajectory or in the order of the training views. Initially, to test whether the RBF optic flow changes the activity of the RBF model when having different training and testing trajectories, we used strategies 3 and 4 during the learning phase (we used these strategies because they provide higher rotational and scale variation thus maximizing optic flow), and the testing trajectory during the testing phase (as in Experiment 1). If the order of the presentation of the

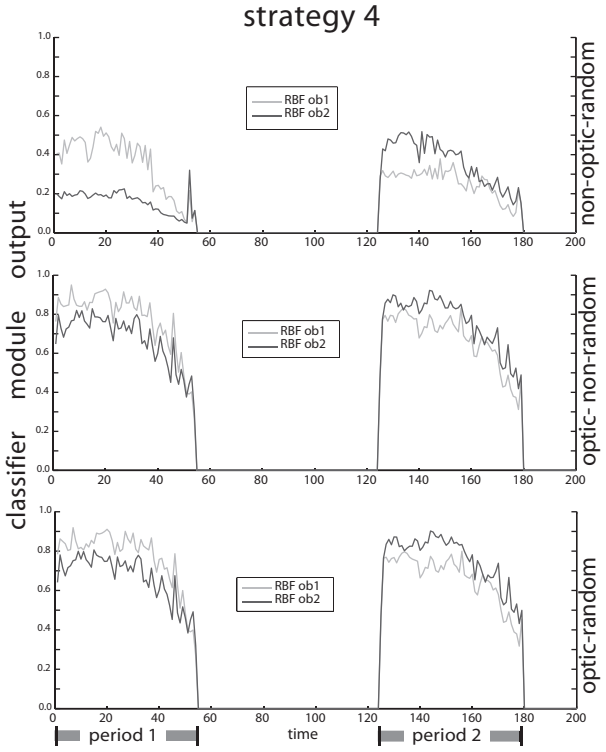


Figure 7: RBF model activity for strategy 4 under various conditions. Top: no optic flow and randomized training view, model activity is the same as in Experiment 1. Middle: optic flow and non-randomized training views. Bottom: optic flow and randomised order of training views.

views during the training phase is important, we expect the model activity to be affected when the order of the training views is randomised. Figure 7 shows model activity in three cases: randomized training views without optic flow (top), non-randomized views with optic flow (middle), and randomized views with optic flow (bottom). In the case of non-optic flow scenario (top), the activation is the same as in Experiment 1 (figure 4) since there is no temporal information present. However, when using optic flow the model activity is not significantly affected by randomizing the training views (compare middle and bottom panels of 7). These results therefore show that the order of the training views does not significantly affect the RBF model activity when using optic flow, and when the training and testing movement strategies are not the same. Thus, in contrast to the previous result (in figure 6), under these conditions the temporal information provided by optic flow is not exploited by the RBF model.

Conclusion

In this paper we have compared the performance of the RBF and HMAX models, on their performance when utilizing embodied movement strategies for training and testing. In the first experiment, four different movement strategies were used to collect the training views and a single, distinct testing strategy was used to assess object recognition performance. Each training strategy offered different degrees of variation in point of view and distance, potentially supporting the development of rotation invariance and scale invariance respectively. When no rotation variance is present in the training views, the HMAX model shows a good performance. However, when more points of view are provided, not only does the RBF model outperform the HMAX model, but the HMAX model performs worse than before. Thus, in what arguably reflects natural viewing conditions, when incorporating variance in both point-of-view and distance, the simple RBF outperforms the more complex HMAX model. In the second experiment, the role of time dependent visual information in the learning process was tested using the RBF model. We found that an RBF model trained on an approximation of optic flow could exploit the temporal information in the difference of consecutive views but only in the restrictive condition in which training and testing trajectories were identical. However, when there are significant differences between training and testing strategies, the RBF model is unable to take advantage of this temporal information. This result suggests that optic-flow style information cannot be assumed to improve visual processing in these conditions and invites further modelling to investigate how such information can best be leveraged by simple models of object recognition.

Our results exemplify the idea that the natural computations underlying adaptive behavior are best understood as being implemented not only in the brain of an organism, but as well in the interactions that cut across brain, body and environment. Improved insights into these natural computations are likely to support the development of enhanced artificial object recognition technologies. This work can be extended in different directions. One is considering more objects in the simulated world. In addition, a study of conditions where temporal information can be exploited to improve object recognition in mobile agents.

Acknowledgements

Edgar Bermudez was funded by the National Council of Science and Technology (CONACyT, Mexico). Andrew Philippides was funded by EPSRC grant GR-T08753-01

References

Aloimonos, Y., editor (1993). *Active Perception*. Erlbaum, Hillsdale, NJ.

Andreasson, H. and Duckett, T. (2003). Object recognition by a mobile robot using omni-directional vision. In *Proc. Eighth Scandinavian Conference on Artificial Intelligence (SCAI 2003)*.

Arbel, T. and Ferrie, F. P. (2001). Entropy-based gaze planning. *Image and Vision Computing*, 19(11):779–786.

Arbel, T. and Ferrie, F. P. (2002). Interactive visual dialog. *Image and Vision Computing*, pages 639–646.

Bermudez, E. and Seth, A. (2007). Simulations of simulations in evolutionary robotics. In Almeida e Costa, F., Mateus Rocha, L., Costa, E., Harvey, I., and Coutinho, A., editors, *Proc. European Conference of Artificial Life (ECAL)*, pages 796–806. Springer-Verlag.

Bermudez-Contreras, E., Buxton, H., and Spier, E. (2007). Attention can improve a simple model for visual object recognition. (In Press) Image and Vision Computing.

Carwright, B. and Collett, T. (1983). Landmark learning in bees: Experiments and models. *Journal of Comparative Physiology*, 151:521–543.

Collett, T. S. and Rees, J. A. (1997). View-based navigation in hymenoptera: multiple strategies of landmark guidance in the approach to a feeder. *Journal of Comparative Physiology A: Neuroethology, Sensory, Neural, and Behavioral Physiology*, 181(1):47–58.

Edelman, S. and Duvdevani-Bar, S. (1997). A model of visual recognition and categorization. *Phil. Trans. R. Soc. Lond. B*, 352(1358):1191–2002.

Gvozdjak, P. and Li, Z. N. (1998). From nomad to explorer: active object recognition on mobile robots.

Howell, J. and Buxton, H. (1995). Receptive fields functions for face recognition. In *Proc. 2nd International Workshop on Parallel Modelling of Neural Operators for Pattern Recognition*, pages 221–226.

Lazebnik, S., Schmid, C., and Ponce, J. (2006). Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. volume 2, pages 2169–2178.

Lehrer, M. and Bianco, G. (2000). The turn-back-and-look behaviour: bee versus robot. *Biological cybernetics*, 83(3):211–229.

Mutch, J. and Lowe, D. G. (2006). Multiclass object recognition with sparse, localized features. volume 1, pages 11–18.

Peters, G. (2000). Theories of three-dimensional object perception: A survey. *Recent research developments in pattern recognition*, 1:179–197.

Pinto, N., Cox, D. D., and Dicarlo, J. J. (2008). Why is real-world visual object recognition hard? *PLoS Computational Biology*, 4(1):151–156.

Poggio, T. and Edelman, S. (1990). A network that learns to recognize 3-d objects. *Nature*, 343:263–266.

Riesenhuber, M. and Poggio, T. (1999). Hierarchical models of object recognition in cortex. *Nature Neuroscience*, 2(11):1019–1025.

Riesenhuber, M. and Poggio, T. (2000). Models of object recognition. *Nature Neuroscience*, 3:1199–1204.

Serre, T., Wolf, L., and Poggio, T. (June, 2005). Object recognition with features inspired by visual cortex. In *Proceedings of 2005 IEEE Computer Society Conference on Computer Vision and Patter Recognition (CVPR)*.

Wang, G., Zhang, Y., and Fei-Fei, L. (2006). Using dependent regions for object categorization in a generative framework. In *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 1597–1604.

Zhang, H., Berg, A. C., Maire, M., and Malik, J. (2006). Svm-knn: Discriminative nearest neighbor classification for visual category recognition. In *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2, pages 2126–2136.

The Emergence of Specialization

Hugues Bersini

IRIDIA-CP 194/6
Université Libre de Bruxelles
50, av. Franklin Roosevelt – 1050 Bruxelles
Belgium - bersini@ulb.ac.be

Abstract

The emergence of specialization remains a true challenge. Suppose a world initially filled with specialized and generalist agents, and let's define these later as able to endorse the various competences characterizing the specialists and to endorse them as well as the specialists. In an evolutionary perspective, it is obvious to see why a specialist will always be less adapted than a generalist, which can indeed alternatively act as many experts. The generalists will meet much more agents and much more situations to which they are adapted to and then cumulate much more payoff (unless the tasks done by generalists are systematically of pitiful quality). This is indeed a paradox to face in order to make sense of a world nonetheless full of specialists. This paper will discuss various ways, beyond the obvious possibility of unfavoring multi-specialization by paying a high cost, to allow specialists to survive the presence of generalists.

Introduction: Specialist vs generalist

We will here define a specialist as an agent which, in contrast to a generalist, can only accomplish a subset of specific tasks. This restriction can be explained by either genetic or phenotypic specificity or any educational or cultural bias. As a direct consequence, a specialist suffers from lack of autonomy; it is never self-sufficient and requires complementary specialists to interact with in order to survive. A first version of this necessary specialists grouping with other complementary specialists is generally designed as "mutualism", for which each specialist contributes by its own competence to the interest of another. It is certainly one possibility among others to understand the "invisible hand" metaphor of Adam Smith, which famously claims that it is not from the benevolence of the butcher, the brewer or the baker, that we expect our dinner, but from their regard to their own interest. The butcher enjoys good bread as much as the baker enjoys good meat. Naturally so, the baker will make the best bread to the advantage of the butcher who reciprocally will prepare the best meat to the benefit of the baker.

Another version or justification for this grouping, different (as figure 1 shows) from "mutualism", is "division of labor", for which specialists join together to the benefit of an external user, outside of the group. Only this external user, exploiting the group of specialists as a whole, can "make sense" of their joining together and rewards back any member of the group in the case it benefits from their collaboration.

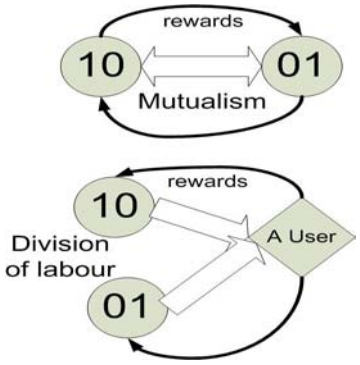


Figure 1: Mutualism vs "division of labor". "10" and "01" are just ways of identifying two specializations.

For specialization to succeed, cooperation is obviously needed, relating the problem of the emergence of specialization with the one of "cooperation", made so popular since the invention of the "prisoner dilemma" (Nowak, 2006). Here we initially see these two problems as orthogonal, assuming that cooperation needs to be resolved before specialization can take place. Without cooperation no specialization turns out to be possible and, indifferently, either generalist or specialist can choose to defect in an interaction. There will be more to say on that precise topic in the conclusions.

A generalist, on the other hand, can stand alone and execute any of the tasks, as a function of which specialist it interacts with. Both in the natural and in the human world, specialization seems to be the common case. Example of "old fashioned" mutualism in the human world could be "husband at work" and "housekeeping wife". Any form of win-win commercial exchange is a successful case of mutualism. Biology is full of example of mutualism, such as associations between plant roots and fungi, with the plant providing carbohydrates to the fungus in return for nitrogenous compounds and water. Other examples are the Lynn Margulis endosymbiotic theory (Margulis, 1998) or the co-virus replication, only possible due to the existence of complementary virions.

Regarding the division of labor, an entrepreneurial team of construction specialists, composed of plumbers, carpenters

and electricians is an alternative to a single handyman. Another nice example of contrast between generalists and a "division of labor" team of specialists is illustrated in figure 2. In nature, examples of "division of labor" are numerous such as in insect societies or in the very well studied biological reality of slime mold moving (Alexopoulos et al, 2004). As will be discussed later, natural selection can favor grouping of complementary specialists to improve the fitness of the whole group.



Figure 2: A generalist vs a divided band of specialists

So either in a case of division of labor or in a mutualistic one, how could specialization be favored in the presence of generalists able to adopt any form of specialization? Obvious solutions are that either it costs too much to be generalist or the way a generalist accomplishes the task is always worse than when done by the agent specialized in that specific task. This is an obvious possibility and does not need any further justification. However, the main motivation of this paper is to propose and discuss further evolutionary roads for specialization to emerge in a world possibly inhabited by generalists.

We will first discard other versions of the problem that this paper is definitely not interested in and then describe the ways the issue will be addressed. The technical road taken to study this problem will be the use of evolutionary games, increasingly popular these days for studying the emergence of cooperation in a "genuinely" and "rationally" competitive world. A computer simulation of a spatial version of evolutionary games will be explained and tested on the problem. We will illustrate the different ways for specialization to emerge, insisting more on the original ones, requiring division of labor and the presence of an environment able of a large scale observation of the group of specialists. We will then conclude by relating this solution with a conceptualization of "emergence" that has been advocated in previous papers.

What the problem is definitely not

An interesting related problem is one of combinatorial optimization, as illustrated in figure 3. It can be enunciated as follows. A group of specialists is available and one needs to find the best sub-group obtained by sampling some of these specialists and grouping them together. A score is associated with each possible sub-group, so that in order to find the best solution (i.e. the best group of specialists) some form of grouping optimization algorithm is required.

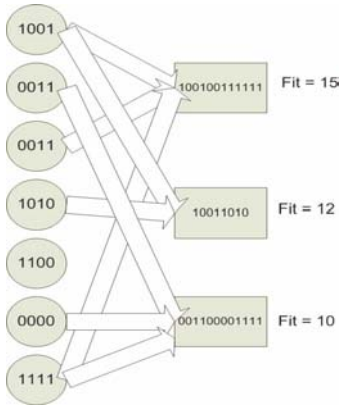


Figure 3: A combinatorial optimization problem to group the specialists in an optimal way.

While it is an interesting problem per se, the problem is definitely not what we are interested in since, in such a case, specialization is already there and does not need any form of justification. The only challenge is to make use of it in the best way but not to question its "raison d'être". However, many engineers remain confronted to this version of the problem and it is quite an interesting one for GA users.

What are the evolutionary driving forces of specialization? An evolutionary game study

The question addressed in this paper is: What can be the evolutionary driving forces for specialization to come out in a world in which generalists could be as likely and perform the task as well as specialists? In other words, what could drive a generalist to become specialized? This question could result to be unexciting because trivial to answer, or tedious, because dealing with a non realistic world (like for instance a world in which agent would have no other choice that popping up endowed with an innate specialization). We believe this not to be the case since both the natural and the human world offer many examples of generalist (stem cell, omnivore, medical generalists and handyman) and many opportunities for these generalists to turn into specialist, provided there is any advantage in doing so. So for what specific reasons could a generalist choose to specialize?

Our favorite way to face the problem is to rely on evolutionary games, pioneered by John Maynard-Smith (Maynard-Smith, 1989) and very actively studied and exploited by researchers such as Martin Nowak (Nowak, 2006). While the Darwinian inspiration of evolutionary games is obvious, these same mathematical and algorithmic tools can help to make sense of the social world in which men tend to imitate the most successful of their neighbors or colleagues.

In what follows, a specialized agent will be defined as a binary string of length n, allowing then 2^n types of specialization. The use of binary strings allows to easily define

complementary specialization, by computing the Hamming distance between two specializations. For instance, the task "00" will perfectly complement the task "11". Additionally, the generalists will be characterized by the presence of the John Holland's don't care symbols into the binary strings (Holland, 1975). For instance, a complete generalist would be "##", while a partial one could be "#1", making thus possible some degree of generalization.

A simple canonical case

Let's focus on the simplest case with n=1 and 3 types of agents: "0", "1" and "#". The evolutionary two players game payoff matrix for such a simple situation could be given as:

	#	0	1
#	b-c	b-c	b-c
0	b-c	0	b
1	b-c	b	0

with $b > c$.

Table 1: The two players payoff matrix

"*b*" is the reward gained by an agent when benefiting from the task done by its complementary partner in the game. Thus when two similar agents meet, they gain nothing: "0". On the other hand, in order to make the appearance of specialists a challenging issue, a cost "c" penalizes a task done by a generalist, so that any agent (either a specialist or a generalist) meeting a generalist for a coupled interaction will receive a "*b-c*" reward. The reasons for the "c" could be twofold: either generalists can do the same task as specialists but with much less competence, or they can perform this task as well but it simply costs to be a generalist capable of so many diverse specializations. When asking to the layman why they do believe that the world is filled with more specialists than generalists, this would generally be the kind of answers they give: being a generalist means a lack of competence or it is simply much harder to achieve.

Many other matrices are possible, with easier to justify and in general more sophisticated reward values attributed to each of the nine possible entries. But for mathematical reasons to follow, this elementary matrix is enough to convey the sole effect of the cost "c" on the success or disappearance of generalists.

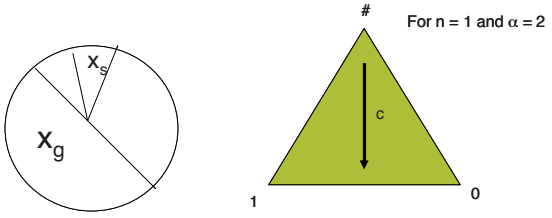


Figure 4: The basis of the mathematical analysis and its result illustrated for $\alpha = 2$ and $n = 1$.

Suppose a population of agents decomposed in the following way (as illustrated in figure 4), a part x_g of generalists and α subgroups of complementary specialists x_s. The time evolution of generalists and specialists can be mathematically described as follows, by supposing that any specialist can only interact with its unique complementary specialists (and we suppose a same concentration x_s for any group of specialists), and generalists can interact with all generalists and all specialists.

$$\dot{x}_s = x_s((b-c)x_g + bx_s - \phi)$$
$$\dot{x}_g = x_g((b-c)(x_g + \alpha x_s) - \phi)$$
$$x_g + \alpha x_s = 1$$

After some easy manipulations:

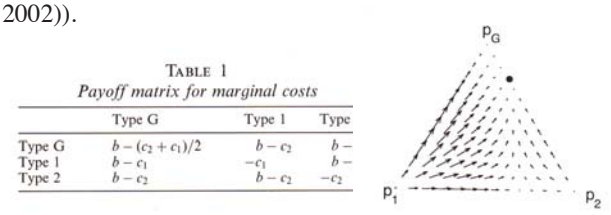
$$\dot{x}_s = (1 - \alpha x_s)x_s^2((b(1-\alpha) + c\alpha)$$

So that the two possible solutions are:

$x_s = 0$ *or* $x_s = 1/\alpha$ depending on the value of c as compared to $b(\alpha\text{-}1)/\alpha$.

We clearly understand here the two main reasons for the dominance of generalists over specialists: the cost c and the frequency α. The bigger the cost the harder for the generalist to survive and the steady-state solution in such a case turns out to be a subdivision of the population in α subgroups of complementary specialists (illustrated in fig. 4 for two groups of complementary specialists). The smaller the specialist frequency of encounters (i.e. the higher the "α") the easier it is for generalists to replace specialists, since specialists have fewer agents to interact with and fewer opportunities of reward.

Such a simple mathematical analysis is the reason behind the simplicity of the payoff matrix as compared with much more sophisticated versions of it (Wahl, 2002, D'Orazzio and White, 2006) (as shown in figure 5 extracted from (Wahl, 2002)).


TABLE 1
Payoff matrix for marginal costs

	Type G	Type 1	Type
Type G	$b-(c_2+c_1)/2$	$b-c_2$	$b-$
Type 1	$b-c_1$	$-c_1$	$b-$
Type 2	$b-c_2$	$b-c_2$	$-c_2$

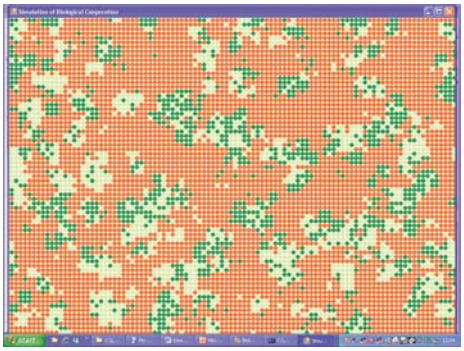
Figure 5: A finer mathematical analysis of the evolutionary game between two groups of specialists and generalists.

Nevertheless, as a matter of fact, the main qualitative results remain unchanged through all the various versions of the problem. The only way to make specialists survive the dominance of generalists is either to increase the cost of generalization or to augment the possibility of specialist opportunities to cumulate payoff, here the number of complementary specialists they can interact with.

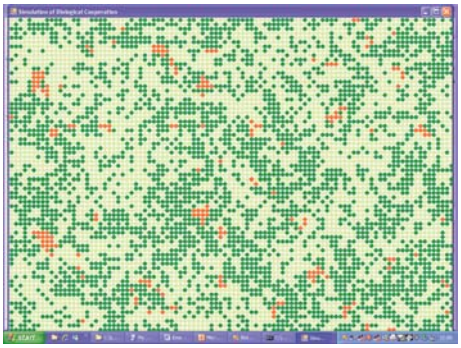
The spatial cellular automata simulation

Again largely inspired by Nowak's spatial version of the prisoner dilemma (Nowak, 2006), we propose a simple cellular automata type of simulation in which every cell contains one agent of length n, either specialist (without "#") or partially or completely generalist. The spatial environment is a 2-D toroidal cellular automata in which every agent/cell can interact with its 8 neighbors (generally called a "Moore neighborhood").

The simulation goes as follows. At every time step, each agent will interact with its 8 neighbors and cumulate the payoff perceived according to the two players payoff matrix. Then asynchronously (by means of a random selection iteratively performed over all agents a number of times equal to the number of cells), each agent is both selected and "reproduced" by being replaced by the best agent in its neighborhood. Thus, at each time step, all agents compute their payoff and reproduce according to their fitness value, the best agents locally invading the neighborhood. Two results are shown in figure 6, for n= 1 (so three types of agents: 1, 0 and #). The first result is shown for a low value of c (generalists win) and the second for a high value of c (the two groups of specialists win and equally divide the population).



In this snapshot of the simulation, the generalists are invading the whole population (the large clusters of homogeneous cells – the generalists – are percolating through the small two-color clusters – the specialists).



In this snapshot of the simulation, the specialists are invading the whole population. The two colors represent the two complementary groups of specialists: "1" and "0".

Figure 6: Results of the spatial simulation: above for a low cost of "generalization" and below for a high cost.

These simulations just reproduce the results anticipated by the mathematical analysis. The outcome of the simulation is binary and depends on the value of the cost, below or above the threshold. Above specialists win, below generalists win.

The question addressed in the following is: "Is there any less trivial driving factor than just the cost of generalization?" According to the mathematical analysis, another important contributing factor, beyond the cost, is the frequency of encounters. This frequency depends on the number of opportunities that a specialist has for cumulating reward, but that generalists miss for one reason or another. So is there any other original way for specialists to increase this frequency of encounters that would however escape the generalists? We admit to enter from now on in the realm of speculation.

Division of labor

In order to make a real profit from their specialization, specialists should better cluster together in the hope of creating by such grouping more opportunities of payoff. This is exactly the message provided by the idea of "division of labor" which definitely needs to be distinguished from the simple mutualism.

The following possibility will be added in the spatial simulation namely that complementary specialists can cluster together when they are neighbors. At every time step, before any reward gain and before any reproduction, a specialist is allowed to connect to at maximum two of these complementary neighbors, like illustrated in figure 7. Two is the minimal number for making possible groups of complementary specialists to appear.

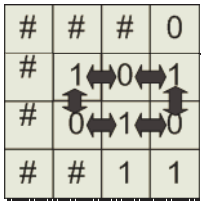


Figure 7: A possible cluster of complementary specialists

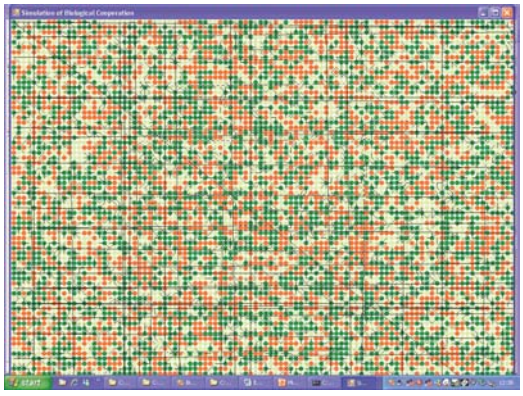


Figure 8: A snapshot of the simulation where the black lines reflect the clustering among the specialists.

The following snapshot of the simulation (figure 8) shows the presence of black lines which testify for the connections among specialists only. The clusters just survive one time step namely the time for the agents to interact (in drawing some payoff opportunity of their presence in the cluster) and then reproduce. Once reproduced, an agent looses its membership to any cluster and its connectivity is simply reset to null. The payoff gained by a specialist agent will be equally shared among all the members of the cluster it belongs to. The simulation at every time step thus takes place in three successive steps: 1) cluster, 2) get payoff by interacting and 3) reproduce.

An obvious concern could come from the impossibility for generalists to cluster. Why should it be so? Are we not resolving the problem in a too much ad hoc way? A simple reason is that, by clustering with, for instance, a specialist, a generalist would loose its capacity to adopt any other form of specialization since it would be forced to take one specific profile. It would sacrifice its "chameleon" side. A human generalist, once convinced of the advantages in being so, would decline any opportunity to freeze himself in one of his multiple possible profiles. In a more primitive world, we will take for granted that a non-specialized organism cannot connect with another non-specialist or with a specialist, since the pattern recognition ability which allows two specialists to recognize each other and to connect could be absent from the generalists. But remember that these are all speculations, wished for discovering more opportunities for a specialized world to replace one populated mainly by generalists.

If nothing is modified in the simulation and the specialist and generalists just interact as before, the results won't be affected by this added clustering possibility. The gain of the clustering and thus the increase opportunity of frequency is perfectly compensated by the sharing of the benefits. For instance, in figure 7, any one of the six agents composing the cluster will take six times more benefit, nevertheless always divided by six. Of course, increasing the benefit of the specialist with respect to the generalist (for instance, by not dividing the benefit by six and just distributing the whole "b" to any one) would favor specialization. However, such a move is far to be satisfactory since this benefit has to be equally shared through the cluster in a way or another. Additionally, this would reduce the whole problem to that same question of cost that we want to avoid. Is there any further way for specialists to increase their frequency of encounters which could make sense while giving them a natural advantage over generalists?

Simulation with varying length of agents

Suppose now a new version of the simulation in which agents can be of any length (with n the maximum length): "1", "00", "##1"… Suppose further that any agent of length x can only interact with and take profit from agent of equal or inferior length: x, x-1, x-2, … (Just as if it would swallow it). So at any time step, an agent looks in its neighborhood to discover equal or smaller agents. It thus computes its total profit only with this restricted part of its neighborhood. As a consequence, the payoff matrix is no more symmetrical, since an interaction between agent x and agent x-1 (if both specialists and complementary) will reward "b" to agent x but 0 to agent x-1. The complementarity between two agents is simply assessed by comparing the first common bits. An agent can be of length n in two different ways: either it is naturally of length n (from birth) or it takes part of a cluster of dimension n. Finally, only agents of length 1 can cluster (remember that, among them, only specialist can cluster) and reproduce.

This makes obviously a lot of assumptions, leaving a strong impression of arbitrariness. Generalists cannot cluster and only the smallest agents can reproduce. These last limitations are simply there to make easier to grasp the results of the simulation, which could similarly happen for less restrictive conditions (for instance bigger agents, due to their size, could simply be slower to reproduce). The simulations are performed with a low value of the cost c, which has always favored generalists in all previous cases. The results of the simulations are shown both for a maximum length of 2 and 3 in the two following plots of figure 9.

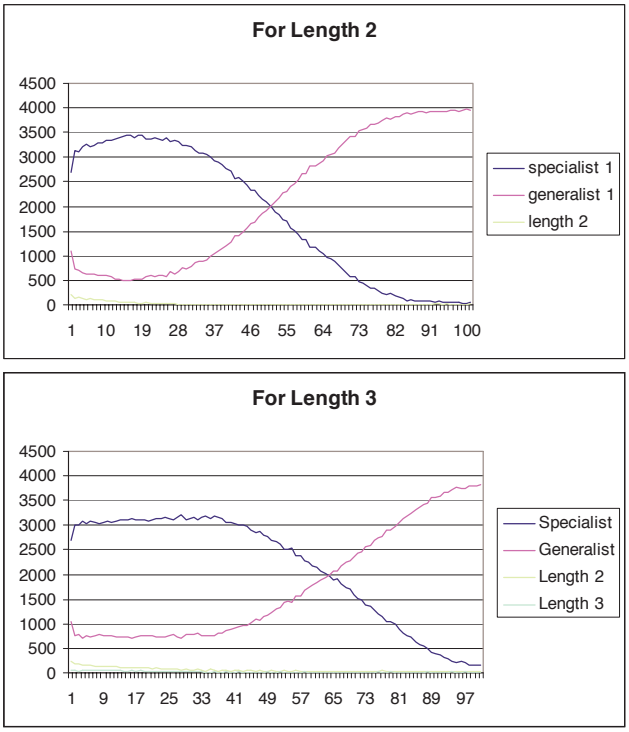


Figure 9: Results of the simulations for maximum length of 2 and then 3 and a low cost of generalization. It is important to look also at the decreasing of the curves at the bottom of the two plots, showing the decreasing concentration of agents of length 2 and 3. The life time of specialization increases with the length of the simulation.

These plots show the cumulated concentration of specialists of length 1, the concentration of generalists of length 1 and the concentration of agents of length 2 and 3. One can observe that during the first time steps, the number of specialists grow in size and easily win the game, the duration of this winning period depending on the agent maximum length: the greater the longer. The reason is easy to understand. Initially, the

specialists, by clustering together, can access to sources of reward that are simply inaccessible to generalists (remember that the generalist cannot cluster so that they can only interact with agents of length 1). For instance, a cluster of two or more specialists can interact with agents of length 2, an impossible source of payoff for generalists.

Then, since the big agents cannot reproduce, they are overcome by the specialists until the instant where only agents of length 1 remain in the simulation. From this instant on, the simulation proceeds as usual (the small value of the cost favors generalization) with the generalists invading the world by benefiting from much more sources of rewards than the specialists. So besides the obvious possibility to excel in the task, another possibility for specialists to impose themselves is to create communities of complementary members which, as a whole, can achieve new functionalities much beyond the possibility of any isolate generalist. Here this is what happens, since the bigger agents in the simulation can be exploited by clusters of specialists. The generalists simply can't see them.

Emergence and conclusions

We were interested in this paper in investigating other evolutionary driving forces, beyond the simple cost, in order to favor the survival of specialists to the expense of generalists. Our mathematical analysis shows that, whatever formulation given to that problem, these forces will always have to do with either the cost (that we discard here because too obvious and commonly accepted) or the frequency of encounters. This last effect is more original since it might give rise to ways for specialist to create opportunities of rewards that escape generalists. In substance, this is what we have attempted to do through our last simulations.

Suppose for instance that in order to construct your house, you can choose between a handyman capable of optimally achieving all parts of the construction alone: bricklaying, carpentry, electricity and plumbing, and an entrepreneurial team composed of as many specialists as required. Whatever the quality of the handyman work, there is clearly one thing he will never be able to achieve alone: work on all parts in parallel, something obviously possible for a team. Therefore a group of specialists is able to work in parallel and achieve the same construction in a much shorter time, whatever the quality of each single part of the construction. This is a simple human example of a clustering benefit inaccessible to a single generalist. When adopting the "division of labor" perspective and by joining together, specialists can discover a whole new set of opportunities that a generalist (which has other good reasons to stay alone) can't even see. Specialists, by grouping together, access to opportunities that generalists simply miss.

As regards the problem of "cooperation vs defection", in a classical world where mutual defections turn out to be the only Nash equilibrium, this "joining together" might provide an extra road for cooperation to emerge. Only by connecting together, can the specialists achieve some reward. So defectors, just like generalists, by refusing to be part of the groups, would disappear to the advantage of cooperators. We further suppose here that all members of the group have to do their own, so that groups allowing for the integration of defectors will never succeed in accomplishing the task. In a mutualistic situation, defection wins over cooperation (as defectors don't give but just receive), while, in a division of labor situation, cooperation has obvious advantages (defectors don't receive any payoff).

In a very stimulating book of Matt Riddley (Riddley, 1998) entitled "The origins of virtue", in the chapter entitled "Division of labor", we can find these excerpts:

> "*In a phrase, therefore, the advantage of society to me is the division of labor. It is specialization that makes human society greater than the sum of its parts… It is this synergy between specialists that makes human societies tick, and it is this that distinguishes us from all other social creature … Adam Smith was the first to recognize that the division of labor is what makes human society more than the sum of its parts… As division of labor between specialists evolves, integration into higher unit systems also advances, and, as social homeostasis evolves, the individual human loses some portion of his self-regulation and becomes more dependent for his existence upon the division of labor and the integration of the social system*".

The repetition of the famous expression "The whole is more than the sum of its parts" relates the concept of "division of labor" with the concept of "emergence". In previous publications (Bersini, 2004; Bersini and Philemotte, 2006), I have discussed this later concept and the necessary three ingredients which together allow a collective phenomenon to be described as "emergent". First the phenomenon, as usual, requires a group of agents entering in a non-linear relationship and entailing the existence of two semantic descriptions depending on the scale of observation: micro or macro. Second, the macro phenomenon (the one that raises philosophical debate) has to be observed and "objectivised" by a mechanical observer, which has the natural capacity for temporal and/or spatial integration. This mechanical observer positively substitutes for the human one, which is blameworthy of endowing "emergence" with an unacceptable dose of subjectivism. Finally, for this natural observer to detect and to select the collective phenomenon, it needs to do so in rewards of the adaptive value this phenomenon is responsible for. Basically, physics can simply explain the proximate causes of the phenomenon while natural selection provides the "adaptive", "engineering", "ultimate" causes of the functionality. The presence of natural selection brings me to defend in these previous papers the idea that emergent phenomena can only belong to biology.

In the European Swarm-bots project, which is being coordinated in our laboratory (Gross et al. 2006), largely inspired by the capacity of some insect species (such as ants) to assemble in order to accomplish tasks that none of them, alone, is able to accomplish, small robots connect together to do as well. For instance, two robots join together in order to pass over a gap that would make any of them fall down if trying alone. I claim that, in the case of real insects, "passing over that gap" is an emergent behavior, since it requires a cluster of agents, an external observer integrating the agent's behavior in space and time (the gap here plays the role of this

required natural observer) and natural selection to favor the agents able to pass over the gap.

In the last simulations presented in this paper, again the same three ingredients are present: the agents cluster together; once they do cluster they can exist for and be observed by the "bigger agents", and once they do exploit these resources, they are favorably selected. Consequently, one extra road for both cooperation and specialization to appear, beyond the quality of the task or the cost of generalization, might really be any new opportunity made possible by groups of specialized agents which can do more than their parts. Specialization really emerges in a genuine sense once specialized agents decide to enter, not in a mutualistic relationship, but in a genuine strategy of "division of labor", so as to make it relevant for an external observer: human or biological, whose presence justifies the fitness increase of the specialists. This jump in fitness provided by the grouping of single specialized agents into new types of organisms is one possible way to construe the concept of the major transitions in evolution (Maynard-Smith and Szathmary, 2005).

References

Alexopolous, C.J., Charles W. Mims, M. Blackwell et al (2004)., *Introductory Mycology, 4th ed*. John Wiley and Sons, Hoboken NJ.

Bersini, H. (2004). Whatever emerges should be intrinsically useful. In *Artificial life 9*, pages 226--231. The MIT Press, 2004.

Bersini, H. and Philemotte, C. (2006). Emergent phenomena only belong to biology. In Proceedings of the 9h European Conference on Artificial Life.

D'orazzio, A.E. and White, T.A. (2006). Incomplete Division of labor. Error-prone multitaskers coexist with specialists . J. Theoret. Biol. 2007

Groß R., Bonani M., Mondada F., Dorigo M. 2006. Autonomous Self-Assembly in Swarm-Bots. *in IEEE Transactions on Robotics*, volume 22(6).

Holland, John H (1975), *Adaptation in Natural and Artificial Systems*, University of Michigan Press, Ann Arbor

Margulis, Lynn, (1998). *Symbiotic Planet : A New Look at Evolution*, Basic Books

Maynard Smith, J. (1989) *Evolutionary Genetics*. Oxford: Oxford University Press.

Maynard Smith, J. and E. Szathmary (1995). *The Major Transitions in Evolution*, Freeman press, Oxford.

Nowak, MA (2006). *Evolutionary Dynamics*. Cambridge, MA: Harvard University Press.

Riddley, M. (1998). *The origins of virtue*. Penguin (Non-Classics);

Wahl, L.M., 2002a. Evolving the division of labor: generalists, specialists and task allocation. *J. Theoret. Biol.* 219, 371-388

Autocatalytic Replication of Polymers Revisited

Ben Blundell[1]

[1]Evolutionary and Adaptive Systems
University of Sussex, Brighton, BN1 9RH, UK
ben@benblundell.com

Abstract

A simple computational model for the emergence of autocatalytic sets as described in (Farmer et al., 1986) is re-implemented. Results are found to generally agree with the major theme in the original work: increasing the initial polymer variety in a toy chemical soup scenario increases the likelihood that a complex autocatalytic set will suddenly bootstrap itself into existence. Quantitatively, however, critical probabilities derived from this careful re-implementation are very much higher than those reported in the original work. A full resolution is not reached, but a theoretical argument supports the simulation results gained in this instance.

Introduction

The principle of an autocatalytic set, a set of molecules which collectively catalyses its own production, holds intuitive interest. There exists obvious relations to primitive metabolic systems, and contemporary minimal definitions of life such as autopoiesis (McMullin, 1999).

By achieving catalytic closure, a set of relatively inert molecules can organise into a self-sustaining identity, a persistent presence in a chemical soup.

Different questions can be asked of autocatalytic sets. In general, one might be interested in (1) *how a set came to be* and the preconditions necessary for it's emergence, (2) *which critical molecular species the set consists of*, or (3) *how the set chemically operates in real physical space and time*.

Original work on autocatalytic sets (Kauffman, 1986; Farmer et al., 1986) pursued the first question as the main point of interest, although all questions are inter-related to some extent. Question 2 has recently been given a deep formal treatment (Hordijk and Steel, 2004). Question 3 is of considerable depth and of most contemporary interest, involving concepts such as dynamics, spatial compartmentalisation, reaction kinetics and concentrations in particular physical autocatalytic instantiations (see, for example (Ono and Ikegami, 2000) for application in a spatial abstract cell model).

This paper describes a careful re-implementation of the original (graph theoretic) model investigating the inevitable emergence of complex autocatalytic sets (Farmer et al., 1986). Section 1 recaps the motivations and assumptions of the original model. Section 2 describes in detail the re-implementation carried out. Remaining sections present and discuss the results, which generally follow the same qualitative pattern as original results, but differ by a factor some 100 in quantitative predictions of the critical probability of autocatalysis.

1 Original Work

The original work on autocatalytic sets by Stuart Kauffman is concerned with making a tentative link to the grand problem of the origin of life itself (Kauffman, 1986, 1993) and levelling a respectable argument against entrenched explanations of template based replication.

In the original model (Farmer et al., 1986), the emergence of autocatalytic sets is investigated as a connectivity feature of directed graphs.

A reaction graph captures the core chemical relationships in a system of polymers, expressing the *reaction possibilities* in that system. Operational details such as space, time and quantity are not represented in this canonical description. The chemical system is assumed to exist in a well-stirred overflowing reactor environment.

The central idea is built upon the *phase transition* phenomena in connectivity problems. As systems become increasingly connected a critical limit is reached when, very suddenly, each component of the system is connected directly or indirectly to every other. A large component crystallises from a mass of independent sub-systems.

By the same logic, when a reaction network is expressed as a reaction graph, there must exist some critical catalytic connectivity beyond which each polymer will directly or indirectly catalyse every other - at which point the existence of a complex autocatalytic set can be inferred with almost certainty.

The original model focuses on finding this critical connectivity. A basic reaction system - where polymers consist of directional strings of characters - is successively grown from an original 'firing disk' (food set). In this scenario,

the reaction system is always autocatalytic in a strict sense, but criticality is judged when the rate of change of polymer species becomes exponential (autocatalytic networks which continually create large complex proteins were of prime interest to the authors).

Significant assumptions of the model include the prerequisite of flow reactor conditions and the assumption that the distribution of catalytic capacities in peptide space can be modelled by a fixed probability *P* that any one polymer will catalyse any other.

Farmer et al. find that that the critical value of *P* required for an autocatalytic set decreases as the initial polymer variation in the system increases, lending support to their general autocatalytic account for the origin of life from a sufficiently diverse pre-biotic chemical soup.

2 Re-implementation

Original Graph Growth Algorithm

For clarity, the original graph growth algorithm is presented below. Square braces represent cross-references to the more detailed implementation to follow.

> *Our rule for random assignment of reactions is implemented as follows: For a given starting list of molecular species, we compute the maximum number of allowed condensation and cleavage reactions by counting the number of distinguishable combinations of string concatenations and string cleavages* **[see Note 2]**. *The number of reactions that we actually assign is obtained by multiplying the number of allowed reactions of each type by a probability P. To assign condensation reactions, we chose two molecules at random* **[Box 1]**, *while for cleavage reactions we chose a molecule and a cleavage point at random* **[Box 2]**. *In both cases enzymes are chosen at random from the set of species currently present.*
>
> *Assignment of reactions can be viewed as a dynamical process. We initialise the system by choosing a starting list, called the "firing disk", typically chosen to be all possible strings shorter than a given length L* **[Iteration 0, Step 1]**. *Reactions within the firing disk are assigned as described above* **[Iteration 0, Step 2]**. *Condensation reactions may generate new species outside the firing disk, thereby expanding the list* **[Iteration 0, Steps 3 and 4]**. *The introduction of new species creates new reaction possibilities; to take these into account, on the next time step we count the number of combinatorial possibilities involving the new species* **[Iterations 1 to 1000, Steps 2 and 3]**. *Multiplying by P gives the number of new reactions* **[Iterations 1 to 1000, Steps 6,7,8,9]**. *This process is repeated on subsequent time steps. As long as new species are created on each step the graph continues to grow; otherwise growth stops.* (Farmer et al., 1986), p. 54

Graph Growth Algorithm as Implemented

Definitions

S Set of all distinct polymer species currently in the system. Initially empty.

N Set of distinct polymer species, new on the current iteration. Initially empty.

s_n Size of set S. Number of distinct polymer species in the system.

n_n Size of set N. Number of new distinct polymer species on current iteration.

B Alphabet size of polymers.

M Order of initial firing disk (or 'maximum sized polymer' in firing disk, also referred to as L_f elsewhere in this paper).

P Probability that a random polymer catalyses an arbitrary reaction.

Iteration 0

1. Seed firing disk. Make set S contain all possible polymers of alphabet size B up to length M. S will contain a total of $s_n = \sum_{L=1}^{M} B^L$ polymers.
2. Calculate the number of *distinct* condensation reactions *possible* in the firing disk, $R^*_{cond} = s_n \times s_n$. (See Note 2 below).
3. Calculate a number of random condensation reactions to *assign*, $R^+_{cond} = P \times R^*_{cond}$.
4. Assign R^+_{cond} condensation reactions to the firing disk as in Box 1 below, thereby expanding the disk. Cleavage reactions need not be assigned here. The graph currently consists of all possible polymers up to size M, and thus cleavage reactions cannot introduce any new species at this stage.

Box 1: condensation reaction assignment

1. Pick a random polymer of sequence a from S
2. Pick another random polymer of sequence b from S ($a = b$ is allowed)
3. Concatenate a and b to create polymer sequence $p = a+b$
4. If $p \notin S$ then add p to N (the set of new polymers created)
5. Otherwise, disregard reaction. The condensation does not create a new species and thus is of no significance. Another reaction is not assigned in place.

Iterations 1 to 1000

1. Record n_n, the number of new species for this iteration.
2. Calculate the number of *possible* cleavage reactions that the set of new polymers N introduces to the system. $R^*_{cleav} = \sum_{L=2}^{J}[N(L) \times (L-1)]$, where J is the maximum length polymer in set N, and $N(L)$ is the total number of polymers of size L in N. (See Note 2 below).
3. Calculate the number of *possible* condensation reactions that the set of new polymers N introduces to the system. $R^*_{cond} = n_n n_n + 2n_n s_n$ (See Note 2 below).
4. Add the new polymers N to the total species set S, so S becomes $S \cup N$.
5. Set $N = \phi$
6. Calculate a number of random condensation reactions to assign for this iteration, $R^+_{cond} = P \times R^*_{cond}$.
7. Calculate a number of random cleavage reactions to assign for this iteration, $R^+_{cleav} = P \times R^*_{cleav}$.
8. Assign R^+_{cond} condensation reactions as in Box 1 above.
9. Assign R^+_{cleav} cleavage reactions as in Box 2 below.
10. Goto step 1. Perform next iteration.

Box 2: cleavage reaction assignment

1. Pick a random polymer of sequence p from S
2. If the length of $p < 2$, disregard.
3. Pick a random break point on p, splitting it into two fragment substrate sequences $p = a + b$
4. If $a \notin S$ then add a to N, otherwise disregard
5. If $b \notin S$ then add b to N, otherwise disregard

Determining Criticality of Graph Graph is SUPRA critical if, during the 1000 graph iterations,

1. The number of polymer species in the system, $n_s > 10^5$
2. A polymer in the system exceeds a length of 1000
3. The new condensation reactions *possible* on any iteration, $R^*_{cond} > 2 \times 10^9$
4. The new condensation reactions *assigned* on any iteration $R^+_{cond} > 5 \times 10^5$

If 1000 iterations are completed, the graph is still judged SUPRA critical if $n_s > 5s_f$, i.e. if the number of species in the system after the last iteration are five times greater the number of species in the firing disk, s_f.

Otherwise, the graph is judged as SUB critical.

Note 1: Distinct Reactions In a fairly common sense way, this study regards two reactions $a + b \rightleftharpoons c$ and $x + y \rightleftharpoons z$ as *distinct* if their substrates do not match. That is to say, they are only the *same* reaction if $a = x$ and $b = y$ (and thus $c = z$).

Two reactions may of course have the *same product* (e.g $aaa + a \rightleftharpoons aaaa$ and $aa + aa \rightleftharpoons aaaa$) and still remain distinct. If two reactions have *different products*, the reactions will *certainly* be distinct.

Note 2: Counting Distinct Condensation and Cleavage Reactions Although the original description does not specify exactly, this paper calculates the number of new condensation and cleavage reaction possibilities at each iteration in a straightforward way.

The number of *new possible cleavage reactions* introduced into the system is simply the sum of the number of ways the new species can be broken apart. Each new species, by definition, is a product not encountered before. By Note 1 above, breaking this new product on any of it's bonds will in turn reveal a substrate *combination* not encountered before, and thus a distinct reaction.

The number of *new possible condensation reactions* introduced into the system by the new species can be calculated in two parts. Firstly, each new species polymer can be combined in two ways with every member of the existing species (by concatenating to the left and right hand side of the existing species). By Note 1 above, both of these condensation reactions are distinct, new reactions, because either the left or right hand substrate is a new species. New species thus make possible at total of $2n_n s_n$ new condensation reactions with the existing species.

Secondly, the new species can be combined amongst themselves. Each new species polymer can be appended to the left or right hand side of every other new species polymer, including itself. However, because *both* substrates in these reactions are new species, there will be some double counting.

For example, new species g may be combined with new species h by left concatenation $g + h$ or by right concatenation $h + g$, yielding two distinct concatenations. When h is considered and combined with g to the right and left the mirror is true, yielding two non-distinct concatenations. The distinct concatenations are thus just the set of left-hand concatenations between the new species, a total of $n_n n_n$ reactions.

Estimation of P_{crit} algorithm

As in the original paper, estimation of the critical probability of catalysis, P_{crit}, is performed by using a simple trial-and-error algorithm.

For a graph of alphabet size B and firing disk order M, 10 independent estimates of P_{crit} are made as in Box 3 below and then averaged to provide a more reliable result.

Box 3: To estimate P_{crit}

1. Set P_{min} to a low probability, known to cause the graph to go firmly SUB critical.
2. Set P_{max} to be a higher probability known to cause the graph to go firmly SUPRA critical.
3. Add a little random noise to P_{max}.
4. Perform 15 iterations of gradient descent, where on each iteration
   (a) Set P to the value in-between P_{min} and P_{max}
   (b) Grow the graph at P three times
   (c) If the graph goes SUPRA critical at least 2 out of 3 times, set $P_{max} = P$
   (d) If the graph goes SUB critical at least 2 out of 3 times, set $P_{min} = P$

Notes:

- After 15 iterations, the value of P is found to be sufficiently converged to the critical probability.
- P_{min} and P_{max} are initially set to be fairly close to the critical probability in order that the 15 iterations of gradient descent may be usefully spent.
- The noise initially introduced to P_{max} introduces some variability into the 'halving' of P.

Computational Considerations

For a quality source of random numbers, an implementation of the *Mersenne Twister* random number generator (Matsumoto and Nishimura, 1998) was used. The random number generator was re-seeded on at the beginning of each graph growth run. In this way, no two simulation sessions used the same sequence of random numbers.

3 Results and Discussion

Figure 1 directly compares main results from this study (blue lines) to the main results in the original paper (red lines). Both data sets are in qualitative agreement insofar as the *central idea* goes. The downward trend of lines indicates that increasing the size of the firing disk, or increasing the alphabet size of the polymers lowers the critical probability that a complex autocatalytic set (supra-critical graph) will spontaneously emerge (in a well-stirred environment).

The main point of departure from the original results lies in the actual values of critical probability. Values gained in this study are typically two factors of ten *higher* than those gained in the original experiment. To validate their results, Farmer et al. provide a theoretical estimation of P_{crit} for a chemistry with alphabet size $B = 2$. However, the mathematical derivation is largely unexplained and hard to follow, and thus of limited insightful use.

To support results gained here, it is clear that a critical probability for autocatalysis P_{crit} has to satisfy the greater of the two following conditions:

1. Firstly, as a bare minimum, the value of P_{crit} must be able to create at least one new species outside the firing disk. In a firing disk of s_f species, there exist ${s_f}^2$ viable condensation reactions, and so it follows that $P_{crit} > \frac{1}{{s_f}^2}$ in order to catalyse at least one of these.
2. Furthermore, the value of P_{crit} must be set such that at least one new species *continues* to be created at each iteration. The graph must exhibit continual growth.

At this stage, it must be noted that the values of P_{crit} reported in the original results do not satisfy the first condition. In the simplest case, for example, when the firing disk consists of the two monomers a and b, the critical probability is cited to be less than 10^{-1}. However, with four condensation reactions aa, ab, ba, bb being initially viable in this scenario, this probability would assign around 0.4 reactions on iteration 1, which would be computationally truncated to 0. The graph would stop growing immediately, and would have no chance of being critical.

Additionally, the value of P_{crit} for condition 2 is often higher than that for condition 1. As condition 2 concerns the time behaviour of the graph leading to it's eventual fate, it can be calculated by equating the growth of the graph to the evolution of a discrete dynamical system (and finding the bifurcation point in that system).

The formulation of this dynamical system is possible providing that the following assumptions are made about the graph growth procedure described in Section 2:

1. Cleavage reactions can be ignored. (This assumption is feasible since, at each graph iteration, the number of viable cleavage reactions is very much smaller than the number of viable condensation reactions. Furthermore (through general simulation observations), of the cleavage reactions assigned, even fewer produce new polymer species outside the current system).
2. Every condensation reaction assigned creates a new species previously not in the system.

Assuming all assigned condensation reactions produce new polymer species leads to the "luckiest case" of the reaction graph, which is in fact the situation desired, whereby the graph grows at the *absolute minimum value* of P possible.

The dynamical system describing the graph growth, then, starts at iteration 0 with the total number of species in the firing disk and the total number of condensation reactions these species will have assigned amongst them:

$$n_s = s_f = \sum_{L=1}^{M} B^L$$

$$n_n = trunc\left[P{n_s}^2\right]$$

where $trunc$ denotes a truncation to the nearest integer (polymers only exist as wholes).

At each successive iteration, the total number of polymer species n_s is equal to the total number of species at the beginning of the last iteration, plus the number of new species assigned on the last iteration

$$n_s(t+1) = n_s(t) + n_n(t)$$

and the number of new species n_n is equal to the number of new condensation reactions viable on the last iteration multiplied by P (since we are assuming every condensation reaction assigned creates a new species):

$$n_n(t+1) = trunc\left[P(n_n(t)^2 + 2n_s(t)n_n(t))\right]$$

The dynamical system is parameterised by the alphabet size B, the initial order of the firing disk M and the probability of catalysis, P. Increasing the parameter P over a critical value causes a bifurcation in the dynamics. Below this bifurcation point, the reaction graph goes sub-critical and settles to a fixed point whereby the number of new polymers n_n per iteration become 0, and the total number of species n_s rest at an arbitrary value. Above the bifurcation point, the reaction graph grows exponentially. The bifurcation point thus corresponds to the phase transition in this scenario, and the value of P at which it occurs is the critical probability P_{crit}.

Figure 2 shows the nature of the bifurcation point for the dynamical system described above with firing disk $B = 2$, $M = 6$. More importantly, Figure 3 shows that the theoretical bifurcation points for different alphabet sizes and firing disk orders correspond more or less precisely to the results derived from the simulation in Section 2 (even though the simulation *does* allow cleavage reactions and *does not* require condensation reactions to necessarily produce new species).

Despite the described differences in critical probability estimates to original work, reaction graph growth curves derived from this work (Figures 4-6) tally fairly well with those presented for the original model ((Farmer et al., 1986), p55, Fig 2) both in terms of form and numerical axis values.

Figure 4 shows that a low probability of catalysis leads the reaction graph (which has an initial firing disk of size $L_f = 6$ and alphabet $B = 2$) to decay until there is no further growth. By contrast, increasing the probability of catalysis past the critical threshold leads to supra-critical growth

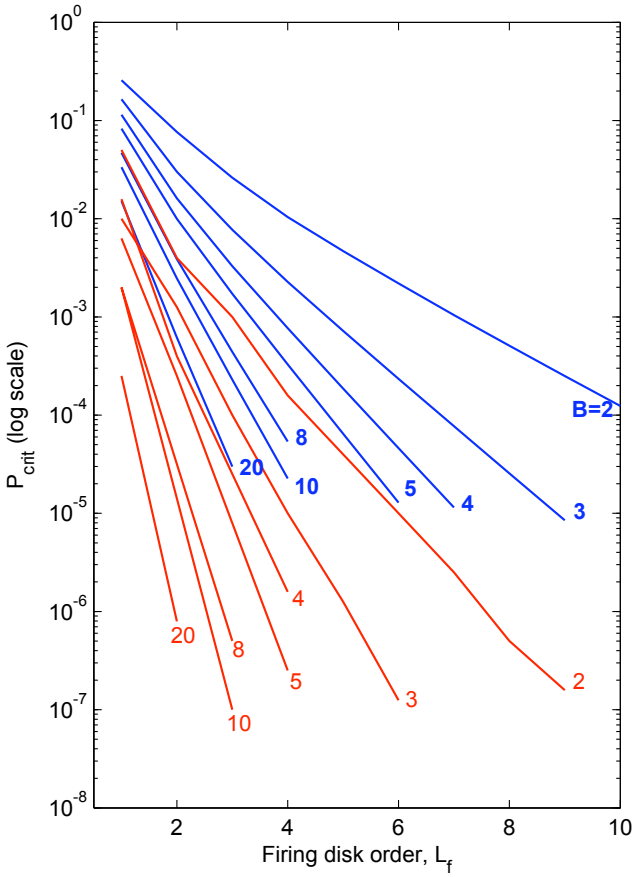


Figure 1: Main comparison of results [blue lines] with those obtained by (Farmer et al., 1986) [red lines]. Graph shows how critical probability of catalysis P_{crit} scales with order of firing disk L_f for different alphabet sizes (labels on lines). Red lines should only be viewed as an estimate of original data.

(Figure 6) where a small initial decay is followed by exponential growth without bound.

Right on the critical threshold, the reaction graph was found to be incredibly fickle, sometimes turning sub-critical, and sometimes supra-critical (Figure 7). Figure 5 was obtained by running the reaction system with firing disk $B = 2$, $L_f = 6$ over many trials at the critical threshold $P_{crit} = 0.002194$ and recording the longest instance of an eventually supra-critical graph. In this study, reaction graphs surviving for any length of time at the critical threshold were delayed supra-critical graphs where fortuitous assignment of condensation reactions meant that only a single new species would be assigned at each iteration (the other condensation reactions being assigned to produce polymers already in the system). This marks another minor departure from the original work where the number of new species per iteration is reported to be erratic at criticality.

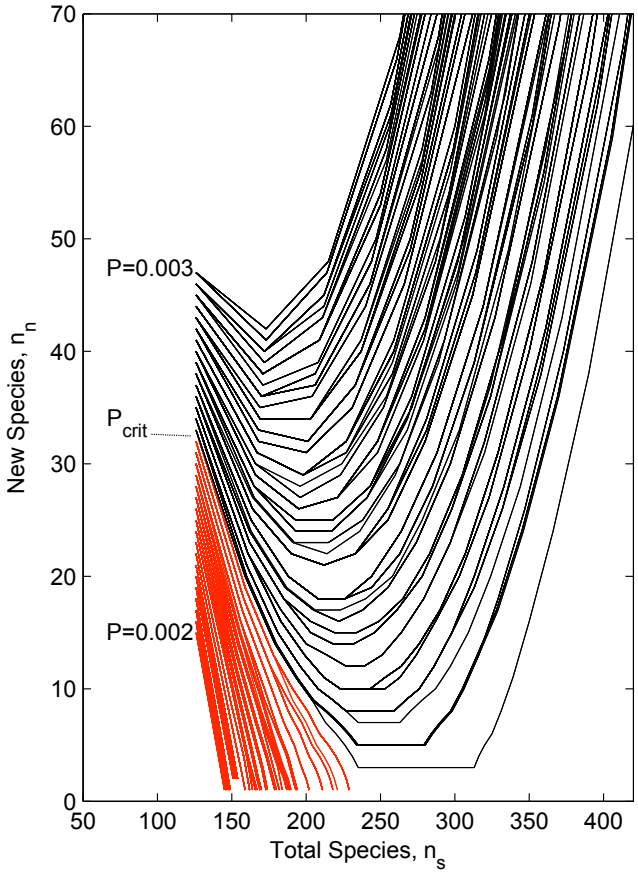


Figure 2: Overlay of phase portraits for the graph growth dynamical system with firing disk $B = 2$, $L_f = 6$. Each trajectory corresponds to the system portrait at a different value of the parameter P. When P approaches P_{crit}, the dynamical system bifurcates and instead of settling to a fixed point (corresponding to a sub-critical graph - red lines), the system spirals to infinity (corresponding to a supra-critical graph - black lines).

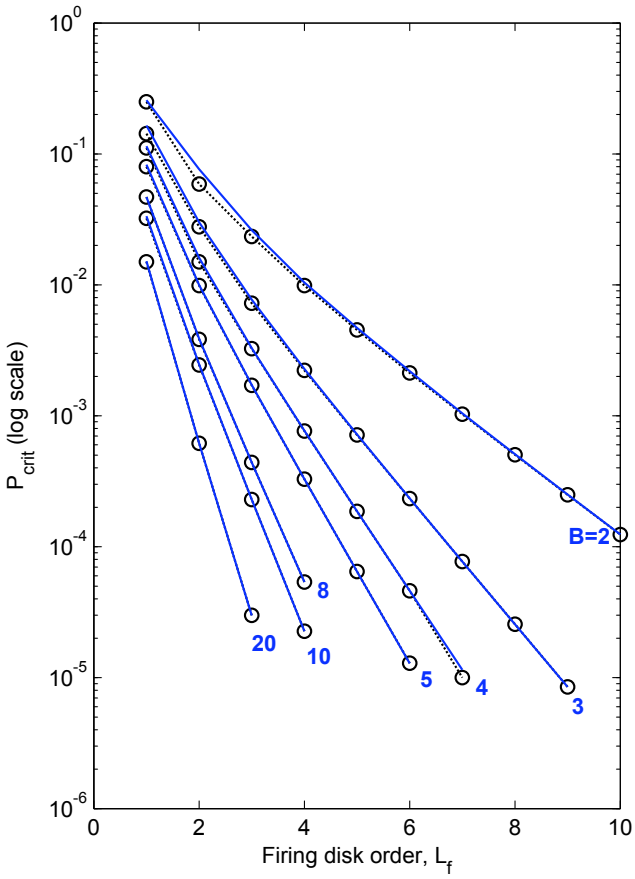


Figure 3: Similarity between theoretical and simulation results. Bifurcation values of P for the graph growth dynamical system (black dotted lines with circle markers) coincide nearly exactly with results obtained in the re-implemented simulation (blue lines) - so much so that the black dotted lines are often obscured on this plot.

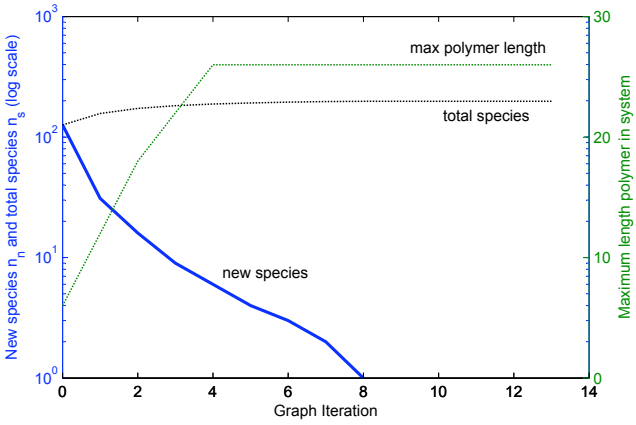


Figure 4: $B = 2$, $L_f = 6$, $P = 0.002000$. Sub-critical. Graph decays. After 8 iterations, graph stops growing.

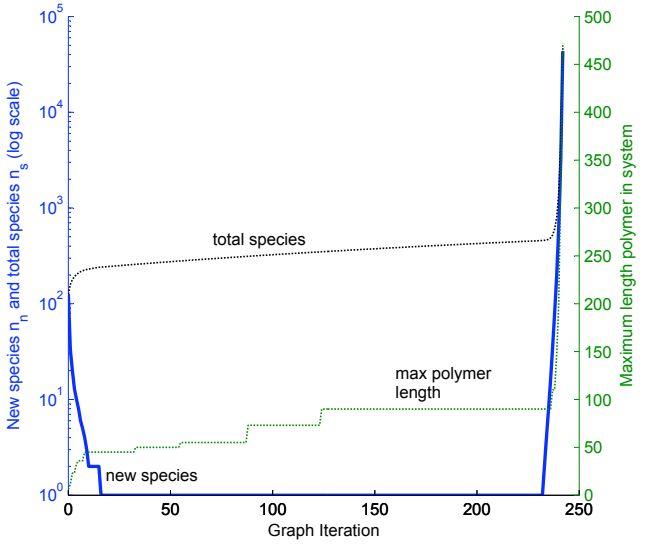


Figure 5: $B = 2$, $L_f = 6$, $P = 0.002194$. "Critical". Graph initially decays to a steady growth rate of 1 polymer per graph iteration (on the line of the x-axis) until an eventual explosion happens around iteration 230.

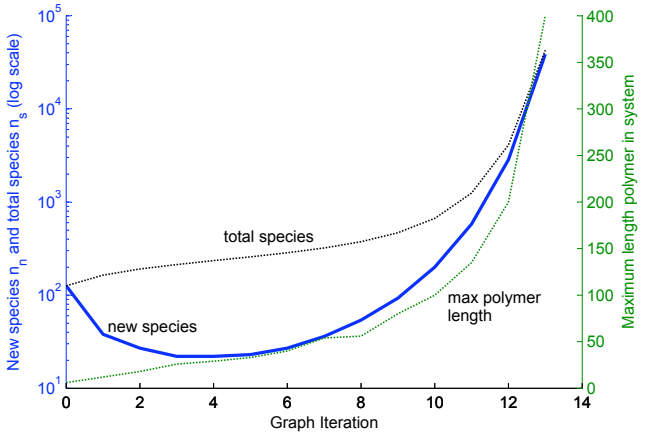


Figure 6: $B = 2, L_f = 6, P = 0.0025000$. Supra-critical. The graph initially decays, but quickly recovers and then snowballs.

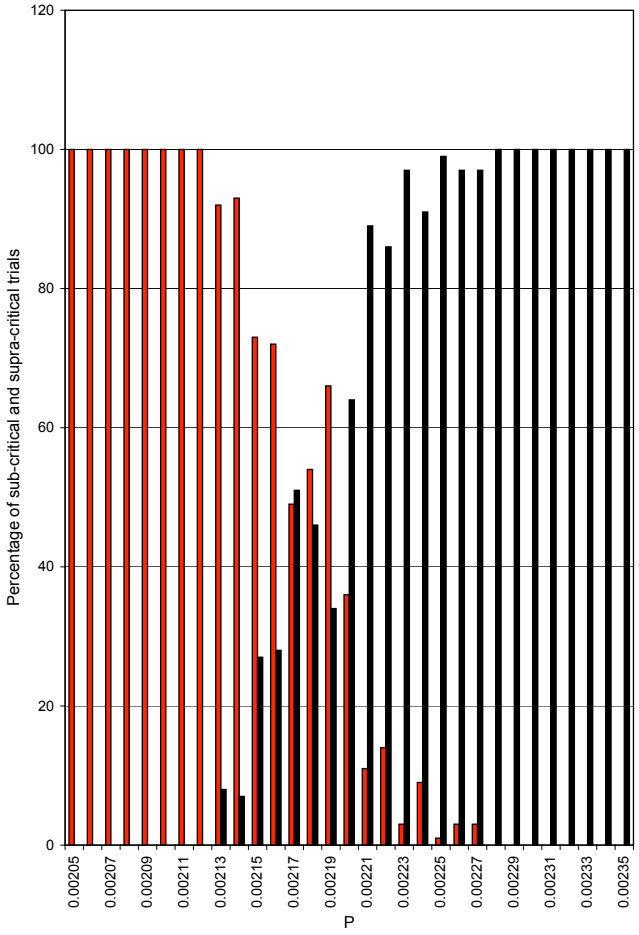


Figure 7: Characterising the phase transition for firing disk $B = 2, L_f = 6$ in terms of trial frequency. The reaction graph was grown 100 times for all values of P between 0.00205 to 0.00235 with interval 0.00001. Red bars represent how many trials went sub-critical, and black bars represent how many trials went supra-critical. The phase transition is clearly visible as a disjoint region separating two regions of stable sub and supra-critical behaviour.

4 Conclusions

This work has sought to conduct a careful re-implementation of one of the first models investigating network autocatalysis (Farmer et al., 1986). Results here follow the same qualitative pattern, and thus do not invalidate the general abstract aims of the original work, but they do present a (fairly major) quantitative discrepancy to the critical catalytic probabilities reported from the original model.

Such discrepancies probably do not hold significant connotations for subsequent work published in the last 20 years, since the spirit of the Farmer et. al. study is one of proving a very general point, but nevertheless they would be nice to resolve. Indeed, the introduction of the original work states that many of the results should be experimentally testable.

In theoretical support of critical catalytic probabilities presented here, a simple discrete dynamical systems model is proposed as an approximation to the more involved reaction graph growth algorithm. The critical catalytic probabilities of the original work can be seen as too low to produce bifurcations in this dynamical model, whereas the bifurcations correspond more or less exactly to the simulation results of this study.

The source of the discrepancy is not ultimately resolved, but it seems that the most outstanding grey area lies with the calculation of the number of new condensation and cleavage reactions at every iteration. However, even with no explicit details mentioned in the original publication, there is little room for manoeuvre, and this study implements a straightforward common-sense interpretation.

Specifics aside, it is worth finally noting that in the last two decades, models relating to the origin of life have gained (considerably) in fidelity from pure abstract autocatalytic notions. Whilst autocatalytic sets remain an important cornerstone, one branch of enquiry for instance ((Mavelli and Ruiz-Mirazo, 2007), (Ruiz-Mirazo and Mavelli, 2008)) focuses on the origins of minimal cells in terms of how active self-producing 'proto-cellular' systems could have started to couple internal chemical reactions to membrane processes. Such efforts are beginning to address the deeper issues raised in Question 3 of the Introduction.

References

Farmer, J., Kauffman, S., and Packard, N. (1986). Autocatalytic replication of polymers. *Physica D*, 22:50–67.

Hordijk, W. and Steel, M. (2004). Detecting autocatalytic, self-sustaining sets in chemical reaction systems. *J. theor. Biol.*, 227:451–461.

Kauffman, S. (1986). Autocatalytic sets of proteins. *J. theor. Biol.*, 119:1–24.

Kauffman, S. (1993). *The Origins of Order: Self-Organization and Selection in Evolution*. Oxford University Press, USA.

Matsumoto, M. and Nishimura, T. (1998). Mersenne twister: A 623-dimensionally equidistributed uniform pseudorandom number generator. *ACM Trans. on Modeling and Computer Simulation*, 8:3–30.

Mavelli, F. and Ruiz-Mirazo, K. (2007). Stochastic simulations of minimal self-reproducing cellular systems. *Phil. Trans. R. Soc. B.*, 362:1789–1802.

McMullin, B. (1999). Some remarks on autocatalysis and autopoiesis. Presented at the workshop: Closure: Emergent Organizations and their Dynamics, May 3-5, 1999, University of Ghent, Belgium.

Ono, N. and Ikegami, T. (2000). Self-maintenance and self-reproduction in an abstract cell model. *J. theor. Biol.*, 206:243–253.

Ruiz-Mirazo, K. and Mavelli, F. (2008). On the way towards 'basic autonomous agents': Stochastic simulations of minimal lipid-peptide cells. *BioSystems*, 91(2):374–387.

Behavior Chaining: Incremental Behavior Integration for Evolutionary Robotics

Josh Bongard
University of Vermont, Burlington, VT 05405
josh.bongard@uvm.edu

Abstract

One of the open problems in autonomous robotics is how to consistently and scalably integrate new behaviors into a robot with an existing behavioral repertoire. In this work a new technique called behavior chaining is introduced, which allows for gradually expanding the behavioral repertoire of a dynamically behaving robot. The approach relies heavily on scaffolding: gradually restructuring the robot's environment such that selection pressure favors the incorporation of a new behavior. This method teaches a robot a compound behavior not yet reported in the literature: dynamic legged locomotion toward an object followed by grasping, lifting and holding of that object in a physically-realistic three-dimensional environment. The method assumes that success is dependent on the order in which behaviors are learned. This is justified by results which show that if a robot is forced to learn lifting first and then incorporate locomotion, it eventually succeeds at both more often than a robot forced to learn locomotion first and then lifting.

Introduction

Useful autonomous robots must exhibit three properties: they must be able to perform behaviors autonomously; they must be able to adapt an existing behavior on the fly in the face of unexpected situations; and they must be able to exhibit different behaviors in different circumstances. Recent work has reported an autonomous physical robot capable of the former and middle property: maintaining a behavior in the face of unexpected body damage (Bongard et al. (2006)) using automated modeling (Bongard and Lipson (2007)). In this work a virtual robot is introduced that exhibits the former and latter property: it is able to autonomously learn one behavior (lifting) and then integrate a second one (dynamic locomotion) into its repertoire.

Evolutionary robotics (Harvey et al. (1997); Nolfi and Floreano (2000)) is an established technique for generating robot behaviors that are difficult to derive analytically from the robot's mechanics and task environment. In particular, such techniques are useful for realizing dynamic behaviors (eg. Reil and Husbands (2002); Hornby et al. (2005)) in which individual motor commands combine in a nonlinear fashion to produce behavior, thereby making analytical derivations of optimal controllers infeasible. However, evolutionary approaches to dynamic behavior generation have focussed up until now on realizing a single behavior, such as locomotion (Reil and Husbands (2002); Hornby et al. (2005)) or grasping (Fernandez and Walker (1999); Chella et al. (2007)). Alternatively, multiple non-dynamic behaviors have been generated for simpler wheeled robots (Nolfi (1997); Lee et al. (1998)).

The approach described here is a type of robot shaping technique (Singh (1992), Dorigo and Colombetti (1994) and Saksida et al. (1997)) in which the organization of the learning or evolution process is guided manually or automatically. However, in behavior chaining it is assumed that there is an underlying order in which behaviors should be learned, and that this order is dictated more by the agent's morphology, controller, task environment and controller optimization process than it is by the agent's current behavioral competency.

Although it is possible to realize multiple behaviors in a robot by gradually incorporating more modules into its controller (Brooks (1986); Calabretta et al. (2000)), this approach does not scale well. A scalable approach to behavioral flexibility should allow the same dynamic controller to exhibit multiple attractor states, in which individual behaviors correspond to individual attractor states, an idea that is gaining currency in the robotics literature (Inamura et al. (2004); Okada and Nakamura (2004)). One of the main difficulties in this approach however is realizing multistability (Foss et al. (1997)) in the controller: it should settle into different attractor states that correspond to the different desired behaviors in the face of the appropriate sensory stimulation.

The approach to realizing multistable controllers described here relies on scaffolding (Wood et al. (1976)), a concept borrowed from developmental psychology: easing the learning agent's task environment at the outset to allow initial learning, and then gradually removing

the constraints to stimulate further learning. The minimal cognition approach (Beer (1996)) has led to agents capable of multiple dynamic behaviors such as legged locomotion and visually-guided orientation (Gallagher and Beer (1999)), but this integration was achieved by awarding the agent for demonstrating both capabilities simultaneously, and without the aid of scaffolding. The work here suggests that it may be easier to evolve a controller that generates one behavior first through scaffolding, and then incorporate additional behaviors by reducing the scaffolding. Further, it is shown here that to realize an agent capable of multiple behaviors some behaviors may have to be learned before others: this suggests that learning multiple behaviors at once may not be scalable, although more investigation into this issue is warranted. Scaffolding has been used with some success in the robotics literature for realizing single behaviors rather than sequences of dynamic behaviors (Pratt et al. (2001); Reil and Husbands (2002); Lungarella et al. (2003); Ziemke et al. (2004)). Alternatively, a teacher may lead a robot through a series of behaviors directly, after which the robot learns to reproduce those behaviors autonomously (Saunders et al. (2007)), but in this approach the exact motions comprising the behaviors must be demonstrated and therefore known *a priori* by the teacher.

In the work presented here we introduce a dynamic scaffolding method that enables a virtual autonomous robot to first learn one dynamic behavior (lifting) and then gradually incorporate a second dynamic behavior (locomotion) using a single monolithic controller. In the next section the method is introduced; the following section reports results demonstrating how this behavioral competency arises; and the final section provides some discussion and concluding remarks.

Methods

In this section the virtual robot is first introduced, followed by its controller. The section concludes with a description of behavior chaining, the dynamic scaffolding method that enables the robot to gradually incorporate new behaviors into its repertoire.

The robot In this work a virtual quadrupedal robot is used (Fig. 1). The robot is comprised of four legs and a front gripper. The legs are comprised of an upper and lower cylinder. The gripper is composed of a small spherical claw base, which connects the main body to the claw pincers. The claw base can be rotated upward relative to the main body, and both the left and right pincers are comprised of a claw arm (proximal to the claw base) and claw tip (distal to the claw base). The robot attempts to grasp and lift a rectangular target object that is placed at varying distances from the front of the robot's body. The physical specifications of the body parts and the joints connecting them together are given in Table 1.

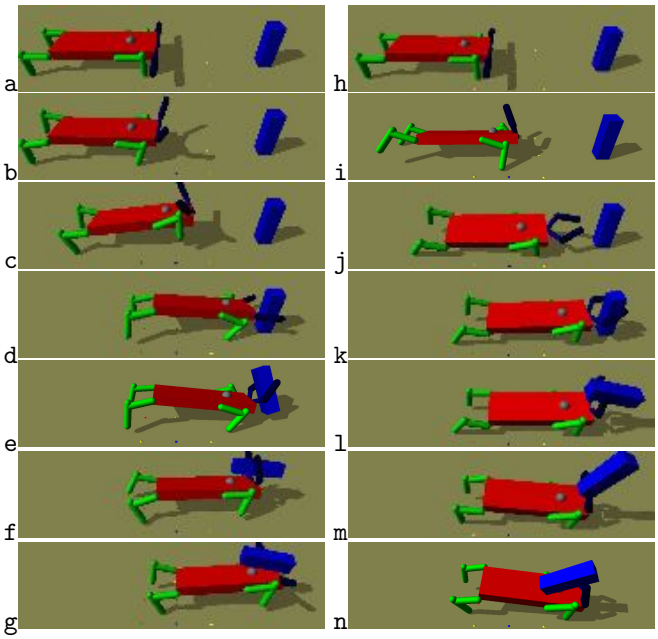


Figure 1: Two evolved behaviors. **a-g**: The robot moves toward the target object placed 2.8 meters ahead (**a-d**), lifts it (**e,f**), and drops it onto its back (**g**). **h-n**: The robot moves toward the target object placed 3.0 meters ahead (**h-k**) and swings it (**l,m**) onto its back (**n**).

Part	length	width	height	mass
Target object[T]	0.4m	0.4	1.4	1kg
Main body[MB]	3.0	1.0	0.3	1
Upper leg[UL]	0.7	0.1*		1
Lower leg[LL]	0.7	0.1*		1
Claw base[CB]	0.1*			0.25
Claw arm[CA]	0.8	0.1*		0.25
Claw tip[CT]	0.8	0.1*		0.25
Joint	min	max	orientation	
[MB][UL]	-20^o	20	sagittal	
[UL][LL]	-20	20	sagittal	
[MB][CB]	-1	120	sagittal	
[CB][CA]	-45	0	frontal	
[CA][CT]	-75	0	frontal	

Table 1: Physical parameters of the robot and environment. *=radius

Eight motors actuate the four upper and lower legs, another motor actuates the claw base, and four motors actuate the base and distal parts of the left and right claw pincers, for a total of 13 motors. A touch sensor and distance sensor reside in both the left and right claw tips, a rotation sensor resides in the claw base, and a distance sensor resides on the robot's back (gray sphere

in Fig. 1), for a total of six sensors. The touch sensors return a value of 1 when the corresponding body part touches another object, and zero otherwise. The distance sensors return a value commensurate with the sensor's distance from the target object: they return zero if they are greater than five meters from the object; and a value near one when touching the object. Object occlusion is not simulated here; the object can be considered to be emitting a sound, and the distance sensors respond commensurately to volume.

The controller A continuous time recurrent neural network (Beer (2006)) is used to control the robot. The CTRNN is composed of 11 motor neurons (the two claw arm motors share the same motor neuron, as do the two claw tip motors to ensure the claw closes symmetrically). The remaining 10 motors each receive commands from their own motor neuron. The value of each motor neuron is updated according to

$$\tau_i y_i' = -y_i + \sum_{j=1}^{10} w_{ji}\sigma(y_j - \theta_i) + \sum_{j=1}^{6} n_{ji}s_j \qquad (1)$$

where τ_i is the time constant associated with neuron i, y_i is the value of neuron i, $\sigma(x) = 1/(1+e^{-x})$ is an activation function that brings the value of neuron i back into $[0,1]$, w_{ji} is the weight of the synapse connecting neuron j to neuron i, θ_i is the bias of neuron i, n_{ji} is the weight of the synapse connecting sensor j to neuron i, and s_j is the value of sensor j.

In this formulation, each sensor may have a direct effect on every motor neuron. However this effect may be minimized or eliminated by low values for n, or by behaviors that cause a target motor neuron to reach minimum or maximum values.

The virtual robot with a given CTRNN is evaluated over a set number of simulation steps in a physical simulator[1]. At the outset of each step, the sensor values are retrieved from the physical simulator and the values of the motor neurons are calculated. The resulting values are scaled to the minimum and maximum rotation angles of the corresponding joint (Table 1), forming the desired angle for that joint. Torque is then applied to the joint commensurate with the difference between the joint's current angle and the desired angle. The positions and velocities of the objects in the simulation are then updated using a step size of 0.005; the CTRNN is updated once for each time step.

Behavior Chaining Behavior chaining is a method for dynamically tuning the robot's task environment to facilitate learning which assumes that the order in which behaviors are learned affects the probability of success. The algorithm is outlined in Fig. 2. A

[1]Open Dynamics Engine, `www.opende.com`

```
1.  BehaviorChaining()
2.     Create and evaluate random parent p
3.     WHILE ~Done()
4.        Create child c from p, and evaluate
5.        IF Fitness(c) ≥ Fitness(p) [see eqns. 2,3]
6.           p = c
7.        IF Failure()
8.           EaseEnvironment()
9.           Re-evaluate p
10.       WHILE Success(p)
11.          HardenEnvironment()
12.          Re-evaluate p
13. Done()
14.    18 hours of CPU time have elapsed
15. Failure()
16.    100 generations since last success
17. EaseEnvironment()
18.    EvaluationTime ← EvaluationTime+100
19. Success(g)
20.    ∃k, k ∈ {1,...,t} |
21.     T(LeftClawTip, k)&
22.     T(RightClawTip, k)&
23.     D(SensorNode, k) ≥ 0.825
24. HardenEnvironment()
25.    TargetDistance ← TargetDistance+0.01m
```

Figure 2: **Behavior chaining pseudocode.** The algorithm executes a hillclimber [1-14]. If the current genome fails [15,16], the task environment is eased [17,18]; while it is successful [19-23], the task environment is made more difficult [24,25]. $T(x,k)$ returns 1 if body part x is in contact with another object and zero otherwise at time step k. $D(x,k)$ returns the distance of body part x from the target object at time step k.

random CTRNN is created by choosing all τ from the range $[0.1, 0.5]$, all w from $[-16, 16]$, all θ from $[-1, 1]$, and all n from $[-16, 16]$. This gives a total of $10 + 10 * 10 + 10 + 6 * 10 = 180$ evolvable parameters. The robot is then equipped with this controller and allowed to behave in the task environment for 100 time steps, where the target object is placed directly in front of the robot. After evaluation the fitness of the controller is computed as

$$f = \max_{k=1}^{t}(D(\text{LeftClawTip}, k) * D(\text{RightClawTip}, k)) \quad (2)$$

if the touch sensors in the left and right claw tip fail to fire at the same time during any time step of the evaluation period, and

$$f = 1 + \max_{k=1}^{t}(D(\text{SensorNode}, k)) \qquad (3)$$

otherwise, where t is the evaluation time, and $D(x,k)$ indicates the distance of body part x from the target

object at time step k. Eqn. 2 rewards controllers for steering the robot toward the target object. Eqn. 3 rewards controllers for also lifting the target object onto the robot's back (where the sensor node is located) after it has touched the target object with both claw tips.

A hill climber (Russell and Norvig (1995)) is used to optimize the initial random CTRNN against this fitness function (Fig. 2[1-12]). Although a more sophisticated optimization process such as a genetic algorithm could be used, hill climbing was found to be sufficient in this case. At each generation a child CTRNN is created from the current best CTRNN and mutated (Fig. 2[4]). Mutation involves considering each τ, w, θ and n value in the child, and replacing it with a random value in its range with a probability of $10/180 = 0.0556$. This ensures that on average, 10 mutations are incorporated into the child according to a normal distribution. It was found that for lower mutation rates runs tended to become mired in local optima. If the fitness of the child CTRNN is equal to or greater than the fitness of the current best CTRNN, it is replaced by the child; otherwise, the child is discarded (Fig. 2[5,6]).

After each possible replacement, the current CTRNN is considered in order to determine whether a failure condition has occurred, or whether it has achieved the success criteria. In the present work the failure condition is defined as 100 generations of the hill climber elapsing before a successful CTRNN is found. A successful CTRNN is defined as one for which, at some time step during the current evaluation (Fig. 2[20]) both claw tips touch the target object (Fig. 2[21,22]) and it is lifted far enough onto the robot's back such that the distance sensor there fires above a certain threshold (Fig. 2[23]).

If the failure condition occurs, the task environment is eased; if the current CTRNN succeeds, the task environment is made more difficult (Fig. 2[7-12]). Easing the task environment involves increasing the current evaluation period by 10 time steps. This has the effect of giving the robot more time to succeed at the current task if it fails. Making the task environment more difficult involves moving the target object 0.01 meters away from the front of the robot. This has the effect of teaching the robot to grasp and lift the target object when it is close, and learning to locomote toward the target object, followed by grasping and lifting it, when it is placed further away. As some CTRNNs that succeeded for a given target object distance also succeed when the object is moved further away, the object is continually moved until the current CTRNN fails, at which time hill climbing recommences (Fig. 2[10-12]). In order to further speed the algorithm an individual evaluation is terminated early if the robot ceases to move before succeeding at the task.

The overall success of a run is indicated by how many times a successful genome was found. That is, how far away the target object has been moved while still preserving a controller that can guide the robot to the object and also enable successful grasping and lifting.

Results

A series of independent **runs** were conducted and are reported on here, where each run is conducted for 18 hours of CPU time. One set of runs were performed using the quadruped robot described above, and are henceforth referred to as **regime I**. Another set of runs were performed in which two additional, middle legs were added to the quadruped, resulting in a hexapod, referred to as **regime II**. The new legs are the same size and have the same orientation as the front legs. The CTRNN for the hexapod requires an additional four motor neurons: two for the middle upper legs and two for the middle lower legs. As all motor neurons are connected to one another, and the sensors are also connected to the additional motor neurons, this gives a total of $14 + 14 + 14 * 14 + 6 * 14 = 308$ evolvable parameters.

Within both regimes, a series of seven **trials** were performed: in the first trial the target object is initially placed directly in front of the robot ($d = 0.0$); in the second trial the object is initially placed 0.5 meters in front of the robot ($d = 0.5$), and so on in increments of 0.5 meters until in the seventh trial the object is initially placed 3.0 meters in front of the robot ($d = 3.0$). For each regime and each trial, 100 independent runs were performed, giving a total of $2*7*100 = 1400$ conducted runs.

Sample run Figure 1 illustrates two compound behaviors that evolved during one of the runs from regime I and trial 1 ($d = 0.0$). Fig. 1a-g illustrates the successful locomotion toward and lifting of the target object after it has been moved 2.8 meters from the robot. Fig. 1h-n illustrates the successful behavior from later in the run when the target object has been moved 3.0 meters from the robot. Both of these controllers are bistable in the sense that when the robot is far from the target object the distance sensors output low values, which push the controllers into a periodic attractor. This periodic attractor moves the robot's limbs in a cyclic pattern leading to locomotion toward the object (i.e. taxis behavior). When the robot nears the object either the high values of the distance sensors, the sudden firing of the touch sensors when the claw tips come in contact with the object, or a combination of both push the controller out of the cyclic attractor and into a point attractor in which the robot's claw lifts the target object onto its back and then the robot stops moving.

Fig. 3 reports the fitness progression of this particular run in more detail. As can be seen in Fig. 3a, a succession of successful CTRNNs allow the target ob-

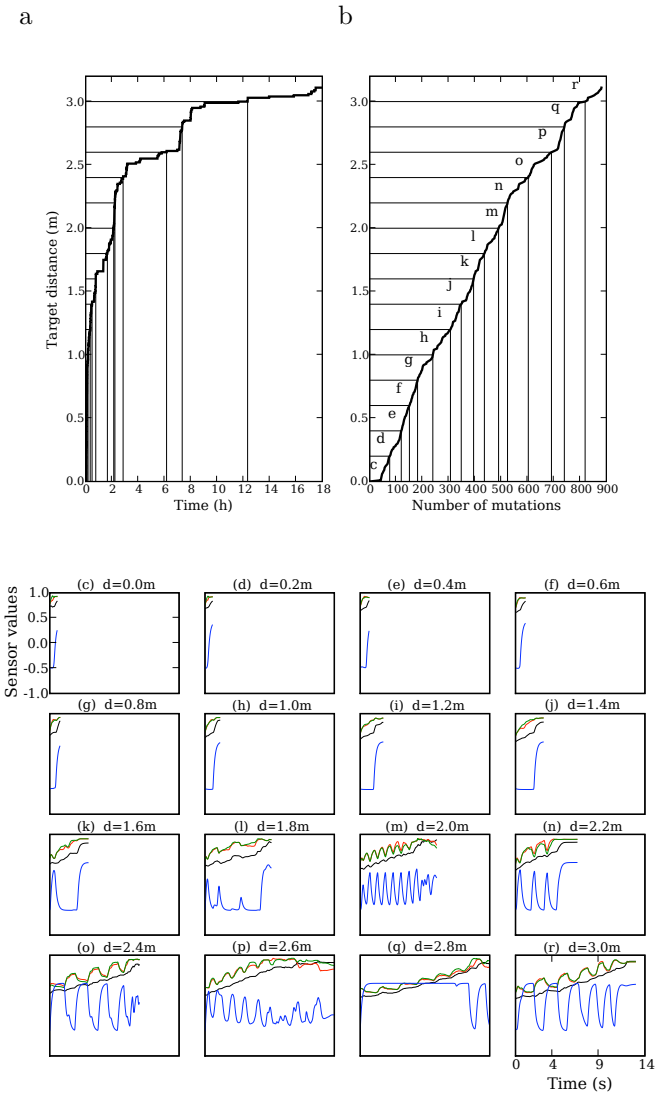


Figure 3: The fitness progression in a sample run. **a**: The fitnesses of the best controllers over the 18 CPU hours of the run (thick line). Thin lines indicate 16 successful controllers discovered when the target object was placed 0.0, 0.2, ... 3.0 meters from the robot. **b**: The same fitness progression is plotted as a function of the number of mutations that separate one successful controller from the next one. The same 16 successful controllers are indicated by the thin lines. **c-r**: Time series produced by the sensors when executing the 16 successful controllers (blue line=claw base rotation sensor; black line=distance sensor on the robot's back; red line=distance sensor on the left claw tip; green line=distance sensor on the right claw tip; the touch sensors are not shown).

ject to be moved just beyond 3.0 meters, at which time the 18 hours elapse. When the target object is much closer (in the first two hours of the run) there is a more rapid succession of successful controllers than when the target object is further out, which is to be expected for three reasons: (1) there are more behaviors that allow for successful lifting when the target object is close at hand than when it is further away; (2) bistability is not required when the target object is close (i.e. a ballistic behavior that blindly lifts the target object may succeed without requiring cyclic behavior beforehand to reach the target object); and (3) a number of failure conditions that occurred between discovery of successful controllers have extended the time period for an evaluation far beyond the initial 100 time steps.

This last factor is removed from consideration in Fig. 3b, in which the same fitness progression is plotted, but each improvement is measured as a function of the number of mutations that occur between the appearance of a fitter controller and its replacement in turn by a superior controller. The thin lines in Figs. 3a,b denote a selection of 16 successful controllers chosen from various stages in the run: when the target object is placed $[0.0, 0.2, \ldots, 3.0]$ meters from the robot. The behaviors in Fig. 1 correspond to the last two such controllers ($d = 2.8$m and $d = 3.0$m; Fig. 3q,r). Strikingly, a relatively constant number of mutations separate one successful controller from another: Fig. 3b indicates that for this particular run, an average of between 50 and 100 mutations separate the discovery of a successful controller when the target object is placed i meters away and the discovery of a successful controller when the target object is placed $i + 0.2$ meters away.

Figs. 3c-r report the time series sensor data for these 16 successful controllers. The blue line corresponds to the angle sensor in the claw base motor, and the other lines correspond to the three distance sensors. While the target object is still within reaching distance of the robot there is no evidence of cyclic activity in the controller (Figs. 3c-j), indicating that the dynamics of the controller are driven toward a point attractor that results in the target object being lifted onto the robot's back: the claw is held by the controller in a horizontal position for a short period (indicated by the low horizontal blue lines in Figs. 3e-j) before being rapidly rotated upward (the upward blue curves in Figs. 3c-j). After this point, when the target object is beyond 1.4 meters (Figs. 3k-r), the time series data from the sensors indicates cyclic activity within the controller. This indicates the discovery of controllers that are pushed into cyclic attractors which lead to rhythmic gaits that bring the robot to within reaching distance of the target object. It can be seen that among the bistable controllers, the cyclic attractors exhibit very different patterns: in Fig. 3l the attractor is hardly periodic; in Fig. 3m the frequency of oscillation is much higher than in the other controllers; and in Fig. 3q one part of the controller is saturated (the blue line maintains a constant, maximum value for most of the evaluation) while the other part is periodic.

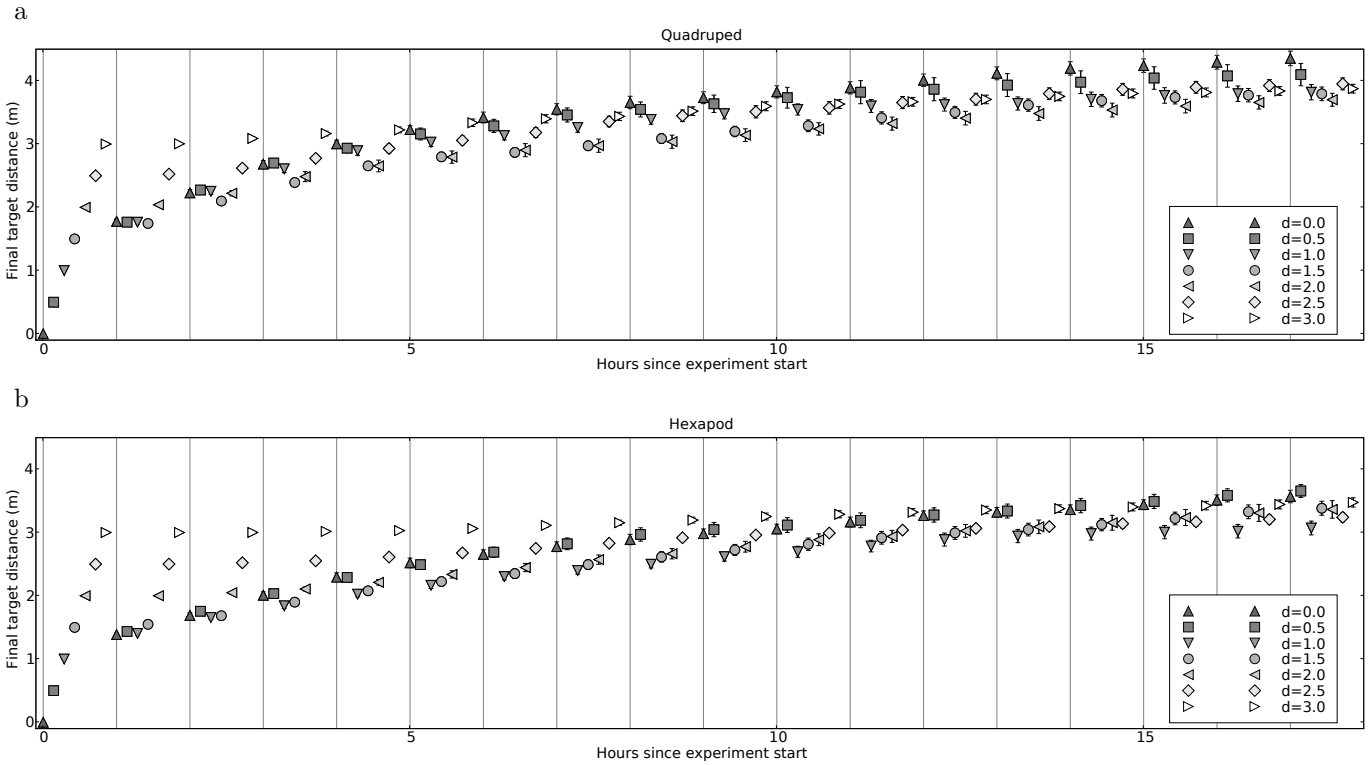


Figure 4: Evidence for behavior trajectory unidirectionality. **a:** The mean performance of the seven trials conducted using the quadruped robot (regime I; trial 1 = upward pointing triangle = initial target object distance is 0.0m from the robot [$d = 0.0$m]; trial 2 = square [$d = 0.5$m]; trial 3 = downward pointing triangle [$d = 1.0$m]; trial 4 = circle [$d = 1.5$m]; trial 5 = rightward pointing triangle [$d = 2.0$m]; trial 6 = diamond [$d = 2.5$m]; trial 7 = rightward pointing triangle [$d = 3.0$m]). **b:** Mean performance of the seven trials using the hexapod robot (regime II). Error bars indicate standard errors of the means ($n = 30$).

This last example behavior allows the robot to keep the claw in a vertical position while walking toward the target object (Figs. 1b,c), but when it nears the object the claw is rotated downward for grasping (Fig. 1d), upward for lifting (Figs. 1e,f) and finally downward to leave the object on its back (Fig. 1g). This example illustrates that under the right conditions the same monolithic controller can be partitioned into different components in which some parts together exhibit similar dynamics (in this case the oscillatory motor neurons involved in locomotion) while other components exhibit different dynamics (in this case the saturated values on the claw base motor keep it raised until it is needed for lifting). This partitioning, however, is evolved and may change over the course of an evolutionary run: when the target object is moved further out to 3.0 meters, behavior shifts such that the claw rotates upward and down in synchrony with the oscillations of the leg motor neurons (Fig. 3r).

Unidirectionality of behavior trajectories Fig. 4 reports the mean performances of the runs when using the quadruped and hexapod. The performance of an individual run is determined as the distance to which the target object was moved beyond the robot at the time of the run's termination: in other words the more successful controllers produced by a run, the further out the target object is moved.

Within both regimes, and within each trial, the 30 runs out of the 100 with the best performances at termination were extracted and the mean performance within that group was calculated at the beginning of the run (the leftmost groupings in Fig. 4), after the first hour (the second leftmost grouping in Fig. 4), and so on up to the mean performance achieved by the group after the 17th hour (rightmost grouping in Fig. 4). As can be seen, for the case of the quadruped the mean performance of the best runs of trial 1 are statistically significantly higher than the same set of runs extracted from trials 3 through 7 (the upward pointing triangle is significantly higher than the third through seventh markers in the rightmost grouping in Fig. 4a).

Also, although not quite significant, the mean performances for the hexapod are higher in trials 1 and 2 after hour 17 compared to the other trials (the heights of the first and second markers are higher than the third

through seventh markers in the rightmost grouping of Fig. 4b).

Discussion

Figs. 1 and 3 demonstrate that by using the method introduced here it is possible to teach a robot to learn one dynamic behavior and gradually incorporate a second behavior into the same controller. The optimization process first discovers a controller which settles into a point attractor corresponding to grasping and lifting. This unistable controller then gradually evolves into a bistable controller which can also settle into a periodic attractor that corresponds to locomotion toward the object.

Fig. 3 indicates that during this process a succession of controllers are discovered with marked differences: the shapes and frequencies of the oscillations are quite different. In addition, there are controllers for which only part of the network displays oscillatory behavior, while the other is held at a saturation point until the robot nears the target object. This approach is attractive in that it may be more scalable than approaches that add new controller components for each new behavior (Brooks (1986), Calabretta et al. (2000), and Reil and Husbands (2002)). In these latter approaches the controller size grows linearly with the number of behaviors. In the proposed approach new controller structure may grow sub-linearly with the number of behaviors: new neurons and connections only need be added when the current monolithic controller can no longer incorporate an additional attractor. However, a more rigorous comparison between these approaches is warranted.

Fig. 4 justifies that scaffolding is necessary to achieve successful multistable controllers. In trials 1 and 2, the target object is initially placed close to the robot, forcing it to evolve a controller capable of grasping and lifting first; as the object is moved further out it incorporates locomotion. In trials 3 onward, the target object is initially placed further out, forcing the robot to learn locomotion first, followed by grasping and lifting. In general, among the best 30 runs these latter trials are less successful after 18 hours than the best 30 runs of trial 1: this difference is statistically significant for the quadruped, and marked yet not significant for the hexapod. This shows that there is an inherent unidirectionality in at least some behavioral trajectories: for a given set of behaviors it is easier to learn task i and then task j, compared to learning task j and then task i. Only the best 30 of the 100 runs were compared here, as some runs within all trials and all regimes failed to achieve controllers capable of both behaviors: future work is planned to increase the consistency of this approach.

Behavior chaining is a kind of robot shaping technique (Singh (1992), Dorigo and Colombetti (1994) and Saksida et al. (1997)), but in behavior chaining it is assumed that there is an *a priori* optimal ordering by which behaviors should be incorporated into the controller. Further, it assumes that this order is dictated by the agent, its task environment and the optimization process, and less by the agent's current behavioral competency.

For instance in (Goldenberg et al. (2004)) an agent is initially trained against a subset of environments, after which it is tested in unseen environments: the unseen environment in which the agent performs worst is then incorporated into the training set. It was shown that this can, in some cases, increase an agent's behavioral flexibility. However, the approach introduced here indicates that the order in which the agent is presented with environments affects the probability that an agent will be able to increase its behavioral flexibility. Consider an example: an agent undergoing shaping may perform very poorly in unseen environment i and less poorly on unseen environment j. The shaping schedule as described in Goldenberg et al. (2004) will incorporate environment i into the training set first. However, it may be that the agent should learn to behave successfully in environment j first, and will only then be able to behave successfully in environment i. The hypothetical shaping schedule described above may therefore fail to yield a behaviorally flexible agent. Future investigation will determine whether the optimal sequence in which behaviors should be learned can be predicted before learning begins, or whether it can be determined by the agent's current behavioral competency.

Conclusions

This paper has introduced a method that automatically trains a robot to exhibit a sequence of dynamic behaviors by drawing on evolutionary robotics, developmental psychology, and in particular on advances in embodied artificial intelligence (Pfeifer and Bongard (2006)) that equate specific behaviors with attractor states arising from the interaction of a robot's brain, body and environment, rather than the more subjective labeling of behaviors by an external observer. This method enabled a simulated robot to exhibit a compound behavior not yet reported in the literature: dynamic legged locomotion toward an object followed by grasping, lifting and holding of that object in a physically-realistic three-dimensional environment.

Automated methods such as evolutionary robotics are particularly well suited for domains where it is difficult for a human operator to translate a desired high-level behavior into a detailed sequence of motor commands. This is particularly true when the robot is capable of nonlinear behavior such as dynamic locomotion. However, this advantage has to date seemingly been

counterbalanced by a corresponding drawback: scalability. That is, there is no known scalable method for gradually expanding the behavioral repertoire of an autonomous robot. This work suggests scalable behavior generation is possible if both the fitness function and a dynamic scaffolding schedule are carefully chosen. Rather than attempting to create a purely automatic method, this approach takes advantage of the natural ability of a human operator to break down a compound behavior (such as locomotion toward and then manipulation of a distal object) into separate behaviors (such as minimizing the distance to the object, grasping, and then lifting) each of which can then be sequentially mastered using automated optimization methods.

The operator's intuition is formalized by requiring them to determine what constitutes failure or success, and what modifications to the task environment should be made in either case. Future work is planned to determine just what failure and success definitions, and their associated scaffolds, are appropriate to realize robots capable of an increasing number of behaviors such as locomotion, object manipulation, object transport, locomotion over uneven terrain, and teamwork.

Source code The source code, data files and Python scripts for visualizing results are available at `www.cs.uvm.edu/~jbongard`.

References

Beer, R. (2006). Parameter space structure of continuous-time recurrent neural networks. *Neural Computation*, 18:3009–3051.

Beer, R. D. (1996). Toward the evolution of dynamical neural networks for minimally cognitive behavior. *Proceedings of the Fourth International Conference on Simulation of Adaptive Behavior*, pages 421–429.

Bongard, J. and Lipson, H. (2007). Automated reverse engineering of nonlinear dynamical systems. *Proceedings of the National Academy of Science*, 104(24):9943–9948.

Bongard, J., Zykov, V., and Lipson, H. (2006). Resilient machines through continuous self-modeling. *Science*, 314:1118–1121.

Brooks, R. (1986). A robust layered control system for a mobile robot. *IEEE J of Robotics and Automation*, 2(1):14–23.

Calabretta, R., Nolfi, S., Parisi, D., and Wagner, G. P. (2000). Duplication of modules facilitates the evolution of functional specialization. *Artificial Life*, 6(1):69–84.

Chella, A., Dindo, H., Matraxia, F., and Pirrone, R. (2007). Real-Time Visual Grasp Synthesis Using Genetic Algorithms and Neural Networks. *Lecture Notes in Computer Science*, 4733:567–578.

Dorigo, M. and Colombetti, M. (1994). Robot shaping: Developing situated agents through learning. *Artificial Intelligence*, 70(2):321–370.

Fernandez, J. and Walker, I. D. (1999). A Biologically Inspired Fitness Function for Robotic Grasping. *Genetic and Evolutionary Computation Conference*, pages 1517–1522.

Foss, J., Moss, F., and Milton, J. (1997). Noise, multistability, and delayed recurrent loops. *Physical Review E*, 55(4):4536–4543.

Gallagher, J. C. and Beer, R. D. (1999). Evolution and analysis of dynamical neural networks for agents integrating vision, locomotion and short-term memory. *Proceedings of The Genetic and Evolutionary Computation Conference (GECCO 1999)*, pages 1273–1280.

Goldenberg, E., Garcowski, J., and Beer, R. D. (2004). May we have your attention: Analysis of a selective attention task. *Proceedings of the Eighth International Conference on Simulation of Adaptive Behavior*, pages 49–56.

Harvey, I., Husbands, P., Cliff, D., Thompson, A., and Jakobi, N. (1997). Evolutionary robotics: the Sussex approach. *Robotics and Autonomous Systems*, 20(2-4):205–224.

Hornby, G., Takamura, S., Yamamoto, T., and Fujita, M. (2005). Autonomous evolution of dynamic gaits with two quadruped robots. *IEEE Transactions on Robotics*, 21(3):402–410.

Inamura, T., Toshima, I., Tanie, H., and Nakamura, Y. (2004). Embodied symbol emergence based on mimesis theory. *International Journal of Robotics Research*, 23(4):363–377.

Lee, W., Hallam, J., and Lund, H. (1998). Learning Complex Robot Behaviours by Evolutionary Computing with Task Decomposition. *Learning Robots: 6th European Workshop*, pages 155–172.

Lungarella, M., Metta, G., Pfeifer, R., and Sandini, G. (2003). Developmental robotics: a survey. *Connection Science*, 15(4):151–190.

Nolfi, S. (1997). Evolving non-trivial behaviors on real robots: A garbage collecting robot. *Robotics and Autonomous Systems*, 22(3):187–198.

Nolfi, S. and Floreano, D. (2000). *Evolutionary Robotics*. MIT Press, Boston, MA.

Okada, M. and Nakamura, Y. (2004). Design of the continuous symbol space for the intelligent robots using the dynamics-based information processing. *Proceedings of the IEEE Intl. Conf. on Robotics and Automation*, pages 3201–3206.

Pfeifer, R. and Bongard, J. (2006). *How the Body Shapes the Way We Think: A New View of Intelligence*. MIT Press.

Pratt, J., Chew, C., Torres, A., Dilworth, P., and Pratt, G. (2001). Virtual Model Control: An Intuitive Approach for Bipedal Locomotion. *The International Journal of Robotics Research*, 20(2):129–143.

Reil, T. and Husbands, P. (2002). Evolution of central pattern generators for bipedal walking in a real-time physics environment. *IEEE Transactions on Evolutionary Computation*, 6(2):159–168.

Russell, S. J. and Norvig, P. (1995). *Artificial Intelligence: A Modern Approach*. Prentice-Hall, Upper Saddle River, NJ.

Saksida, L. M., Raymond, S. M., and Touretzky, D. S. (1997). Shaping robot behavior using principles from instrumental conditioning. *Robotics and Autonomous Systems*, 22(3-4):231–249.

Saunders, J., Nehaniv, C. L., Dautenhahn, K., and Alissandrakis, A. (2007). Self-imitation and environmental scaffolding for robot teaching. *International Journal of Advanced Robotic Systems*, 4(1):109–124.

Singh, S. P. (1992). Transfer of learning across sequential tasks. *Machine Learning*, 8:323–339.

Wood, D., Bruner, J., and Ross, G. (1976). The role of tutoring in problem solving. *J Child Psychol Psychiatry*, 17(2):89–100.

Ziemke, T., Bergfeldt, N., Buason, G., Susi, T., and Svensson, H. (2004). Evolving cognitive scaffolding and environment adaptation: a new research direction for evolutionary robotics. *Connection Science*, 16(4):339–350.

Evolving CSR Strategies in Virtual Plant Communities

Stefan Bornhofen, Claude Lattaud

Laboratoire d'Intelligence Artificielle de Paris 5
Université Paris Descartes
45, rue des Saints-Pères, 75006 Paris, France
{stefan.bornhofen,claude.lattaud}@math-info.univ-paris5.fr

Abstract

This paper introduces a functional-structural plant model based on Artificial Life concepts and reports studies on evolutionary dynamics in virtual plant communities. The characteristic of the present approach lies in plant evolution at both functional and structural levels. The conducted experiments focus on the emergence of different life history strategies in an environment with heterogeneous resource availability and disturbance frequency. It is found that, depending on the encountered conditions, the plants develop three major strategies classified as competitors, stress-tolerators and ruderals according to Grime's CSR theory. Most of the evolved characteristics comply with theoretical biology or field observations on natural plants.

Introduction

Life history theory seeks to understand the variation in traits such as growth rate, number and size of offsprings and life span observed in nature, and to explain them as evolutionary adaptations to environmental conditions (Stearns, 1992). In the realm of plant life, Grime (1977) identified two major environmental factors limiting growth. Stress is defined as "conditions that restrict production", e.g. shortages of resources or suboptimal temperatures. Disturbance is "the partial or total destruction of the plant biomass" and arises from the activities of herbivores or from abiotic phenomena such as wind damage or fire. Grime suggested the existence of three primary strategies, i.e. sets of life history traits, which prevail in the environment depending on the encountered levels of stress and disturbance:

- Competitors (C) live in fertile undisturbed habitats and are adapted for long-term occupation.
- Stress-tolerators (S) persist in low resource environments, or where survival depends on the allocation of resources to maintenance and defense.
- Ruderals (R) are found in frequently disturbed habitats and exhibit rapid development and reproduction.

These types are extreme variants of the whole spectrum of plant life history strategies. The disturbance axis recalls the concept of the r-K selection continuum that depends on the predictability of the environment (MacArthur and Wilson, 1967; Pianka, 1970). Grime additionally assumed that plants cannot grow where disturbance and stress are both high.

Although Grime's classification is central in plant life history theory, only few studies using computer simulation have been published on the subject. Mustard et al. (2003) addressed the evolution of CSR strategies in a virtual environment by means of a mutable model of single plant growth based on a number of life history traits. They observed the emergence of a variety of physiological adaptations consistent with field and theoretical evidence. However, the model was restricted to a highly simplified morphology which could not evolve.

In the area of plant modeling, there exists a variety of functional-structural plant models (FSPM) combining a 3D representation of the plant with the simulation of a number of physiological processes (Allen et al., 2005; Perttunen et al., 1998), but they are typically not designed for experiments at evolutionary scale. The present paper intends to study CSR strategies through experiments with an evolutionary FSPM and addresses the question of if and to what extent recognizable growth patterns evolve, and which morphological characteristics emerge in addition to the physiological ones. Pertinent results would constitute a success in bringing Artificial Life concepts to bear in the science of plant modeling.

The experiments extend the studies on life history evolution described in (Bornhofen and Lattaud, 2006) by applying "implicit" selection in contrast to "explicit" selection. Explicit selection uses iterated generation steps and evaluates the whole population of every generation by an imposed fitness function. Implicit selection is not guided. It corresponds to the struggle for existence observed in natural systems, as originally proclaimed by Darwin (1859), and results in the emergence of characteristics that lead to high survival and reproduction in the encountered environment.

The next section gives an overview of the state of the art in evolutionary plant modeling. In Section 3 the used plant model is briefly presented. The conducted experiment is de-

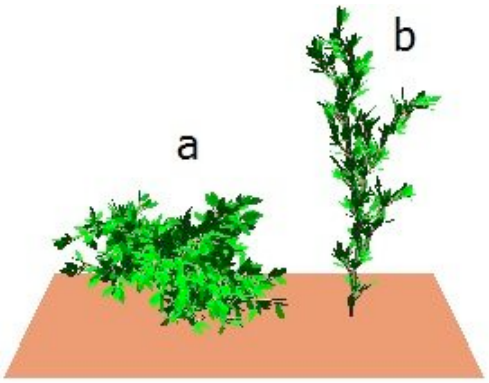


Figure 1: Evolved plants: isolation (a) and competition (b)

scribed and analyzed in Section 4. Section 5 concludes the paper and discusses the perspectives of the approach.

Plant modeling

FSPM are designed for the study of growth dynamics and the impact of environmental factors on plant form development (Sievanen et al., 2000). Their detailed calculations of spatial architecture and resource flow draw a faithful picture of real plants in a virtual environment, giving rise to the notion of "virtual plants" (Room et al., 1996). In order to accurately represent real plants, the model complexity most often involves a computational cost per individual which renders simulations of large communities difficult to realize for simple reasons of memory and time. Moreover, FSPM are typically customized by botanical data for individual or population level scenarios of specific natural species.

Aside from FSPM conceived within the scientific community of biologists, an amount of studies on plants have been carried out in the research field of Artificial Life. Their primary objective is the application and adaptation of ALife concepts and notably evolutionary algorithms (Holland, 1975) in the context of plant development. As the purpose of the conducted studies is different, priority is given to simplification. Plants are represented as structures based on a set of morphological growth rules, most often expressed by variants of the L-system formalism (Prusinkiewicz and Lindenmayer, 1990), with no or only minimal physiology and interactions with their environment.

Jacob (1994) published works concerning the evolution of context-free and context-sensitive L-systems representing simple artificial plants. He developed the "Genetic L-systems Programming" paradigm, a general framework for evolutionary creation of parallel rewriting systems. His approach was extended by Ochoa (1998) who evolved 2D plant structures and concluded that L-systems are an adequate genetic representation for studies which simulate natural morphological evolution.

With regard to more user interactivity, Mock (1998) modeled artificial plants for a virtual world where the human observer chooses the most interesting-looking individuals for further reproduction and evolution. Likewise, some applications such as the Second Garden (Steinberg et al., 1999) or the Nerve Garden (Damer et al., 1998) appeared in the past years on the Internet, allowing users to grow and interact with artificial plant communities in online worlds.

The above cited models focus on the morphological aspect of a plant and hold no or only minimal physiological and environmental dynamics, so that experimental results possess a limited significance with respect to natural plants. Recently, ALife plant models featuring more biological considerations have appeared. Most notably, Ebner et al. (2002) incorporated interactions between plant and environment by evaluating the individuals for their amount of captured virtual sunlight. As a major result, it was shown that under competition plants grow high whereas they grow small and bushy when developing independently (Figure 1).

Model Description

To take a further step on the path of evolutionary plant models, the following section introduces virtual plants that not only interact with the environment, but also combine morphology with physiological processes. The plants are based on ALife concepts, as they are emergent and adaptive structures with simple underlying rules, but at the same time they contain all the major elements of an FSPM, that is a 3D architecture combined with a framework of resource assimilation, flow and allocation. An artificial genome contains mutable information which describes numerous characteristics concerning morphological as well as physiological growth processes, and evolutionary forces can act on these traits by favoring reproduction of those individuals which turn out to be adapted to a given selection process. Previous papers (Bornhofen and Lattaud, 2006, 2007) already introduced the model and suggested its utility for studies on adaptations of morphology and life history parameters in comparison with natural plants. A detailed mathematical description of the model is given in (Bornhofen and Lattaud, 2008).

Table 1: L-system alphabet of the used plant model

Character	**Function**
l	leaf, captures virtual light
f	flower, represents a reproductive module
b	branch, creates a supporting structure
r	fine root, assimilates nutrients in the soil
c	coarse root, creates a supporting structure
A...Z	apex, predecessor of a production rule
[ ]	indicates a ramification
+−<>$ &	represents a 3D rotation

Environment

The plants grow in a continuous 3D virtual environment which is composed of two components, the soil and the sky, providing light and minerals respectively. These two resources are of prime importance for the growth of natural plants (Westoby et al., 2002). Other significant resources such as water and CO_2 are currently not modeled, which corresponds to the assumption that their supply is constant and sufficient. Environmental heterogeneity is achieved by subdividing the soil and the sky into voxels that contain locally available resources.

The sky holds a vertical light source parameterized by its initial irradiance. If an object is situated in a sky voxel, it casts shadows such that the luminosity in all subjacent voxels is decreased. In order to avoid time-consuming computation such as geometrical calculations or the use of computer graphics, the shading factor does not depend on the exposed surface of the object but on its volume. Just as sky voxels contain a local light intensity, soil voxels contain minerals. A resource flow from regions of high concentration to regions of low concentration is modeled by Fick's first law of diffusion (Fick, 1855). All the assimilated nutrients of a virtual plant are eventually redeposited in the soil so that their total amount in the environment is constant within a simplified mineral cycle. The nutrients of dead roots are put in the corresponding voxels and those of the aerial compartment in a mold layer which gradually penetrates the upmost soil layer.

Plant phenotype

A virtual plant is divided into a shoot and a root component. The morphologies are expressed by two independent L-systems (Prusinkiewicz and Lindenmayer, 1990), whose alphabet is detailed in Table 1. The model allows for stochastic L-systems, but in the scope of this paper only deterministic context free L-systems are applied. This choice was made to disengage the evolutionary dynamics from contingencies at individual level.

The physiological processes of the plants are based on a two-substrate version of the transport-resistance model

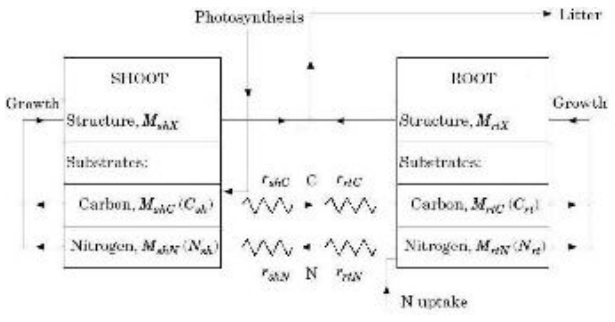


Figure 2: The transport-resistance model

(Thornley, 1998), where an aboveground and a belowground compartment assimilate, exchange and allocate the two resources carbon and minerals (Figure 2). However in the presented plant model, new biomass is not stored in a real-valued aggregate variable, but distributed to the apices of the current plant morphology. An L-system rule is applied once the biomass of an apex reaches the required cost for the production of the corresponding successor string. This value is calculated from the genetically defined costs of all plant modules that will be produced. Growing apices also have to pay for the thickening of the supporting modules below them. This stipulation guarantees that the growth cost increases with the distance from the ground and refers to the pipe model theory (Shinozaki et al., 1964) which states that any cross sectional area in a branching system, whether shoot or root, is proportional to the biomass of the captors, leaves or fine roots, that it serves.

Plant genotype

The development of the virtual plants is ruled by a set of "genetic information" recorded in a genotype. It contains the variables of the transport-resistance model such as growth and litter rates or resource assimilation and inhibition, as well as twelve additional real-valued physiological parameters like longevity, duration of bloom and seed biomass. Moreover, it specifies the parameters and production rules of the root and shoot L-systems.

Just as in (Mustard et al., 2003), real-valued parameters are mutated by selecting a new random value within a range of twenty percent around the current value. L-system mutations occur via genetic operators each of which is associated with a probability of ten percent. They are chosen such that any set of production rules can be constructed by evolution. The following three operators modify the number of rules:

- DeleteR (a rule of the L-system is deleted)
- InsertR (an empty rule is appended)
- DuplicateR (a rule is duplicated and appended)

Five other operators act on the successor strings. Only minor changes, i.e. character by character, are possible between successive generations. For example, if the production $A \rightarrow blfA$ is selected to be mutated, some of the possible mutations are

- DeleteC (a character is deleted): $A \rightarrow blf$
- InsertC (a character is inserted): $A \rightarrow b\&lfA$
- PermuteC (two characters are swapped): $A \rightarrow bflA$
- DuplicateC (a character is duplicated): $A \rightarrow blffA$
- MutateC (a character is replaced): $A \rightarrow b{+}fA$

In order not to obscure the results by too large a genetic search space, the evolving elements in the genotype have been limited for the purpose of this paper. Apart from the morphological growth rules, i.e. the L-system production rules, only five real-valued physiological parameters, controlling five major life history trade-offs, are allowed to mutate (Table 2). The significance of these parameters in the

Table 2: Genetic parameters and their trade-offs

Parameter	Trade-off
$0 <$*longevity*	Long life - early reproduction
$0 <$*maturity*< 1	Vegetative - reproductive allocation
$0 < k_G$	Rapid growth - resource conservation
$0 <$*seedX*	Seed size - seed number
$0 <$*seedD*	Seed propagation - seed survival

plant model is specified in the following subsection. Note that a number of other life history traits such as plant height or seed number are not encoded in the genotype but are emergent properties of the model.

Life cycle

The shoot and root morphologies of a seedling both start with the single non-terminal character A. A small amount of initial biomass *seedX* allows the young plant to develop its first modules, but subsequently it has to rely on the acquisition of resources and the production of biomass. In this process, the parameter k_G of the transport-resistance model denotes the utilization rate of stored resources (Thornley, 1998). Sexual maturity is determined by *maturity*, a fraction of the overall life span *longevity*. When a plant reaches the age of *maturity*longevity*, the reproductive modules initiate the development of a seed. Reproduction occurs asexually, i.e. seed genotypes are a mutated version of a copy of the mother plant genotype. Mutation is sufficient to explore the entire genotype space, and previous studies using explicit selection (Bornhofen and Lattaud, 2006, 2007) suggest a low efficiency of the applied crossover operators inspired by (Ebner et al., 2002). Therefore, no pollinisation mechanisms have been implemented for implicit selection. During seed production, reproductive modules become a resource sink and compete with the apices for a share of newly produced biomass. When they attain the final seed biomass *seedX*, the seed is considered ripe and dispersed in the neighborhood of the plant at a maximum distance of *seedD*. After a limited span of life *longevity* the plant dies and its resources are restituted to the environment.

Experiments

The presented simulations focus on evolutionary adaptations in an environment with heterogeneous levels of disturbance and mineral stress. If recognizable CSR strategies emerged, the result would not only provide new theoretical support for Grime's theory by simulation in silico but also, more generally, point out how the scope of FSPM can be extended to the study of evolutionary dynamics in plant communities.

Setup

The environment is a bordered square terrain (extent: 40 length units) divided into 5x5 patches called A1 to E5 and

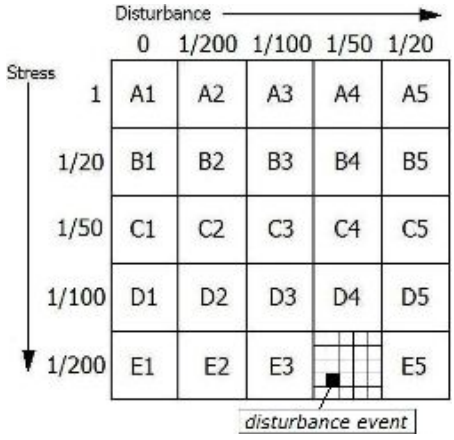


Figure 3: The different patches

featuring unequal levels of disturbance and stress. Along the horizontal dimension, "disturbance events" kill plants with a probability increasing from column 1 to 5. Such events are not applied to an entire patch, but they potentially occur in each cell of a 5x5 subgrid. The subdivision was chosen such that a single disturbance does not erase the whole population of a patch, but provides sufficiently large gaps for the establishment of new plants. Along the vertical stress dimension, an abiotic mineral cycle has been added to the environment. Starting from an initially homogeneous amount of nutrients, the resources of the downmost soil layer of each patch drain into a separate pool which is flushed back to the surface by random events. They correspond to rainfall which fertilizes the soil at irregular intervals, and mineral stress increases from row A to E with decreasing probabilities for these "nutrient flushes". In order to maintain the induced soil heterogeneity during simulation, diffusion only takes place between the voxels of the same patch. Nutrient flow across the overall environment would blur the different levels of stress. Figure 3 schematically plots the environmental setup and indicates the applied probabilities of disturbance events and nutrient flushes per time step. The values along both dimensions are experimentally determined such that they allow the virtual plants to evolve different life history strategies under the extreme conditions of the patches A1, A5 and E1, whereas no population succeeds to settle in patch E5.

At the beginning of the simulation, one thousand seeds are dispersed across the terrain. Their non-mutable genetic parameters are identical and have been adopted from previous simulations on life history evolution (Bornhofen and Lattaud, 2006). However, the L-system derivation depth of the plant morphology has been restricted to five productions. Higher values lead to an exponential increase of simulation complexity, and previous works attest that they do not induce evolutionary tendencies that are fundamentally different from those observed in this paper (Bornhofen and Lattaud, 2006, 2007). The mutable physiological parameters

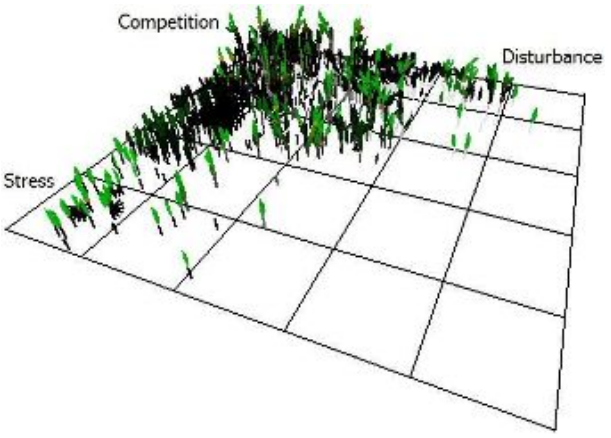


Figure 4: Sample view on the virtual environment

are randomly initialized within suitable limits which have been assessed experimentally by analyzing the outcome of a series of evolutionary test runs in the same environment. To grant the morphological evolution as much freedom as possible, the initial seeds all start with the L-systems of a "minimal" reproducing virtual plant containing the single rule $A \rightarrow r$ in the root compartment and $A \rightarrow lf$ in the shoot compartment. During the simulation, the plants grow, compete and reproduce freely via intrinsic selection, i.e. without imposed fitness criteria. Differences in life history dynamics emerge from mutations in every new seed genotype, and if a strategy turns out to ensure better survival and reproduction, it has a greater chance to increase its abundance in the population.

Twenty replicate runs are performed for a period of 10000 time steps. The size of the terrain and the length of the simulations represent a trade-off between maximizing the number of simulated individuals and harnessing simulation time and allocated computer memory. One run would take about ten hours and nearly use the full memory on a PC - 3GHz, 1Go RAM. Throughout the simulations, the following measures are regularly recorded for each patch:

- the number of plants
- the number of produced seeds
- the total plant biomass
- the averaged five mutable parameters

The results of the next section present mean values over the twenty simulations.

Propagation dynamics

The initial plants, dispersed throughout the entire environment, rapidly perish in most parts of the terrain and only persist in the upper left corner, i.e. the neighborhood of patch A1. All other regions turn out too hostile for random plants. The remaining individuals start to reproduce and spread new seeds. As seed dispersal is not limited by the patch borders, the population steadily invades the terrain along the two dimensions toward the patches A5 and E1. Note that it is the gradual increase in difficulty that allows the plants to discover suitable survival strategies for these extreme environmental conditions. After only a few generations, the formation of the CSR triangle is recognizable. Figure 4 shows a view on the virtual environment during a typical simulation. According to the experimental setup, the plants establishing in patch A1 will be called "competitors", those of patch E1 "stress-tolerators" and those of patch A5 "ruderals".

Figure 5a plots the number of plants that grow in the three key patches throughout the simulations. Starting from the dispersed random seeds, the plants directly increase their population in the competitor's corner A1. Stress-tolerators do not exist yet, and the initial plants of patch E1 disappear without offspring. Around time 1000, the population originating from A1 evolves a strategy to survive in this difficult environment and reinvades the patch. Similarly, the first plants of patch A5 are rapidly wiped out by disturbance before being able to reproduce, and it is not before time 2000 that a small population starts to persist.

After an initial peak, the number of competitors diminishes and nearly comes into balance at the simulation end. Although one might expect evolutionary adaptation to lead to a continuous plant increase per patch, a decrease is observed. This phenomenon is explained by the fact that from the initially defined minimal morphology, featuring one leave and one fine root, the plants evolve toward architectures consuming more resources per individual, which affects the carrying capacity of the patches. It is not the number of plants, but the amount of plant biomass per patch that is maximized by evolution (Figure 5b).

Physiological adaptations

Due to the five mutable real-valued parameters allowing the plants to physiologically adapt to the environmental conditions, each genotype maps to a vector in a five-dimensional space (however a one-to-one mapping is not given because the genotypes also contain the morphological L-system rules). In order to better apprehend the physiological component of the evolving strategies, the vectors of all the plants

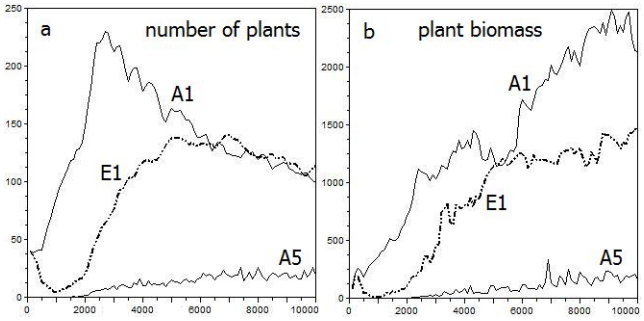


Figure 5: Number of plants and plant biomass per patch

in a patch are averaged. It is important to note that the resulting aggregated data is meaningful because the low numbers of plants per patch, i.e. not more than one hundred individuals occupying the same ecological niche, allow supposing that multiple strategies cannot coexist during one simulation. By evolution, these mean values move within the vector space toward positions which correspond to adapted strategies for a particular patch.

Just as in (Mustard et al., 2003), the resulting strategies at the simulation ends are analyzed using principal component analysis (PCA) (Jolliffe, 1986). The algorithm transforms a multi-dimensional data set to a new coordinate system such that maximum variability is visible. By considering lower-order principal components and ignoring higher-order ones, potential clusters in the cloud of data points may become recognizable. Figure 6 plots the first two components of the PCA applied to the set of evolved strategies in the key patches A1, E1, A5 during all replicate simulations. It can be observed that the results associated to each patch tend to cluster. The pattern attests that the environmental factors disturbance and stress lead to the emergence of contrasting strategies in the virtual plant model. As a next step, it is studied if these physiological adaptations match the predictions of Grime's CSR theory or show other similarities to natural plants found in analogous environments. The evolved mean values of the mutable parameters are summarized in Table 3.

Ruderals possess a low *maturity*, i.e. only a minimum share of lifetime is devoted to individual growth before investing biomass into seeds. Frequent catastrophes force them to spawn as early as possible, so that there is selection pressure toward small values. For the same reason, selective forces lead to the evolution of low *longevity*, as the threshold of sexual maturity scales linearly with life span in the model (see Section 3). A low seed biomass *seedX* allows the production of many seeds in a short time. Ruderals also evolve a high growth rate k_G since this parameter is responsible for the amount of resources consumed per time step, and selection turns out to favor high resource utilization in order to accelerate the life cycle. This suite of traits matches the life history strategy of r-selected plants in unpredictable environments (Pianka, 1970).

Competitors feature a significantly higher *maturity* than ruderals. They need a distinctive period of vegetative growth in order to gain height and get access to light. Moreover, as no disturbance events occur in their patch, *longevity* tends to evolve high values in order to obtain more time for reproduction. Due to strong competition in the patch, these plants develop a high seed biomass *seedX* in order to increase seed survival. Again, the observed values comply with the theory of K-selected plants in constant environments (Pianka, 1970).

Stress-tolerators evolve the longest life span. Due to few soil resources, growth and reproduction are slow. Therefore,

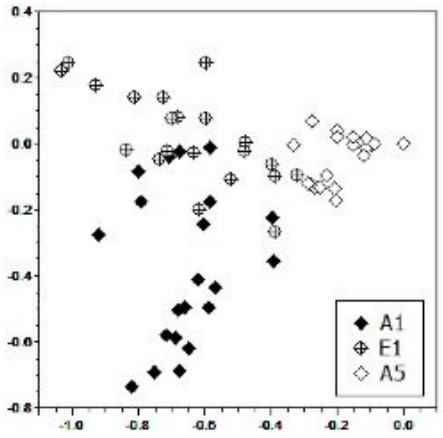


Figure 6: PCA of the final plant strategies

only high values of *longevity* may grant enough time to run through a complete life cycle. The delayed *maturity* suggests that there exists significant competition between the individuals so that they have to ensure survival before producing offspring. Natural stress-tolerators typically feature an inherently slow biomass production in order not to overconsume the available resources (Chapin et al., 1993). In the simulations, their virtual counterparts likewise develop low k_G, but the difference to competitors is not significant. The environmental nutrient flushes in patch E1 might not be rare enough to induce a more distinct result.

Interestingly, in contrast to the other physiological values, the evolution of *seedX* does not exhibit a monotonically increasing or decreasing curve. Figure 7a indicates that, starting from the initial random values, *seedX* first rapidly drops in all patches before it starts to rise again around time 2500. This phenomenon is caused by the fact that the pioneering plants do not encounter severe competition so that, in the short term, there is selection for small and frequently produced seeds. However, when the plant population densifies and morphological evolution decreases the carrying capacity of the patches, seedlings require more biomass to survive and grow toward resources. The simulations attest that this constraint is particularly crucial for competitors. Just as in nature, there is a relationship between large seed size and establishment in shady stable plant associations (Foster and Janson, 1985). Figure 7b shows that the number of produced seeds is opposite to seed biomass. In particular, ruderals are selected for a high number of offsprings.

The evolution of *seedD* involves a trade-off between propagation speed and individual survival. Too small values impair the spread of genetic information, and moreover seedlings may suffer resource deficiency from the proximity to each other and their mother plant. With high *seedD*, on the other hand, offspring potentially ends up in regions they are not adapted to, or even outside the virtual environment. The simulations yield no significantly contrasting

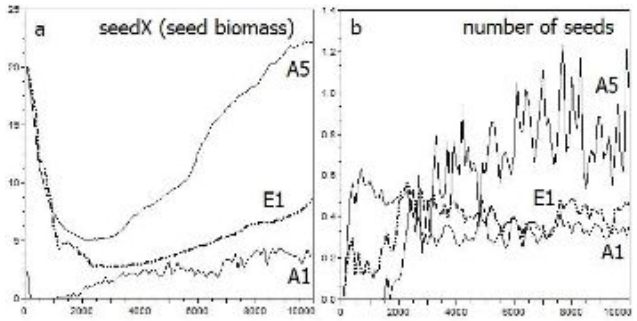


Figure 7: Mean seed biomass and number per patch

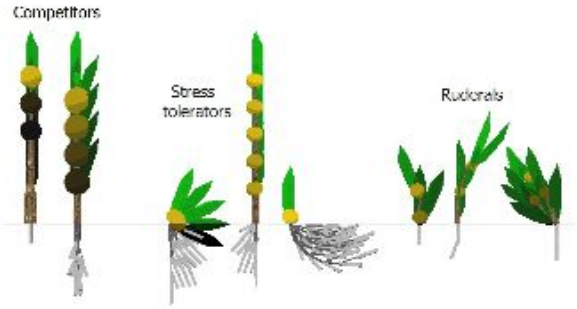


Figure 8: Evolved morphologies

results for this parameter. The evolved values in all three key patches correspond to slightly less than their extent (8 length units). An explanation can be found in the experimental setup. In the corners of the virtual terrain most of the adjacent areas are lethal, so that strong selection pressure exists toward spawning offspring inside the same patch, and no further differences depending on disturbance and stress can be observed. Although *seedD* does not yield differentiated results as regards the CSR strategies, the values demonstrate an evolutionary adaptation to the risks of long-distance seed dispersal. As an example in nature, it has been observed that plants which colonized islands started to evolve reduced dispersal distances presumably because selection favored individuals whose seeds do not get lost in the surrounding ocean waters (Cody and Overton, 1996).

Morphological adaptations

The virtual plants evolve in their environment not only by changes in physiology. The mutating shoot and root L-systems additionally lead to the emergence of distinct adapted above- and belowground architectures. A look at the plant forms growing in the key patches at the end of the runs reveals that the three life history strategies are associated with recognizable morphological characteristics. Figure 8 illustrates some typical plant architectures which evolved during the simulations. In all the runs, competitors develop a high stem without branches in order to rapidly reach the light in their crowded environment. Small plants are penalized as they do not photosynthesize enough carbon for reproduction. As mineral nutrients are abundant, competitors do not invest much biomass into roots. Note that, since no mechanical constraints such as gravity or wind are modeled, high and slim shoot structures do not require deep roots to provide physical support.

Ruderals have the most simple, condensed morphologies. They do not struggle for minerals, and biomass needs to be invested into the rapid production of seeds, so that the root structure remains elementary. Moreover, catastrophes constantly remove plants and create clear gaps in the patches. Enough light attains the surface and it is sufficient for photosynthesis to deploy a small number of leaves near the ground.

Stress-tolerators feature the greatest variety of shoot morphologies without distinct evolutionary tendencies. Some runs lead to competitor-like stems, others to only a tuft of low growing leaves. However, due to the phenomenon of "functional balance", plants in low resource patches typically possess a decreased shoot-to-root ratio. This principle states that the resource assimilation of shoot and root tend to an equilibrium with respect to their relative utilization. Lower light provokes a stronger growth of leaves, and few soil nutrients lead to a boosted root growth (Davidson, 1969). Thus, the stress-tolerators tend to invest an important share of their biomass into root structure which results in the evolution of differentiated belowgound architectures.

Table 3: The resulting averaged mutable parameters

	A1(comp.)	E1(stress)	A5(rud.)
longevity	627.58	801.47	196.33
maturity	0.09	0.12	0.03
k_G	0.95	1.09	3.62
seedX	22.16	8.65	3.85
seedD	6.25	7.40	6.25

Conclusion

An experiment on the emergence of life history strategies has been conducted with a simulation platform of virtual plants. The plants, growing in a 3D environment, are based upon the fusion between a two-substrate transport-resistance model as functional component, and an L-system formalism as structural component. Evolution occurs at both functional and structural levels. It was observed that, depending on the degree of encountered disturbance and stress, the plants develop three major strategies which can be termed competitors, stress-tolerators and ruderals according to Grime's CSR theory. Most of the evolved characteristics correspond to hypotheses in life history theory or field observations on natural plants. The emergence of the CSR triangle corroborates the conjectured impact of disturbance and stress on plant evolution and illustrates that plant strategies depend

on the intensity of both environmental factors.

Extending the current simulations, the impact of crucial parameters in the experimental setup such as patch size and disposition needs to be studied more closely. In particular, a toroidal environment can be used to avoid edge effects. The virtual environment could also feature low light as a second kind of stress, which might lead to other morphological and physiological adaptations of the stress-tolerating plants.

The presented results do not only support plant strategy theory by simulations in silico. More generally, they suggest that the scope of FSPM is not restricted to population level experiments, but they also allow for studies on plants at evolutionary scale by integrating adaptive algorithms based on Artificial Life concepts. Due to their inherent contingencies and the qualitative character of emergent phenomena, such models might offer reduced accuracy from a strict biological point of view, but in return they yield insight into the selective forces and constraints which rule adaptation in natural plant life.

References

Allen, M., Prusinkiewicz, P., and DeJong, T. (2005). Using l-systems for modeling source-sink interactions, architecture and physiology of growing trees: the l-peach model. *New Phytologist*, 166:869–880.

Bornhofen, S. and Lattaud, C. (2006). Life history evolution of virtual plants : Trading off between growth and reproduction. *Lecture Notes in Computer Science*, 4193:808–817.

Bornhofen, S. and Lattaud, C. (2007). Evolution of virtual plants interacting with their environment. In *Proceedings of the 9th International Conference on Virtual Reality (VRIC'07), Laval, France*, pages 172–176.

Bornhofen, S. and Lattaud, C. (2008). Competition and evolution in virtual plant communities: A new modeling approach. *Natural Computing*, accepted for publication.

Chapin, F., Autumn, K., and Pugnaire, F. (1993). Evolution of suites of traits in response to environmental stress. *American Naturalist*, 142:78–92.

Cody, M. and Overton, J. (1996). Short-term evolution of reduced dispersal in island plant populations. *J. Ecol.*, 84:53–61.

Damer, B., Marcelo, K., and Revi, F. (1998). Nerve garden: A public terrarium in cyberspace. *In Heudin, J.C. (editor), Virtual Worlds, Springer-Verlag*, pages 177–185.

Darwin, C. (1859). *On the origin of species*. John Murray, London.

Davidson, R. (1969). Effect of root/leaf temperature differentials on root/shoot ratios in some pasture grasses and clover. *Annals of Botany*, 33:561–569.

Ebner, M., Grigore, A., Heffner, A., and Albert, J. (2002). Co-evolution produces an arms race among virtual plants. In Foster, J., Lutton, E., Miller, J., Ryan, C., and Tettamanzi, A., editors, *Proceedings of the Fifth European Conference on Genetic Programming, Kinsale, Ireland*, pages 316–325.

Fick, A. (1855). Über diffusion. *Ann. Phys. (Leipzig)*, 170:59–86.

Foster, S. and Janson, J. (1985). The relationship between seed size and establishment conditions in tropical woody plants. *Ecology*, 66:773–780.

Grime, J. (1977). Evidence for the existence of three primary strategies in plants and its relevance to ecological and evolutionary theory. *Amer. Nat.*, 111:1169–1194.

Holland, J. (1975). *Adaptation in Natural and Artificial Systems*. University of Michigan Press, Ann Arbor.

Jacob, C. (1994). Genetic l-system programming. In Davudor, Y., Schwefel, H., and Maenner, R., editors, *PPSN III. The 3rd Int. Conf. on Evolutionary Computation. Jerusalem, Israel*, pages 334–343.

Jolliffe, I. (1986). *Principal component analysis*. Springer-Verlag, New York.

MacArthur, R. and Wilson, E. (1967). *The Theory of Island Biogeograph*. Princeton University Press, Princeton.

Mock, K. (1998). Wildwood: The evolution of l-system plants for virtual environments. In *Int. Conf. on Evolutionary Computation, Anchorage*, pages 476–480.

Mustard, M., Standing, D., Aitkenhead, M., Robinson, D., and McDonald, A. (2003). The emergence of primary strategies in evolving plant populations. *Evol. Ecol. Res.*, 5:1067–1081.

Ochoa, G. (1998). On genetic algorithms and lindenmayer systems. *Parallel Problem Solving from Nature V*, pages 335–344.

Perttunen, J., Sievnen, R., and Nikinmaa, E. (1998). Lignum: A model combining the structure and functioning of trees. *Ecol. Modell.*, 108:189–198.

Pianka, E. (1970). On r and k selection. *American Naturalist*, 104:592–597.

Prusinkiewicz, P. and Lindenmayer, A. (1990). *The Algorithmic Beauty of Plants*. Springer- Verlag, Berlin.

Room, P., Hanan, J., and Prusinkiewicz, P. (1996). Virtual plants: new perspectives for ecologists, pathologists and agricultural scientists. *Trends in Plant Science*, 1:33–38.

Shinozaki, K., Yoda, K., Hozumi, K., and Kiro, T. (1964). A quantitative analysis of plant form - the pipe model theory, i. basic analysis. *Jpn. J. Ecol.*, 14:97–105.

Sievanen, R., Nikinmaa, E., Nygren, P., Ozier-Lafontaine, H., Perttunen, J., and Hakula, A. (2000). Components of functional-structural tree models. *Ann. For. Sci.*, 57:399–412.

Stearns, S. (1992). *The Evolution of Life Histories*. Oxford Univ. Press, UK.

Steinberg, D., Sikora, S., Lattaud, C., Fournier, C., and Andrieu, B. (1999). Plant growth simulation in virtual worlds : towards online artificial ecosystems. In *Proceedings of the First Workshop on Artificial Life Integration in Virtual Environment, Lausanne, Switzerland.*

Thornley, J. (1998). Modelling shoot:root relations: the only way forward? *Annals of Botany*, 81:165–171.

Westoby, M., Falster, D., Moles, A., Vesk, P., and Wright, I. (2002). Plant ecological strategies: some leading dimensions of variation between species. *Annual Review of Ecology and Systematics*, 33:125–159.

Building artificial personalities: expressive communication channels based on an interlingua for a human-robot dance

John Bryden[1], David Hogg[1], Sita Popat[2] and Mick Wallis[2]

[1]School of Computing, [2]School of Performance and Cultural Industries,
University of Leeds, Leeds, LS2 9JT, UK
johnb@comp.leeds.ac.uk

Abstract

The development of artificial personalities requires that we develop a further understanding of how personality is communicated. This can be done through developing human-robot interaction (HRI). In this paper we report on the development of the SpiderCrab robot. This uses an interlingua based on Laban Movement Analysis (LMA) to intermediate a human-robot dance. Specifically, we developed measurements to analyse data in real time from a simple vision system and implemented a simple stochastic dancing algorithm on a custom built robot. This shows how, through some simple rules, a personality can emerge by biasing random behaviour. The system was tested with professional dancers and members of the public and the results (formal and anecdotal) are presented herein.

Introduction

Can the study of human-robot interaction lead to the development of embodied agents with emergent *artificial personalities*? We define an artificial personality as a machine that is (and was been intentionally built to be) socially interactive. Socially interactive robots should:

> "express and/or perceive emotions; communicate with high-level dialogue; learn models of or recognize other agents; establish and/or maintain social relationships; use natural cues (gaze, gestures, etc.); exhibit distinctive personality and character; and may learn and/or develop social competencies." (Fong et al., 2003; Dautenhahn, 2007)

Focusing on studying systems that express and perceive emotion, we must understand the principles of emotional communication. We focus here on non-verbal communication as much human-human communication is done through body language (Mehrabian, 1981): i.e., communication which is expressive in its nature. In the spirit of ALife research, we look for simple models that are hopefully applicable across a broad range of systems.

Considering expressive human movement, we take our inspiration from analysis of dance—an art-form of expressive human movement. Dance, and specifically expressive movement quality in dance, has been studied in detail (Laban, 1971; Laban and Lawrence, 1974). We therefore both explore the principles of dance to outline potential models of expressive communication and also test those models by embodying them in a dance context.

Attribute	Description
Simple	The protocol for communication between agents should use a tractable mechanism
Expressive	The protocol should be identifiable by humans as containing emotive human content
Embodied	The protocol should work in an embodied system—we use an improvisational dance

Table 1: The attributes required for our development of communication channels for an embodied artificial personality

We specify the important attributes for our models of communication channels in Table 1. These three attributes (*Simple*, *Expressive* and *Embodied*) are important in the production of an artificial personality. We outline three communication channels in this work and discuss them in light of the attributes given in the table.

From a broader scope, we also consider the relevance of the communication channels we have identified to an evolutionary or ALife perspective. Many artificial life projects look for *emergent communication* [see work by Quinn (2001), Marocco et al. (2003) and Nolfi (2005) for examples], where a new communication channel emerges in a system without any prior specification. Such emergent communication channels are interesting from an evolutionary perspective as they can define simple mechanisms by which communication can occur with very little extra function being developed. A sociobioligical definition of communication (Wilson, 1975) sheds further light on the topic:

> "Communication occurs when the action or cue given by one organism is perceived by and thus alters the probability pattern of behavior in another organism in a fashion adaptive to either one or both of the participants." (p. 111)

Category	Description
Body	Studies the way an individual uses their body joints to generate movements
Space	Studies the way an individual interacts with space outside the body
Shape	Studies the sorts of shapes made by individuals as they move
Effort	Studies the way an individual moves

Table 2: The four LMA categories.

Subcategory	Description
Space	Is the movement direct or indirect
Weight	Is the movement strong or light
Time	Are the movements quick or sustained
Flow	Are the movements under body control (bound) or are they allowed to flow

Table 3: The four subcategories of the LMA effort category.

Given this definition there need be no intentionality on the behalf of the sender to transmit a signal—thus a communication channel may emerge just from observations of another's behaviour. We consider whether the expressive channels of communication developed here are relevant as forms of emergent communication.

Dance background

The field of postmodern dance stands out as being particularly relevant to the direction of study outlined in the previous section. Other dance forms, such as ballet, focus on structured movements. Ballet dances are commonly formed from a grammar of dance positions called *Key Aesthetic Poses*. Expression is conveyed through changing the style of movement between each dance position. Alternatively, postmodern dance seeks to remove all syntax and structure from dancers movements (Banes, 2003). Dancers are taught to unlearn their usual movement vocabulary so they can move on a purely expressionistic level. Commonly, dancers improvise in pairs (or greater numbers) where they each either *copy*, *oppose* or *innovate* qualities from the other's movements. Since their movements are no longer consciously motivated they therefore become examples of emergent communication.

We look here at a method which is widely used for interpreting and understanding expression qualities in dance (Laban, 1971; Laban and Lawrence, 1974): Laban Movement Analysis (LMA). This has four main categories which are outlined in Table 2. As we are focused on movement quality, we looked more closely at the effort category, which has four subcategories outlined in Table 3.

In a modern improvisational dance context, the dancers make *offers* to each other through their movement quality. From an LMA perspective, movements can be classified and dances can be interpreted through this language. In improvisation, movement quality is often copied with occasional innovations and oppositions—this gives the dancers a sense of *performative merging*.

Other work

Previous work for generating expressive movements has focused on a computer model of the human arm and torso (Chi et al., 2000) and a computer model of a ballet dancer (Neagle et al., 2004). In both systems, key positions and times were defined for body parts and heuristics, inspired by LMA, were specified for movement between positions. An important factor was found to be the velocity profiles of movement (Neagle et al., 2004).

The analysis of expressive qualities of human movements has been attempted before (Castellano et al., 2007). This approach used features generated from many different movement characteristics (Acceleration, Contraction Index, Fluidity, Quantity of Movement and Velocity) taken from actors making gestures expressing one of four emotions (Joy, Anger, Pleasure and Sadness). Various *classifiers* were tested with the data in order to generate models which would identify the correct emotion using the features available. The most significant feature was found to be Quantity of Movement with the Contraction Index (the degree of contraction and expansion of the body) playing a minor role. However, the system was not able to classify all emotions accurately. The complexity of this approach is not compatible with our requirement that models be tractable (Table 1) because of the large number of features used by the classifiers.

The SpiderCrab system

Given analogues between LMA (see the Dance background section) and mathematical analysis, and success using it in the past, LMA was chosen to act as an *interlingua* for an improvisational human/robot dance. The SpiderCrab robot was chosen to embody an artificial personality developed within the requirements of Table 1. The design phase of the robot was done through embodying it (into human form) at dance workshops. This was, in part, to study the application of performance arts methodology to the design process (Bayliss et al., 2007). It was also useful to form a picture of how the robot may be capable of expressive behaviour. In this project we focused specifically on the development of a controller for the robot to explore its potential as an improvisational dance partner.

An overview of the system we developed is presented in Fig. 1. The Robot system has three subsystems: the *sensory input*, *robot controller* and *improvisation* subsystems. Both the sensory input and robot controller subsystems use the *expressive communication model*, which is a common framework, based on LMA, for classifying both input from the dancer and output to the robot. Decisions about how

the robot should react to different sensory inputs from the dancer were made by the improvisation subsystem.

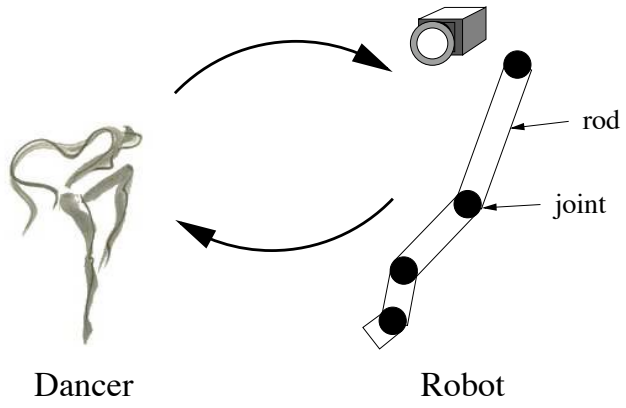


Figure 1: An overview of the SpiderCrab system showing expression flowing within a closed loop between the dancer and the robot. The robot system controls the robot, responding to the quality of the dancer's movement by biasing the robot's random movements. The dancer responds to the robot's movement.

The expressive communication model

The expressive communication model uses LMA to provide a common language for dancer and robot movement. For simplicity we focus on only one primary subcategory of LMA and analyse its role in the expression and communication of emotion. We chose the *weight* subcategory in the *effort* (see Table 3) category for several reasons. Previous work (Neagle et al., 2004) has shown that it is possible to generate movement in a virtual dancer in which humans can distinguish three emotions: Sadness, Happiness and Anger. These same emotions can also be distinguished when human dancers perform movements on different regions of the weight subcategory spectrum.

Two secondary subcategories of LMA were also considered, to try and understand their roles in the expression of emotion. These were the *space* subcategory (see Table 3) which is in the *effort* category and the *kinesphere* subcategory in the *space* category. The kinesphere subcategory relates to the area the dancer is moving within and how that relates to other dancers.

The two Effort subcategories were modelled as three discrete nouns (see Table 4). The kinesphere subcategory was modelled as a 3D coordinate which represented the position of the armband, and the 3D locations of the joints and rods of the robot limb.

Sensory input subsystem

A simple vision subsystem was used to generate real time data for our system. Some bright green material was fixed to the dancer (often as an armband) and its location was tracked by digital cameras. We recorded the centre of the green pixels at each timestep on each camera's image and this coordinate $[x(t), y(t)]$ was used to generate measurements for each camera.

LMA subcategory	Settings
Weight	Strong; Medium; Light
Space	Direct; Indirect

Table 4: The two subcategories of the LMA Effort category can have different settings in the expressive communication model. An LMA noun can be formed by choosing a setting from each subcategory, e.g., Strong+Direct would mean strong, direct movements.

Measurements were taken using the values of $x(t), y(t)$ to model the two LMA effort subcategories, weight and space. First, we propose that the LMA subcategory weight of movement may be modelled by the power delivered to the armband over a period of time of length T. This is approximated by assuming the mass of the armband is 1.0 (we do not use standard units). The force on the armband is thus equal to the absolute value of the acceleration of the armband at the camera frame. The power over time T is given by,

$$power(t) = \frac{1}{T} \sum_{t-(T-1)}^{t} Fl \ , \tag{1}$$

where F is the force on the armband at t and l is the distance travelled by the armband over the timestep at t. An alternative measurement was also considered, the average absolute speed over time T,

$$speed(t) = \frac{1}{T} \sum_{t-(T-1)}^{t} |s| \ , \tag{2}$$

where s is the speed of movement of the armband at time t.

The indirectness of movement was also considered to model the space subcategory. To do this, the direction of movement θ was calculated at each timestep. The rate of change of direction can be approximated by taking $d\theta/dt = \theta(t) - \theta(t-1)$. We introduce an indirectness measure over a period of time T which is given by,

$$indirectness(t) = \frac{1}{T} \sum_{t-(T-1)}^{t} \left| \frac{d\theta}{dt} s \right| \ . \tag{3}$$

Our $indirectness$ measurement is greater when the armband changes direction while moving quickly.

When the robot was in improvisational dance mode, the time period was set to two seconds for the $power$ measurement and one second for the $indirectness$ measurement. For evaluations of the Sensory input subsystem, movements of the dancers were broken down into gestures and the measurements were calculated for each gesture. Gestures were

Joint	Degrees of freedom	Rod length (m)
Shoulder	2	2.10
Elbow	2	1.30
Wrist	2	0.80
Finger	1	0.53

Table 5: The four joints of the SpiderCrab robot. The joints are connected in sequence. The Shoulder joint is fixed to the environment with each following joint connected by a sturdy rod.

identified by looking at the acceleration time trace: gesture start and end points were taken from when the acceleration moved from negative to positive values. Very short gestures (< 0.6 seconds) were combined together into longer gestures.

Tests with a simple 2 camera setup did not produce accurate enough 3-dimensional locations for the $power$ and $indirectness$ measurements. So we estimated motion orthogonal to the viewing direction of each camera using position in the image and assuming a fixed depth. This is a reasonable approximation assuming the dancer would stay roughly the same distance away from the cameras and face the cameras, although it will not include contributions to power and indirectness that arise from motion in depth. Course grained 3-dimensional location information was generated for the kinesphere aspect of the Expression communication model by using a second camera and triangulating the position.

The robot and the robot controller subsystem

The robot[1] is a single limb with four joints, see Table 5. The robot was designed to interact with the public, so needed to be as light and flexible as possible. The four joints of the limb are thus moved using air muscles with each axis of rotation having a pair of air muscles—a flexion muscle and an extension muscle. Valves connected to the air muscles are computer controlled, letting air in or out and contracting or extending the muscle respectively. Sensors on each joint return its angle(s) to the controller.

The quality of movement of the robot is defined by one of the six LMA nouns given in Table 4. The current LMA noun is received from the Improvisation subsystem (see the next section for more details on the Improvisation subsystem).

To control the robot's quality of movement, three variables are changed according to the LMA noun received from the Improvisation subsystem: $joint_instruction_length$, $robot_movement_power$ and $joint_direction_consistency$. A pair of air muscles rotate the joint around an axis of rotation for a random period of time (selected from a flat distribution between 0 and 1) multiplied by the $joint_instruction_length$. The amount of air fed to the muscles (per second) is sampled randomly from a uniform distribution between 0.5 and 1.5 and multiplied by the $robot_movement_power$. At the end of each movement the joint either continues its rotation in the same direction or will reverse direction with a probability depending on the $joint_direction_consistency$. If at any time a joint rotates past a limit (commonly the maximum rotation of a joint), the rotation direction will be reversed.

The three variables were set by hand for each of the six LMA nouns. They were tuned by assessing the robot's movement by eye. The $robot_movement_power$ variable corresponded with the weight subcategory, and the $joint_instruction_length$ and $joint_direction_consistency$ variables corresponded with the space subcategory.

The other important aspect of the robot's movement is determined by the kinesphere subcategory of the LMA space category. Here the robot will either point the elbow joint toward the dancer's general location, or ignore the dancer's general location and move the elbow freely. When the elbow must point, it rotates toward the target with an angular velocity proportional to the target's angular distance from the rod extending from the joint.

Improvisation subsystem

The robot was designed to perform within a postmodern dance improvisation context. This means that the robot will embody the expressive communication model by interfacing between the sensory input module and the robot controller. The improvisation subsystem implements an improvisational dance by switching between three different modes: Copy, Follow-copy and Oppose. Table 6 describes the three modes.

Improvisation mode	Description
Copy	The robot movement quality directly copies the movement quality of the dancer (using the effort and space subcategories of the LMA effort category)
Follow-copy	As **Copy** but with the elbow joint pointing at the dancer (using the kinesphere subcategory of the LMA space category)
Oppose	The robot movement quality is the opposite to that of the dancer (using the effort and space subcategories of the LMA effort category)

Table 6: The robot responds to the quality of movement of the dancer by selecting an LMA noun depending on its improvisation mode.

[1]The robot was designed in partnership with, and built by, the Shadow Robot Company. See `http://www.shadowrobot.com` for further technical information.

The $power$ and $indirectness$ measurements were used to identify which LMA noun the dancer was using. When in copy mode the dancer's noun was output to the robot controller, when in oppose mode, the opposite noun (i.e., Strong $\rightarrow$ Light, Light $\rightarrow$ Strong and Direct $\rightarrow$ Indirect) was output to the robot controller. The robot cycled through the 3 modes spending 30 seconds in Copy mode, 30 seconds in Follow-copy mode and then a random number of seconds (between 5 and 15) in Oppose mode.

Essentially, the robot responds to three key elements measured from the dancer's movement: the weight of movement, the directness of movement and the location of the dancer. It responds by either producing movements with a similar quality, or by producing movements with an opposing quality.

Evaluation

The SpiderCrab system was evaluated from two perspectives. First, we focused on the sensory input subsystem to evaluate its capabilities of perceiving emotional quality in movements. Second, the full system was evaluated by members of the public and dancers from the Salamanda Tandem dance company.

Sensory input subsystem evaluation

We tested the Sensory input subsystem over two dance sessions. In the first session the dancer was asked to make the same gesture with different qualities of movement. In the second session the dancer moved freely making varying gestures to different qualities of movement. Evaluations are made with reference to the attributes required by Table 1.

In both sessions the dancer stood at a fixed distance from a single camera. Approximately 20 gestures were made for each movement class by both dancers.

In the first session, we considered whether the sensory input subsystem was capable of assessing the emotional content of a dancer's movement (the Expressive attribute in Table 1). Dance movements were taken from three different movement classes expressing the three different emotions: Sadness, Happiness and Anger. For each individual gesture, the power and speed measurements (see Eqs. 1 and 2) were calculated using the Sensory Input subsystem. Box plots of the power data, collected within each movement class, are shown in Fig. 2. Box plots of the speed data, collected within each movement class, are shown in Fig. 3.

Figure 2 shows that the power measurement is a good choice for the Sensory Input subsystem to distinguish between the three emotions expressed by the dancer. In fact, the mean absolute acceleration also worked well (not shown). We decided to work with the power measurement as it relates more closely to our sensations of moving in the three movement classes: in an *ad hoc* experiment, the perceived work done by our muscles when expressing the emotions correlated with the power recordings of Fig. 2. An

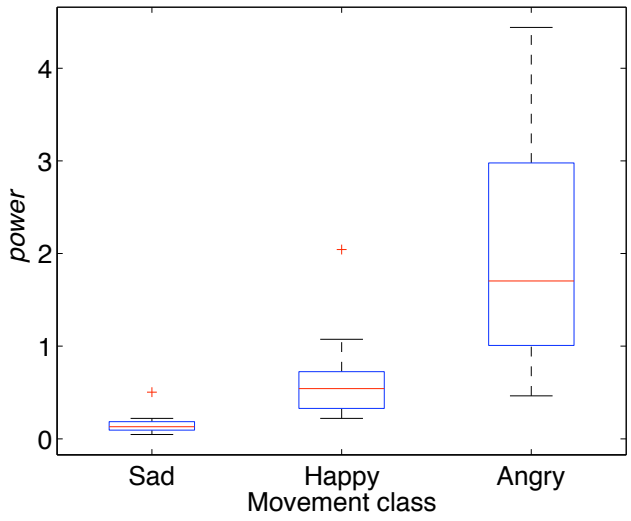


Figure 2: The same gesture made with movement quality expressing different emotions. The Sensory Input subsystem can distinguish between different emotional qualities of movement by calculating the power (see Eq. 1) delivered to the armband. Two-sample t-test comparisons between the three movement classes all give $p < 7.7 \times 10^{-05}$.

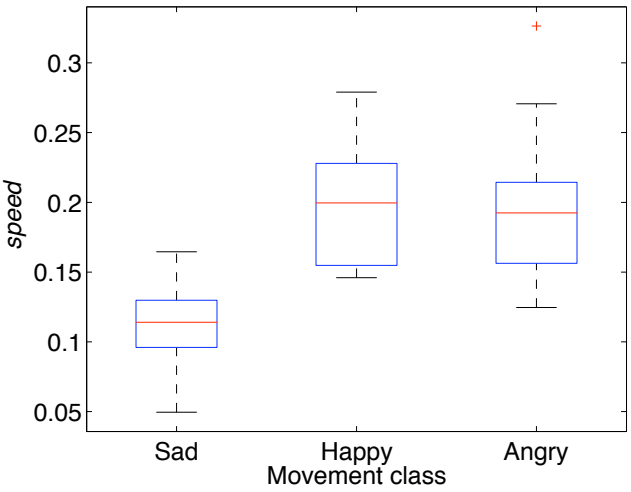


Figure 3: The same gesture made with movement quality expressing different emotions. The Sensory Input subsystem is unable to distinguish between the Happy and Angry gestures when calculating the average absolute speed (see Eq. 2) of the armband. A two-sample t-test comparison between Happy and Angry gestures gives $p = 0.82$

alternative measure of considering the average speed (or, by extension, momentum) of the armband did not distinguish between the Happy and Angry gestures (Fig. 3).

In the second dance session, a dancer was asked to perform gestures freely with relevant LMA nouns of the weight and space subcategories of the effort category. This formed four classes of movement: Strong+Direct, Strong+Indirect, Light+Direct and Light+Indirect. The output of the power measurement for the strong and light movements is shown in a box plot in Fig. 4. The output of the indirectness measurement for the direct and indirect movements is shown in a box plot in Fig. 5.

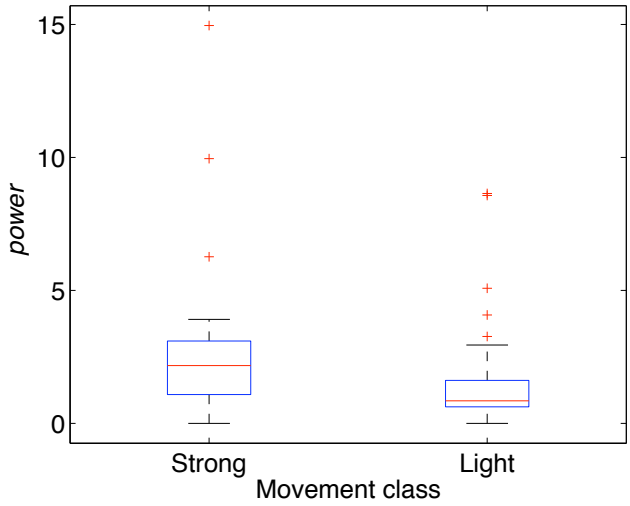


Figure 4: Varying gestures made with strong and light LMA movement qualities. The Sensory Input subsystem can distinguish the quality of movement by calculating the power delivered to the armband. A two-sample t-test comparison between the two movement classes gives $p = 0.018$.

The Sensory Input subsystem was able to distinguish between strong and light movements (see Fig. 4). In comparison to Fig. 2, some gestures were of a much greater power. Greater power can be delivered to the armband when the individual moves their body as well as their arm, rather than just the arm on its own. While the system had some success in distinguishing direct and indirect movements, the results were not significant. This was because it could not distinguish between an individual moving to start a new gesture (not relevant within and LMA context) and an individual moving within a gesture).

Full system evaluation

We evaluated the full system based on an embodiment test for an artificial dancer partner proposed by Wallis et al. (2007). This argues that "success will be measured by whether or not the human dancer feels that he/she is dancing with a true partner". With this in mind, we evaluated the

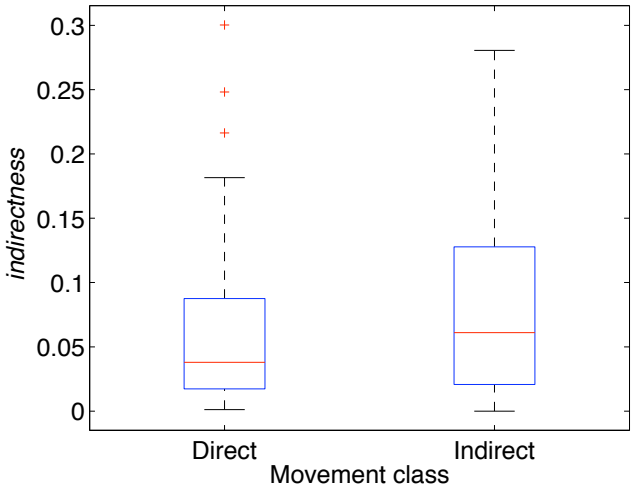


Figure 5: Varying gestures made with direct and indirect LMA movement qualities. The Sensory Input subsystem is unable to distinguish between the two movement classes. A two-sample t-test comparison between the two movement classes gives $p = 0.32$.

robot using a professional and independent dance company that focuses on improvisational dance. The company, Salamanda Tandem, use dance as a means of studying and developing social interaction—particularly with disabled people including those on the autistic spectrum.

Dancers from the company danced with the robot (see Figure 6) over a period of two days and wrote a report (Jones and Hood, 2008) on their interaction with the robot. So that their experience was not biased in any way, the dancers were told as little as possible about the way the system worked before starting to dance, just that it would respond to movements with the green armband.

Their responses to dancing with the robot indicate that it had passed the embodiment test: the robot did feel like a true partner. One of the assessors stated that

> "I felt apprehensive when approaching to move with the robot but it's amazing how quickly I forgot it was a robot and was just dancing with another, it felt friendly" (Julie Hood).

Another dancer felt that

> "[a]t first it's like a robot, then you forget and you are having a duet, getting to know someone—shaking hands... You can build a connection in play and be imaginative with it... It becomes a human limb" (Mickel Smithen).

In general, when the robot was in the copy mode people dancing with the robot (both professional dancers and members of the public) felt that it was responsive to their expressive offers. This meant that there was a bi-directional



Figure 6: A dance student dances with the robot.

expressive communication channel. When noise was introduced so that the system responded erratically (detecting movement when none was there), dancers felt it was less responsive.

Some comments picked up on how we might develop the robot's personality:

> "I've noticed in my work with people that there needs to be a pace, a sense of timing to encourage interaction to take place...I believe that SpiderCrab would need to be able to vary [its timing] in response to different sorts of people" (Isabel Jones).

Development along these lines could mean that the SpiderCrab system could become

> "a fantastic tool to deconstruct and analyse human interaction" (Isabel Jones).

After a while, the dancers started to feel that the robot's behaviour was becoming a little predictable and that its quality of movement was limited when compared with a human dancer.

The Sensory Input subsystem was shown, in the previous section, to be able to distinguish between movements expressing different emotions. When the robot was placed in an environment with real dancers, the robot was clearly able to pick up on the weight of the dancer's movement of the armband.

While there is a long way to go before the robot can fully identify the dancers' movement qualities, the six different qualities of robot movements were, however, qualitatively identifiable from each other. Reports from dancers were that robot movements could range from "menacing" to "smooth".

When the robot's elbow joint pointed toward the dancer's location, we had mixed responses. For safety reasons we had to slow the movement of the elbow. This meant that the robot was slow to copy the dancers' movements. Some dancers did not notice the difference between the copy mode and the follow-copy mode. However, when the follow-copy mode was observed, some dancers felt that the robot was crowding them whereas others felt that the robot was being more friendly.

Discussion

In this project, we have developed a robot system that, through improvisational dance, is capable of bidirectional expressive communication. The SpiderCrab system can distinguish between different human emotions, based on the quality of movement. Furthermore, the robot's movements were responsive to the dancers' movements and interpreted as such by the dancers. This meant that the robot was successful as an improvisational dance partner and was able to achieve social interaction [as specified by Fong et al. (2003); Dautenhahn (2007)] through the embodied expression and perception of emotions.

To consider the robot's potential as an artificial personality, we review the communication channels used by the system in light of the attributes outlined in Table 1. Starting with the primary communication channel, the power measurement (see Eq. 1) does indeed satisfy all the required attributes for use in an artificial personality. This measurement was both simple in that it can easily be calculated and used, but it is also simple in that it is clear how expression can be transmitted through the channel—it can map neatly onto human emotions (see Fig. 2). The success of the channel in the full system evaluation (including the fact that perception of the responsiveness of the system was impaired when noise was introduced) also means that it is successful as both an expressive and embodied channel.

The indirectness measurement (see Eq. 3) was less successful than the power measurement. While simple to calculate, the measurement did not map neatly onto human expression or emotions. However, when the robot generated movements within its more limited movement vocabulary, they were distinguishable to our eyes.

The pointing or following behaviour of the robot (based on the kinesphere LMA subcategory) showed some potential. This is simple to implement, is tractable, and maps neatly onto behaviour. The expressive quality of this communication channel relates to attention. The robot can use its physical location to pay attention to the dancer, or to move away to dance in a different space. It was difficult to implement in the full system due to the inaccuracies in our measuring systems.

As far as the production of artificial personalities is concerned, there is potential to build more sophisticated models into the robot's Improvisation subsystem. This could allow us to experiment more closely with different personality models and explore the system as a tool for analysis of human interaction. The development of a methodological approach for studying this in more detail is an exciting project.

Insights into emergent communication channels can also be gained from studying the communication channels we have outlined. Put simply, the power measurement measures the amount of energy being expended by an agent during its movement. Observers can quickly make judgements as to an agent's internal state based on this measurement and any other measurement that measures energy usage (e.g., sound volume, metabolic output, etc). It should be noted that it is also difficult for an agent to hide or fake its energy consumption so this forms the basis of a communication channel that is unintentional and emergent and therefore likely to be an early channel to evolve. Looking for other, unintentional, movement channels may well be productive.

Acknowledgements

The SpiderCrab project was part of Emergent Objects 2 (`http://www.emergentobjects.co.uk`), funded by the Engineering and Physical Sciences Research Council and Arts and Humanities Research Council through their Designing for the 21st Century initiative (`http://www.design21.dundee.ac.uk`). Thanks to Richard Walker and Matthew Godden for building the robot and to Royce Neagle and the dancers at the School of Performance and Cultural Industries at the University of Leeds for dancing with the robot. Dancing for Figs. 2 and 3 was done by Royce Neagle. Dancing for Figs. 4 and 5 was done by Sita Popat.

References

Banes, S. (2003). *Reinventing Dance in the 1960s: Everything Was Possible*. University of Wisconsin Press.

Bayliss, A., McKinney, J., Popat, S., and Wallis, M. (2007). Emergent objects: Designing through performance. *International Journal of Performing Arts and Digital Media*, 3:269–279.

Castellano, G., Villalba, S., and Camurri, A. (2007). Recognising human emotions from body movement and gesture dynamics. In Paiva, A., Prada, R., and Picard, R. W., editors, *Affective Computing and Intelligent Interaction, Second International Conference, ACII 2007, Lisbon, Portugal, September 12-14, 2007, Proceedings: LNCS, vol. 4738*, pages 71–82. Springer-Verlag.

Chi, D., Costa, M., Zhao, L., and Badler, N. (2000). The EMOTE model for effort and shape. In *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, pages 173–182. ACM Press/Addison-Wesley Publishing Co.

Dautenhahn, K. (2007). Socially intelligent robots: dimensions of human-robot interaction. *Philosophical Transactions of the Royal Society of London. Series B.*, 362:679–704.

Fong, T., Nourbakhsh, I., and Dautenhahn, K. (2003). A survey of socially interactive robots. *Robotics and Autonomous Systems*, 42:143–166.

Jones, I. and Hood, J. (2008). Spidercrab robot evaluation for PCI Leeds University. Technical report, Salamanda Tandem.

Laban, R. (1971). *The Mastery of Movement*. Plays, Inc.

Laban, R. and Lawrence, F. C. (1974). *Effort: Economy in Body Movement*. Plays, Inc.

Marocco, D., Cangelosi, A., and Nolfi, S. (2003). The emergence of communication in evolutionary robots. *Philosophical Transactions of the Royal Society of London: A*, 361:2397–2421.

Mehrabian, A. (1981). *Silent messages: Implicit communication of emotions and attitudes*. Belmont.

Neagle, R. J., Ng, K., and Ruddle, R. A. (2004). Developing a virtual ballet dancer to visualise choreography. In *Proceedings of the AISB 2004 Symposium on Language, Speech and Gesture for Expressive Characters*, pages 86–97.

Nolfi, S. (2005). Emergence of communication in embodied agents: co-adapting communicative and non-communicative behaviours. *Connection Science*, 17:231–248.

Quinn, M. (2001). Evolving communication without dedicated communication channels. In Keleman, J. and Sosík, P., editors, *Advances in Artificial Life: 6th European Conference, ECAL 2001, Prague, Czech Republic, September 10-14, 2001. Proceedings*, pages 302–311. Springer, Berlin.

Wallis, M., Popat, S., Bayliss, A., McKinney, J., Bryden, J., Hogg, D., Godden, M., and Walker, R. (2007). Spidercrab and the emergent object: Designing for the twenty-first century. In *Proceedings of dux07 (Design for User Experience 2007, Chicago)*.

Wilson, E. O. (1975). *Sociobiology: The New Synthesis*. Belknap Press / Harvard University Press, Cambridge, MA.

Basic Principles of Adaptive Learning through Variation and Selection

Mikhail S. Burtsev[1,2]

[1] Keldysh Institute of Applied Mathematics of RAS, 4 Miusskaya sq., Moscow, 125047, Russia
[2] P.K. Anokhin Institute of Normal Physiology of RAMS, 11/4 Mokhovaya, Moscow, 125009, Russia
mbur@ya.ru

Abstract

The evolutionary theory relies on the principles of variation and selection to explain adaptation. It is reasonable to fit these powerful principles to the learning theory. A number of selectionist approaches were proposed but found modest recognition so far. This theoretical paper attempts to review an application of basic ideas of the evolutionary adaptation to the lifetime learning. The analysis demonstrates that an adaptive value can be translated from the level of evolution to the level of individual through the innate repertoire of behaviors. This primary repertoire forms initial attractor for the behavioral dynamics. Learning starts when an environment offsets an organism from the existing attractor trajectory. Blind variations of behavior are generated until the return to the target attractor. These variations are retained to make up new branches of basin of attraction. It is important that the existed behavioral trajectory should not be altered as the learning unfolds because it keeps knowledge about adaptations survived selection through the evolutionary and learning history.

Introduction

Animals learn and this learning is usually beneficial or at least neutral for their evolutionary success. Generally, an adaptive learning is considered to be driven by some value system. The value system categorizes states of an environment in terms of their adaptive value. This categorization results in the feedback used in modification of behavior during learning. Adoption of the value system to explain the adaptive value of learning is not an exclusive solution. This paper addresses an application of the evolutionary principles of variation and selection to the explanation of adaptive outcome of learning. In spite of a number of selectionist theories of learning being proposed (Skinner, 1981; Edelman, 1987; Changeux and Dehaene, 1989) none of them gained widespread recognition. Here I try to analyze and clarify some basic ideas behind the selectionist approach. In particular, I focus on the issues concerning the initiation and finalization of learning, the selection criteria for the behavioral modifications, and memory retention.

An explanation of learning adaptive value is the ultimate problem of learning theory. Learning is adaptive when it leads to the modification of behavior that is evolutionary beneficial. But natural selection operates on the scale of generations and learning unfolds on the interval of minutes or even seconds. The evolutionary values should be transferred to the level of learning. This transfer is maintained by Darwinian evolution of developmental processes. And when an organism starts learning it already has criteria of adaptivness created during ontogeny.

Having representations of evolutionary values on the organismic level is a half of the story. The other half is generation of new adaptive behaviors. During learning an individual should produce behaviors taking into account evolutionary values. A mainstream of modern theories of animal learning in the fields of neuroscience (Schultz and Dickinson, 2000; Suri et al., 2001; Dayan and Balleine, 2002; Berridge and Robinson, 2003) and adaptive behavior (Maes, 1994; Sutton and Barto, 1998; Dorigo and Colombetti, 1998; Adaptive Behavior, 2002) employs "feedback" logic for aligning behavior with values. In this logic, discrepancy in the expected value guides learning. The value of an error signal is used to produce modifications of behavior.

An alternative approach to the generation of adaptation is presented by the explanatory scheme of the evolutionary theory. Evolution requires two processes, the first is generation of variation and the second is selection. The logic of the evolutionary explanatory scheme is *opposite* to the "feedback" logic mentioned above. In the "feedback" logic the adaptivness is evaluated first in terms of "reward" expectation mismatch and then the obtained error signal is used to change behavior. On the other hand, in the evolutionary scheme the generation of possible solutions goes first and then evaluation takes place in the form of selection. This reversal of stages leads to the next important distinction. The process of variation generation precedes the evaluation so it is *independent* of selection criteria, but in the "feedback" approach modifications *depend* on evaluations. The differences are outlined in the table 1.

"feedback" logic	**evolutionary logic**
evaluation then modification	generation then selection
modification depends on evaluation	generation is independent of selection

Table 1: The differences between "feedback" and "evolutionary" logic of adaptation.

It is obvious that the evolutionary logic of adaptation was successfully applied in a numerous studies to the synthesis of adaptive agents (Beer, 1996; Harvey et al., 1997; Pfeifer and Scheier, 1999; Nolfi and Floreano, 2000; Beer, 2000; Harvey et al., 2005) but its application to the problem of individual learning is still on the way.

There are a number of attempts to use principles of variation and selection in the theories of learning. In 1960 W. Ross Ashby published his influential book "Design for a brain" (Ashby, 1960) where one can find a proposal of the cybernetic theory of learning that utilizes trials and errors. Ashby had introduced so called essential variables (variables indicating viability of an organism) and treated them as a source of control for the blind variation. "Design for a brain" is focused mainly on the issue of behavior's stability. Here the important achievement was that adaptation in one behavioral subsystem should not disturb other subsystems of an animal and, hence, the subsystems for different behaviors should be loosely connected. Recently the similar idea of the structural adaptation was studied by Toussaint (Toussaint, 2004). Unfortunately, Ashby said nothing on retention of previous experience of an agent, and in his scheme of learning only the last successful adaptation is conserved. Therefore, each new learning episode should start from scratch and this would lead to the repetition of previous errors and make learning less effective. Also, Ashby allowed only a fixed set of essential variables to control blind variations.

On the conceptual level the evolutionary approach to learning lies at the intersection of evolutionary epistemology (Campbell, 1974; Popper, 1984) and constructivism (Glasersfeld, 1995; Foundations of Science, 2001). The famous Popperian formula describing the growth of knowledge through variation and selection is:

$$P_1 \rightarrow TT \rightarrow EE \rightarrow P_2, \quad (1)$$

here P_1 stands for the initial problem, *TT* are tentative theories or solutions proposed to solve it, *EE* is a process of error elimination, and P_2 is a new problem. For the sequential scheme of generation of solutions formula is extended and takes form:

$$P_1 \rightarrow TT_1 \rightarrow EE_1 \rightarrow TT_2 \rightarrow EE_2 \ldots \rightarrow TT_n \rightarrow success, \quad (2)$$

here *n* is a number of attempts which an agent has performed until solution was obtained.

An important contribution of the evolutionary epistemology is a concept of "vicarious selector" (Campbell, 1974). Vicarious selectors serve as internal representations of external factors of natural selection thus allowing transfer of evolutionary values to the level of learning. Vicarious selection "substitutes" natural selection during lifetime learning. The hierarchy of vicarious selectors accumulates all previous (even unsuccessful) experience and, as a consequence, generation of new behaviors takes the form of progressive growth on the top of existing competence. This process can be interpreted as a construction of individual knowledge by an active agent.

B.F. Skinner advocated his theory of "selection by consequences" (Skinner, 1981). Skinner considered selection by consequences as an explanatory scheme that is common for the three different levels, namely, the Darwinian evolution, learning and social evolution. Unfortunately, radical rejection of any attempts to consider the processes underling selection for all these cases made his approach fruitless.

There are two selectionist theories in the field of neuroscience: the theory developed by Changeux and Dehaene (Changeux and Dehaene, 1989), and the theory of neuronal groups selection (TNGS) proposed by Edelman (Edelman, 1987, 1993). Both theories declare thoroughly application of the "neural Darwinism" to the processes on all levels of brain organization from the synapse to the consciousness. The suggested sources of variation are generation of excessive synaptic connections during development and variable activation of neural assemblies. In their basic form both theories attributes selection to the input matching:

> "At a given stage of the evolution of the organism, some of these spontaneously generated pre-representations may not match any defined feature of the environment (or any item from long-term memory stores) and may thus be transiently meaningless. But, some of them will ultimately be selected in novel situations, thus becoming "meaning full". The achievement of such adequacy (fitness) with the environment (or with a given cognitive state) would then become the basic criterion for selection." (Changeux and Dehaene, 1989, p. 87)

The similar idea for the TNGS is presented in (Izhikevich et al., 2004). But input matching only is not enough for the creation of adaptive behavior because the action selection process should also be specified. The solution to this problem was put forward in (Dehaene and Changeux, 2000):

> "The models that we have introduced thus implement a generalized variation/selection scheme which was initially explored under the name of 'reinforcement learning' by computer scientists (e.g. Sutton and Barto, 1998) and has also been called 'neural Darwinism' by neurobiologists (Edelman, 1987, 1993; Changeux and Dehaene, 1989)."

This solution creates confusion because "generalized variation/selection scheme" refers to the evolutionary explanatory scheme but used by authors as a shortcut to the principles primary to reinforcement learning. The mechanisms of adaptation in the reinforcement learning fall in the domain of "feedback" logic which is opposite to the evolutionary one (see Table 1).

An integration of theoretical proposals devoted to application of the evolutionary principles to learning gives only some features of the required picture. Below an attempt to analyze the process of individual learning in the framework of evolutionary logic is presented and some consequences are discussed in relation to the current theoretical landscape.

Formalization

Below an organism or a robot is considered as an abstract adaptive agent. The agent's "brain" can be represented as an automaton **A** (similar approach was used by Peschl (Peschl, 1997) in his investigation of representations in neural systems). Description of the automaton **A** should include definitions of a set of all the components ***C*** of **A** and a transformation of these components states *B* that generates the "brain" dynamics.

The set of components ***C*** of the brain-automaton **A** consists of the following subsets (fig. 1):

• $\boldsymbol{C_E}$ is a subset of the components which states are determined by the external environment.
• $\boldsymbol{C_I}$ is a subset of components which states are determined by the internal state of the agent (its body).
• $\boldsymbol{C_S}$ is a subset of components which states are not determined directly by the external environment or the internal state of the agent but determined by any other components of the set $\boldsymbol{C}$.
• $\boldsymbol{C_A}$ is a subset of components which states determine the actions executed by the agent.

Hence a "brain" of an agent is specified by the automaton $\mathbf{A}\{\boldsymbol{C_E},\boldsymbol{C_I},\boldsymbol{C_S},\boldsymbol{C_A},B\}$.

$$\mathbf{A}\{\boldsymbol{C_E},\boldsymbol{C_I},\boldsymbol{C_S},\boldsymbol{C_A},B\}$$

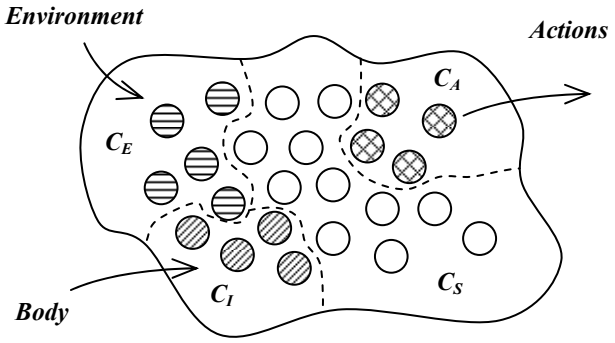


Figure 1: The automaton representation of the agent's "brain" (for details see text).

With the use of notation introduced above a behavior of the agent in discreet time $t\{t_0,t_1,t_2,\ldots,t_n\}$ is represented as modification of it's states by transformation B and can be written like this:

$$\ldots \boldsymbol{C}^{t_n} \xrightarrow{B} \boldsymbol{C}^{t_{n+1}} \xrightarrow{B} \boldsymbol{C}^{t_{n+2}} \xrightarrow{B} \ldots \qquad (3)$$

or equivalently

$$\boldsymbol{C}^{t_{n+1}} = B(\boldsymbol{C}^{t_n}) = B(\boldsymbol{C}_{\boldsymbol{E}}^{t_n}, \boldsymbol{C}_{\boldsymbol{I}}^{t_n}, \boldsymbol{C}_{\boldsymbol{S}}^{t_n}, \boldsymbol{C}_{\boldsymbol{A}}^{t_n}), \qquad (4)$$

here $\boldsymbol{C}^{t_n}$ is a vector of states of the automaton's components at the time t_n, or in other words the state of $\mathbf{A}$ at t_n.

The behavior of the agent is *constant* if the transformation B doesn't change with time. It should be mentioned that according to the equation (4) constancy of behavior is not necessary leads to the same actions in the same environmental and bodily conditions because the next state of the automaton also depends on the states of its internal components $\boldsymbol{C_S}$. It is reasonable to define learning as a transformation L of the brain dynamics B of the agent. Thus L is a transformation defined on the set of possible B's. To introduce learning into the dynamics the both B and L are applied:

$$\ldots \boldsymbol{C}^{t_n} \xrightarrow{L(B^{t_n})} \boldsymbol{C}^{t_{n+1}} \xrightarrow{L(B^{t_{n+1}})} \ldots, \qquad (5)$$

or

$$\begin{cases} \boldsymbol{C}^{t_{n+1}} = B^{t_n}(\boldsymbol{C}^{t_n}) = B^{t_n}(\boldsymbol{C}_{\boldsymbol{E}}^{t_n}, \boldsymbol{C}_{\boldsymbol{I}}^{t_n}, \boldsymbol{C}_{\boldsymbol{S}}^{t_n}, \boldsymbol{C}_{\boldsymbol{A}}^{t_n}) \\ B^{t_{n+1}} = L(B^{t_n}) \end{cases}. \qquad (6)$$

Addition of the learning transformation L to the automaton $\mathbf{A}$ results in an *automaton with learning* $\mathbf{A}\{\boldsymbol{C_E},\boldsymbol{C_I},\boldsymbol{C_S},\boldsymbol{C_A},B,L\}$.

"Feedback" logic of learning

The common assumption about logic of learning is that change in the function which generates a behavior (i.e. B) *is determined* by the states of the agent's "brain". These states can be external reinforcing stimuli which the agent percepts through the activation of some sensory inputs (i.e. $\boldsymbol{C_E}$), or the signals carrying information about the state of the body (i.e. $\boldsymbol{C_I}$), or the activations of pre- and post-synaptic neurons (i.e. $\boldsymbol{C_S}$) in the activity-dependent plasticity. In the framework of automata approach accepted in the paper this means that the transformation L is a function of values of states $\boldsymbol{C}$:

$$L = f(\boldsymbol{C_E}, \boldsymbol{C_I}, \boldsymbol{C_S}, \boldsymbol{C_A}). \qquad (7)$$

In other words, the transformation from B^{t_n} to $B^{t_{n+1}}$ is determined by the state $\boldsymbol{C}^{t_n}$ of the automaton $\mathbf{A}$.

Evolutionary logic of learning

Following the evolutionary logic the adaptation is produced by variation and selection where generation of variation is independent of selection. One can start with the most radical assumption that a change in the function which generates behavior (i.e. transformation B) *is not determined* by the states of the agent's "brain". The simplest form of the learning transformation L in this case:

$$B^{t_{n+1}} = L(B^{t_n}) = B^{t_n} \otimes \xi, \qquad (8)$$

where ξ is a random process (noise) and $\otimes$ denotes acting upon B.

It is obvious that the learning transformation L in the form of (8) leads not to adaptation but to degradation of the behavior. If we look at (8) as on applying a mutation ($\otimes\xi$) to the strategy of agent's behavior (B) it becomes clear that an analog of natural selection is needed to make the process adaptive.

The natural selection acts on variation in a population of individuals. The key difference of the individual learning from the evolution is that in the former the agent cannot evaluate more than one of the different variants of behavior at the same time. The solution is that during the individual learning selection acts not on the variation in the population of behaviors but on the sequence of varying behaviors. Thus the rule of individual evolutionary learning is:
"Produce blind variations of the behavior until adaptation is obtained."

Here selection is implemented as a control of blind variation. The next question, what does control variation, or who does evaluate produced behaviors? It is naturally to

assume that evaluation of the behavior is performed by the agent itself. Then (8) becomes:

$$B^{t_{n+1}} = L(B^{t_n}) = B^{t_n} \otimes m(\boldsymbol{C})\xi, \quad (9)$$

where m is a magnitude of application of ξ to B. The value of m is determined by the state of the "brain" $\boldsymbol{C}$.

The logic of learning expressed in (9) can be summarized as follows:

1. The process of change in the behavioral strategy of the agent (learning) is not determined by the state of the agent's "brain" and has a form of blind variation of already existed behavior.
2. Amount of change through blind variation is not constant in time and is determined by the state of the agent's "brain".

Discussion

The formalization of learning as an evolutionary process presented in the previous section gives only general framework and it is insufficient for the modeling of animal learning or implementation of any learning algorithms for animats.

In the selectionist theories of learning the picture of generation of variation in the behavior seems to be straightforward. It is easily described and related to a number of mechanisms on the neuronal level such as a probabilistic pattern of connections formed during the development and spontaneous excitations of cells or assemblies. An understanding of the selection processes in the brain is a challenge.

Consider the evolution of a population consisting of agents equipped with **A** "brains". The brain **A** is a dynamical system and its dynamics can be represented as a trajectory in the phase space of possible values of the components $\boldsymbol{C}$. This trajectory determines a sequential unfolding of the agent's behavior (eq. (4)). When a new born agent has some innate behavior this behavior is represented by a primary repertoire of the trajectories. In the course of evolution the agents with adaptive sequences of actions will be selected. Hence, these innate trajectories represent the evolutionary beneficial or at least "safe" (neutral in respect to the natural selection) sequences of agent-environment interactions. Moreover, the set of innate trajectories is the only source of adaptive values for the learning process. For the behavior being adaptive these trajectories should define the target dynamical attractor. Then the goal of learning is creation of a basin of attraction for it. If the point in $\boldsymbol{C}$ which corresponds to the current state of the agent's "brain" **A** moves along the trajectory which is already "approved" by selection then no modification of the behavior (of the transformation B) is needed and $m(\boldsymbol{C}) = 0$ in eq. (9). But movement along the target trajectory might be disturbed. For example, instead of a "normal" transition

$$\ldots\{\boldsymbol{C}_{\boldsymbol{E}}^{t_n}, \boldsymbol{C}_{\boldsymbol{I}}^{t_n}, \boldsymbol{C}_{\boldsymbol{S}}^{t_n}, \boldsymbol{C}_{\boldsymbol{A}}^{t_n}\} \xrightarrow{B} \{\boldsymbol{C}_{\boldsymbol{E}}^{t_{n+1}}, \boldsymbol{C}_{\boldsymbol{I}}^{t_{n+1}}, \boldsymbol{C}_{\boldsymbol{S}}^{t_{n+1}}, \boldsymbol{C}_{\boldsymbol{A}}^{t_{n+1}}\}\ldots$$

which should result in the environmental feedback $\boldsymbol{C}_{\boldsymbol{E}}^{t_{n+1}}$ the agent's "brain" might receive an "unexpected" reaction from the external world $\boldsymbol{C'}_{\boldsymbol{E}}^{t_{n+1}}$. If the "unexpected" state of the "brain" $\{\boldsymbol{C'}_{\boldsymbol{E}}^{t_{n+1}}, \boldsymbol{C}_{\boldsymbol{I}}^{t_{n+1}}, \boldsymbol{C}_{\boldsymbol{S}}^{t_{n+1}}, \boldsymbol{C}_{\boldsymbol{A}}^{t_{n+1}}\}$ is not lying at any part of the "safe" target trajectory then adaptation is required. To start learning the magnitude of blind variation $m(\boldsymbol{C})$ should become positive to allow generation of behavioral variations. During learning a new part of the trajectory is creating which departs from the "unexpected" state $\{\boldsymbol{C'}_{\boldsymbol{E}}^{t_{n+1}}, \boldsymbol{C}_{\boldsymbol{I}}^{t_{n+1}}, \boldsymbol{C}_{\boldsymbol{S}}^{t_{n+1}}, \boldsymbol{C}_{\boldsymbol{A}}^{t_{n+1}}\}$. The process of learning ends up ($m(\boldsymbol{C}) = 0$) when "approved" trajectory is reached. When after learning the agent's "brain" will sometime fall again into the state which is equivalent to $\{\boldsymbol{C'}_{\boldsymbol{E}}^{t_{n+1}}, \boldsymbol{C}_{\boldsymbol{I}}^{t_{n+1}}, \boldsymbol{C}_{\boldsymbol{S}}^{t_{n+1}}, \boldsymbol{C}_{\boldsymbol{A}}^{t_{n+1}}\}$ it will already have a trajectory to follow and no additional modifications of the behavior will be necessary.

Now, the selection criteria for the learning as evolution can be summarized as:

"The sequences of the agent-environment interactions that lead to the target trajectory should be retained".

The primary repertoire of trajectories of the agent extends only to a limited fraction of possible dimensions of the phase space. Initially only deviations along these dimensions evoke the learning process and variations in all other dimensions are "don't matter". A new branches added to the initial target trajectory by lifetime learning extend it to a new dimensions forming a secondary repertoire. Then deviations in both the primary and the secondary repertoires are used for the learning initiation and finalization.

The innate and learned behavioral trajectories are an adaptive knowledge gathered trial by trial during the evolutionary and individual history; hence, losing them is losing evolutionary advantage. This raises a requirement to the process of learning, namely, that the growth of the new branches of the attractor should not change the existing traces.

This "behavioral trajectory" analysis brings some conceptual extensions in comparison to the other selectionist approaches to the learning as evolution.

Control of learning by deviation from the target behavioral trajectory is similar to the homeostatic adaptation controlled by essential variables suggested by Ashby (Ashby, 1960). However, the homeostatic control has no mechanism for the retaining of knowledge gained during the learning through trials and errors, when the same deviation of the essential variables occurs next time the procedure of adaptation should be repeated again. Thus, Ashbian theory addresses the question of sustainability of behavior but not of its adaptive modification.

Discussing the work of Ashby Di Paolo (Di Paolo, 2003) suggested a hypothesis that is very close to the "behavioral trajectory" scheme:

> "Habits, as self-sustaining dynamic structures, underly the generation of behaviour and so it is them that are challenged when behaviour is perturbed. An interesting hypothesis is that often when adaptation occurs in the animal world this is not because organismic survival is challenged directly but because the circular process generating a habit is." (Di Paolo, 2003, p.31)

In relation to the general conceptual framework of the evolutionary epistemology (Campbell, 1974; Popper, 1984) the scheme proposed in this paper contains the next level of details. Recognition of the divergence from the target trajectory allows the agent to detect the problem situation. New behavioral trajectories sinking into the target attractor are retained, so the behavioral attractor plays the role of vicarious selector.

The selection of behavioral sequences toward attractor seems, on the first sight, to be similar to the match/mismatch selection of the neural Darwinism theories (Edelman, 1987, 1993; Changeux and Dehaene, 1989). When the agent encounters an unexpected situation it detects mismatch between perception and internal state. According to the neural selectionism the internal state is transformed to match the input. No action upon the environment is needed, neuronal dynamics only is sufficient to do that. Contrary, in the trajectory paradigm the internal state is a valuable knowledge which is kept intact and actions performed to change the situation, i.e. the input toward target values.

The analysis presented in this paper deals with the phenomenological level of description of behavior and learning. At the next level, the issue of cellular mechanisms compatible with the evolutionary scheme of learning should be addressed. The rules of cells interactions should allow detection of the deviations from the target behavioral attractor and creation of new neuronal functional systems while preserving existed ones.

Summary

The theory of learning should have an explanation why learning normally results in evolutionary adaptive modifications. The common explanatory scheme for the adaptivness of learning is based on the "feedback" logic. In this scheme the reward system of an organism or an agent evolves by the natural selection to effectively evaluate stimuli in terms of their expected contribution to the evolutionary success. The error between predicted and received reward is used as a "feedback" signal for correction of the behavior.

Variation and selection principle provides an alternative and opposite explanation to the "feedback" logic (see Table 1). Applied to the lifetime learning it assumes that at the first new behavioral variants are produced and then selected to meet evolutionary demands.

A number of approaches to utilize the evolutionary logic for the explanation of learning were proposed but the theory is still not satisfactorily. An attempt to clarify and extend the basic ideas underlying these approaches presented in this paper resulted in the following contributions:

- The innate behavior shaped by natural selection brings evolutionary values to the level of learning. This innate behavior constitutes an initial target trajectory of the agent-environment interactions.
- The generation(variation)/selection cycle of learning starts from the critical deviation from the existing behavioral trajectory and stops when deviation is eliminated.
- During learning new behaviors are created by blind variations. The behaviors leading to the target trajectory are selected.
- New branches of the behavioral trajectory produced by learning are included in the target set and start to play the role in controlling generation(variation)/selection cycle.
- The existed behavioral trajectory should not be altered as the learning unfolds. It keeps knowledge about adaptations survived selection through the evolutionary and learning history.

Acknowledgments

I deeply acknowledge the anonymous reviewers for their thoughtful suggestions and comments. Thanks to Konstantin Anokhin for the critical discussion of the ideas underlying early versions of this paper, Valentin Nepomnyashchikh and Carlos Gershenson for thier useful comments and suggestions. This work was supported by the Russian Fund for Basic Research, project 07-01-00180.

References

Adams, P. (1998). Hebb and Darwin. *Journal of theoretical Biology*, 195:419-438.

Ashby, W. R. (1960). *Design for a brain: The origin of adaptive behaviour*.(Second edition). London: Chapman and Hall.

Beer, R.D. (1996). Toward the evolution of dynamical neural networks for minimally cognitive behavior. In P. Maes, M. Mataric, J. Meyer, J. Pollack, & S. Wilson (Eds.), *From Animals to Animats 4: Proceedings of the Fourth International Conference on Simulation of Adaptive Behavior*, 421–429. Cambridge, MA: MIT Press.

Beer, R.D. (2000). Dynamical approaches to cognitive science. *Trends in Cognitive Sciences*, 4(3):91–99.

Berridge, K.C., and Robinson, T.E. (2003). Parsing reward. *Trends in Neurosciences*, 26(9):507-513.

Campbell, D.T. (1974). Evolutionary Epistemology. In P.A. Schilpp (ed.), *The Philosophy of Karl Popper*, 413–463. LaSalle, IL: Open Court.

Changeux, J.-P. Dehaene, S. (1989). Neuronal models of cognitive functions. *Cognition*, 33:63-109.

Dayan, P., and Balleine, B.W. (2002). Reward, motivation, and reinforcement learning. *Neuron*, 36:285–298.

Dehaene, S., and Changeux, J. P. (2000). Reward-dependent learning in neuronal networks for planning and decision making. *Progress of Brain Research*, 126:217—229.

Di Paolo, E. A. (2003). Organismically-inspired robotics: Homeostatic adaptation and natural teleology beyond the closed sensorimotor loop. In: K. Murase & T. Asakura (Eds), *Dynamical Systems Approach to Embodiment and Sociality*, 19-42. Adelaide, Australia: Advanced Knowledge International.

Dorigo, M., and Colombetti, M. (1998). *Robot Shaping: An Experiment in Behavior Engineering*. Cambridge, MA: MIT Press.

Edelman, G. M. (1987). *Neural Darwinism: The theory of neuronal group selection*. New York: Basic Books.

Edelman, G.M. (1993). Neural Darwinism: selection and reentrant signaling in higher brain function. *Neuron*, 10:115-125.

Foundations of Science, (2001). 6 (special issue on 'Radical Constructivism and the Sciences').

Glasersfeld, E.V. (1995). *Radical Constructivism: A Way of Knowing and Learning*. London: Falmer Press.

Harvey, I., Di Paolo, E., Wood, R., Quinn, M., and Tuci, E. (2005). Evolutionary Robotics: A New Scientific Tool for Studying Cognition. *Artificial Life*, 11(1):79–98.

Harvey, I., Husbands, P., Cliff, D., Thompson, A., and Jakobi, N. (1997). Evolutionary robotics: The Sussex approach. *Robotics and Autonomous Systems*, 20:205–224.

Izhikevich, E.M., Gally, J.A., and Edelman, G.M. (2004). Spike-timing Dynamics of Neuronal Groups. *Cerebral Cortex*, 14:933–944.

Journal of Adaptive Behavior. (2002). 10(3-4).

Maes, P. (1994). Modeling adaptive autonomous agents. *Artificial Life*, 1(1-2):135-162.

Nolfi, S., and Floreano, D. (2000). *Evolutionary robotics: The biology, intelligence, and technology of self-organizing machines*. Cambridge, MA: MIT Press/Bradford Books.

Peschl, M.F. (1997). The Representational Relation Between Environmental Structures and Neural Systems: Autonomy and Environmental Dependency in Neural Knowledge Representation. *Nonlinear Dynamics, Psychology, and Life Sciences*, 1(2):99–121.

Pfeifer, R., and Scheier, C. (1999). *Understanding intelligence*. Cambridge, MA: MIT Press.

Popper, K.R. (1984). Evolutionary Epistemology. In J.W. Pollard (ed.), *Evolutionary Theory: Paths into the Future*, 239-255. John Wiley & Sons, Chichester and New York.

Schultz, W., and Dickinson, A. (2000). Neuronal coding of prediction errors. *Annual Review Neuroscience*, 23:473–500.

Skinner, B.F. (1981). Selection by Consequences. *Science*, 213(4507):501-504.

Suri, R.E., Bargas, J., and Arbib M.A. (2001). Modeling functions of striatal dopamine modulation in learning and planning. *Neuroscience*, 103(1):65-85.

Sutton, R.S., and Barto, A.G. (1998). *Reinforcement learning an introduction*. Cambridge, MA: MIT Press.

Toussaint, M. (2004). *The evolution of genetic representations and modular adaptation.* PhD thesis, Institut für Neuroinformatik, Ruhr-Universiät Bochum, Germany. Berlin: Logos Verlag.

Design Strategies for Open-Ended Evolution

Peter Cariani

Harvard Medical School
cariani@mac.com

Abstract

Open-endedness is an important goal for designing systems that can autonomously find solutions to combinatorically-complex and ill-defined problems. We distinguish two modes of creating novelty: combinatoric (new combinations of existing primitives) and creative (new primitives). Although combinatoric systems may differ in numbers of possible combinations, their set of possibilities is closed. Creative systems, on the other hand, have open-sets of possibilities because of the partial- or ill-defined nature of the space of possible primitives. We discuss classes of adaptive and self-modifying cybernetic robotic devices in terms of these two kinds of processes. We consider material systems constructed from genetically-directed pattern-grammars. Although spaces of accessible structures are closed, function spaces can nevertheless be open. Thus, genome sequence spaces and gene-product structure spaces are regarded as closed, while partially-defined, phenomic function-spaces are potentially open.

Introduction

Intuitively, much of the natural world appears to us to be open-ended in character. When we consider the origins and evolution of life, the appearance and evolutionary elaboration of immune and nervous systems, and add the possible concomitant emergence of consciousness, it is difficult to imagine how the universe of evolved structures, functions, and phenomenal dimensions might be predicted from basic physical laws alone. The world cannot yet be described in closed form: there are too many incommensurable categories, structures, functions, functional organizations, and material/phenomenal distinctions among them to achieve such a grand reduction.

Yet most of us also optimistically believe that a comprehensive theory of life is possible in the future once we fully understand the space of structural and organizational possibilities that physico-chemical systems afford. Perhaps even more optimistically, many of us also believe that once the neural codes and pulse computations that constitute the informational organization of nervous systems are understood, then we will be able to understand and predict the structure and contents of phenomenal experience. But even if the structure of experience is predictable from patterns of neuronal activity, its existence as an aspect of the world is still an emergent if it depends on evolution of particular kinds of complex organizations. If the phenomenal realm emerged over biological evolution, then even if it is closed under physical causation, the material world may nevertheless be irreducibly open-ended in its aspects, i.e. there is more to describing what goes on in the world than in terms of material process alone. Even leaving aside such deep ontological questions, for the foreseeable future, living organisms and nervous systems will remain systems whose structures and functions are only partially-defined for us, and therefore whose behaviors can therefore surprise us in unexpected ways. Until a predictive "theory of everything" is achieved, if one ever is, living systems will continue to appear to us to be capable of open-ended self-modification.

The Importance of Open-Ended Design

Open-endedness is an important goal for designing creative systems. Creative systems are needed when we face ill-defined problems that defy direct solution, when we don't know what observables (sensors, features) and actions (effectors) are needed, and how they should be coupled and controlled (coordinations, computations). In these cases, we want the system itself to come up with a solution that we have not in some sense foreseen (or we would design that solution by fiat). We therefore seek to design and construct devices that act autonomously to go forth into the world to interact with it, to modify themselves in some way in order to find solutions that we cannot already anticipate. Open-ended devices are critical if we are to build robots that autonomously construct their own meanings and artificial immune systems that automate the search for new pharmaceutical agents.

Reliability and Closure

Unlike naturally evolved biological organisms, at present most of our artefacts are designed and constructed to behave in completely reliable and predictable ways that efficiently satisfy our needs. When they are performing within specifications, their structures are well-defined and highly constrained; they are expressly designed *not* to surprise us. The physical hardware of the modern digital electronic computer is the epitome of reliable design – astonishingly complex computations are invariably carried out without error. We manifestly want to avoid surprises (errors) creeping into our computations. When errors do occur, our systems are designed to immediately terminate computations and to indicate that a failure has occurred. In these reliable real-world, finite computational systems, to the extent that we specify all aspects of our devices (structure, operation), we know all possible input-output behaviors. We can circumscribe this closed set of possibilities, and no novel states of input-out behaviors will occur that will lie outside this box.

Open-endedness, Novelty, and Epistemology

Open-endedness requires creation of novel entities. Novelty requires some degree of ignorance – if all the parts and laws of a finite system are perfectly known, then all of the system's possible states and behaviors are known. Novelty (and hence open-endedness) is simply not possible in discourses where one assumes an omniscient, complete, God's-eye view of the world (e.g. realist-materialist and platonic ontologies). By their inherent construction, such discourses in the axiomatic-deductive mode categorically disallow *de novo* creation of new primitives.[1] Effectively, omniscience in a given realm implies closure; partial-knowledge permits the possibility of open-ended surprise. Novelty *is* possible in discourses where a limited observer compares the observed behavior of a system with his/her predictive model of it, since processes unrepresented in the model can cause the system's behavior to deviate from expectations ("emergence-relative-to-a-model"; (Cariani, 1989; Cariani, 1992; Cariani, 1997; Rosen, 1985). We therefore believe that an epistemological stance is necessary when we confront problems involving novelty, creativity, open endedness, and emergence.

In order for open-endedness to be a meaningful and useful criterion for considering natural and artificial systems, it should be principled (not an *ad hoc* construction) and clear; we must be able to construct operational definitions that allow us to unambiguously determine whether a given system is open-ended or not vis-à-vis some criteria. In order for us to ask whether a system has produced novel behavior, we first must ask the question of exactly what are our expectations: "novel relative to what?" In practice, change must be measured relative to some state-of-affairs, some concrete set of expectations we have of the system's structure and organization. Although operational criteria have been developed for restricted kinds of emergent functionalities (see below), open-endedness is a broader and less easily defined attribute than either closure or emergence-relative-to-a-model, mainly because it deals in spaces of possibility rather than the circumscribability of sets of elements.

A simple example (Fig. 1.) is helpful in conveying the differences between closed vs. open-ended realms The set of all 6-digit permutations of digits 0-9 is well-defined and contains 6^{10} elements, which can be enumerated. The set of all permutation sequences of 6 arbitrarily defined objects, however, is ill-defined, because the number of possible objects is indefinite. As a result this latter set is unbounded, ill-defined, and open-ended – one can always augment the set by specifying 6 more objects. In the first case, the primitives are exhaustively described by their token-types; consequently, the set is well-defined and closed. In the second case, the space of possible primitives themselves are not well-defined, and therefore the set of possibilities is ill-defined and open. Like the set of all

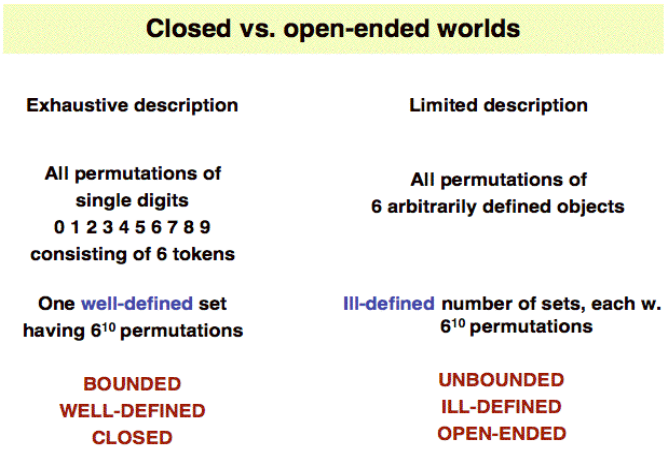


Figure 1. Closed vs. open sets of possibilities.

[1] In his 1975 debate with Jean Piaget about the possibility of new ideas (mathematical systems) appearing over (historical) time, Jerry Fodor famously, in platonic-realist fashion, argued for a closed universe in which there are no new, emergent ideas, but instead only selective fixation of previously existing ones (Fodor, 1980)

possible distinguishable objects, the set of possible measurements (observables) and actions that can be carried out respectively by sensors and effectors is ill-defined and open. This means that biological organisms and artefacts that are capable of evolving new sensors and effectors have an open-ended set of possible ways of interacting with the world, and, further, that the space of possible epistemic life-worlds, *umwelts* (Uexküll, 1925), is open-ended.

Combinatoric vs. Creative Novelty

One can envision systems that simply recombine fixed primitives vs. those that somehow create new ones. Emergent novelty can be generated in two ways: *combinatoric emergence* and *creative emergence* (Fig. 2). In a similar vein Lloyd Morgan (Morgan, 1931) distinguished "emergents" from "resultants": emergents being the result of novel creation, resultants, of novel combination. Both kinds of emergent orders are built up from basic sets of possibilities that constitute the most basic building blocks of the order, its "primitives." Emergence then entails either the appearance of new combinations of previously existing primitives or the formation of entirely new ones. The primitives in question depend upon the discourse; they can be structural, material "atoms"; they can be formal "symbols" or "states"; they can be functionalities or operations; they can be primitive assumptions of a theory; they can be primitive sensations and/or ideas; they can be the basic parts of an observer's model. To say that an entity is "primitive" relative to other objects or functions means it cannot be constructed from combinations of the others, i.e. its properties cannot be logically deduced from those of other entities. Thus, in this way of thinking, simple combinations of "lower-level objects" do not create "higher-level primitives" because the higher-level systems can be decomposed into yet lower-level objects (atoms).

Combinatoric Novelty and Closure

Combinatoric emergence assumes a fixed set of primitives that are combined in new ways to form emergent structures. This is very compatible with the way we often think about structure spaces, where parts can be combined to form larger structures. Thus in biological evolution, new genetic DNA sequences arise from combinations of pre-existing nucleotides, codons, and codon-sequences. Microevolution entails generation of novel combinations of genes; new genes arise through novel combinations of nucleotide sequences. Likewise, new, emergent structures are thought to arise

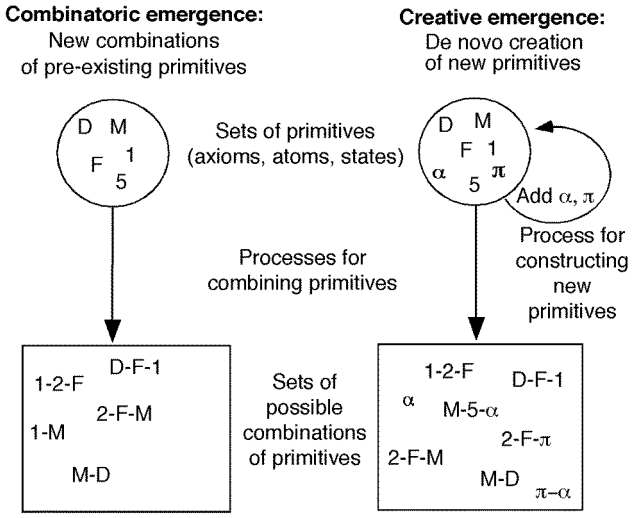


Figure 2. Combinatoric vs. creative emergence.

from novel combinations of previously existing molecular, cellular, and organismic structures.

This strategy for generating variety from combinations of relatively small set of primitive parts is a powerful one that is the basis of the systematicity of human and computer languages. Digital computers are ideally suited for generating combinations of symbol-primitives and logical operations on them that can then be evaluated for useful, interesting, and/or unforeseen formal properties. Correspondingly, in the realm of adaptive, trainable machines, directed searches optimize combinations of pre-specified features and actions (i.e. feature-action mappings, classifications). What formally distinguishes different kinds of trainable machines, such as neural networks or genetic algorithms, are the structures of the respective combination-spaces being traversed, and the rules that direct the search processes through them. In artificial life contexts, genetic algorithms using generative pattern grammars search through complex quasi-organic structure spaces[2] or find more optimal percept-action coordination strategies for simulated robots and organisms. In both types of applications, search spaces are large, but nevertheless closed.

Closure with Ill-defined Elements

We have argued above that well-defined finite sets are closed, while ill-defined, indefinite sets are open-ended. But what about sets of ill-defined elements? The genetic algorithms and pattern grammars mentioned above

[2] Dawkins demonstrated his Blind Watchmaker evolutionary graphics program at the first workshop on Artificial Life in 1987 at Los Alamos (Dawkins, 1987).

involve selection of well-defined, discrete entities (in the Blind Watchmaker program, these are discrete graphical elements; in a robotic controller, they are parameter values). However, combinatoric strategies can be used to select combinations of ill-defined parts that can interact in nonlinear, and unpredictable ways. Despite this incorporation of ill-defined elements, the set of possible behaviors is still closed under the set of discrete possibilities of the selection process. For example, if we had considered the set of 6-object permutations of 10 distinguishable, but ill-defined objects in the above example in Figure 1, the set of permutations would have 6^{10} members. Here, even though we don't know all the properties of the objects themselves, we can reliably treat them as individuals (name and distinguish them), and therefore draw a box around the space of possible permutations. We were not able to do this for sets of arbitrarily-defined objects, because there is no clear method by which we can clearly enumerate their elements or circumscribe their content.

Even if the set of possible combinations is closed, it may be useful to consider the respective cardinalities of two different systems as a comparative measure of structural complexity. In biological contexts, many different structural and functional criteria are possible: numbers of cells, cell types, expressed genes, protein conformations, metabolic states, informational states, etc. (Bonner, 1988). Complexity, however, does not by itself beget open-endedness. However staggeringly large the combinatorics become, mere number alone does not transform a closed set into an open one (finite, but large ≠ infinite, indefinite).

Ashby's Homeostat: Combinatoric Adaptivity

An apt historical example of combinatoric novelty using ill-defined elements is the homeostat of Ross Ashby (Ashby, 1960; de Latil, 1956). The homeostat consisted of four subsystems each in dynamic equilibrium with the others (Fig. 3). In each subsystem was a 25-position "uniselector" switch that determined the analog control parameters (capacitance, resistance) of that subsystem's electronic circuit. The circuits were "randomly" constructed and assigned to the uniselector positions, such that their structure and arrangement was not critical to the device's operation and might not even be fully known by the device's designer or user. The homeostat therefore had 25x25x25x25 (390,625) clearly defined uniselector-combination states that determined ill-defined analog control parameters and their associated behaviors. Particular combinations of parameters in interaction with a particular external signal could lead to stability or to chaotic instability.

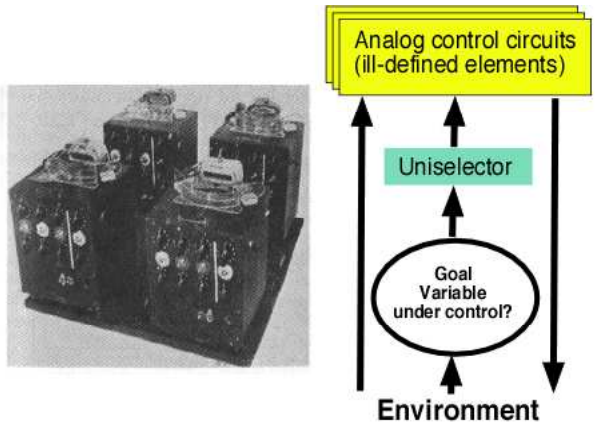


Figure 3. The homeostat and its operational structure.

The goal of the homeostat was to keep the value of a control variable near a given goal state, within specified tolerances. The homeostat thus evaluated whether a particular set of circuit parameters (resistances, capacitances) made a "good controller" vis-à-vis a particular environment. If the controlled variable did not achieve stability within some specified period of time, changing the positions of the uniselector switches would randomly choose another set of parameters to be tested.

The homeostat is a device that has no explicit model either of its environs or its internal workings. As de Latil says, "The homeostat works through the exploration of possibilities and the sifting of eventualities. The machine itself cannot 'know' the best solution of its problems, so it tries either systematically or at random, all possible solutions" (p. 308). Ashby also realized that not only could the homeostat be ignorant of the details, so could the designer: a designer need not understand at all how any of the analog controllers worked in order to choose which one worked better.

This use of constrained random search of ill-defined substrates is a departure from the dominant engineering philosophy of conscious, "rational" design, where designers are guided by some model of the processes they seek to control. The epistemic context of the homeostat is obviously the normal case in biological organisms and brains in homeostasis, learning and evolution – the parts of the system that do the selecting need not (and almost as a rule never do) have any understanding or model of the detailed processes they control. Biological evolution is blind in this sense, genetic mechanisms possess no anticipatory models of themselves or their environs that would guide which mutations would enhance survival and reproduction and which would not. But as long as one has a rich source of alternatives (high in variety), and an evaluative process that steers a selective mechanism, one can find solutions

to real world problems without understanding how they work or why they succeed. As long as a system is steerable, by selection or feedback, performance can be improved even if the agent steering the system has no model of the underlying processes that are being chosen or modified.

The homeostat may well have been the first artificial adaptive device to incorporate this principle of an "ill-defined" adaptive system, a principle that Gordon Pask was to carry to an extreme a few years later in his electrochemical assemblages (see below).

Limits of Combinatoric Novelty

Combinatoric novelty is a dynamic, creative strategy insofar as it constantly brings into being new combinations of elements. However, its use of fixed sets of primitive elements mean that the set of possible combinations is closed. In the example of Fig. 2, one cannot create new alphabetical letter types by stringing together more and more existing letters – the new notations must be introduced from outside the system by external agents or processes. Similarly, the homeostat could switch between 390k different circuits but it had no way of creating new circuits or of modifying existing ones to carry out new functions. Had the homeostat possessed the means of perturbing the structure of the circuits in an unforeseen way, say contingent on the structure of environmental input, then the device would have had an open-ended structure.

Within a computer simulation, all simulated activity occurs within the state-space and determined by the rules of the simulation program. However, if the observer is ignorant of the program, even partially, or if the computer is connected to unpredictable, external inputs, then novel behaviors vis-à-vis the observer's expectations *can* occur, and new primitives can potentially be created (e.g. a computer suddenly starts displaying Asian ideograms in addition to Roman text.) In such circumstances the computer's behavior would appear open-ended relative to the observer's set of expectations.

Creative Emergence and Open-endedness

Classically, "emergence" has concerned those processes that create new primitives, i.e. properties, behaviors, or functions that are not logical consequences of pre-existing ones. One can always ask how the particular primitives of an existing combinatorial system came into being in the first place. In explaining the origins of new primitives, one must appeal to additional processes that are not the primitives themselves. For example, how were the symbols depicted in Fig. 2 fabricated in the first place? By what process can new symbol types be added? In biological systems, how did nucleotide molecules strung together become the primitives of a genetic code?[3]

Primitive objects in the physical world almost always contain properties not fully known to the observer that can support new functions. These hidden aspects can come into play as primitives interact through the underlying material processes that subserve them. In this latter view, creating a new primitive entails the formation of a new property or behavior that in some strong sense was not predictable (by the limited observer) from what came before.

Open-ended Evolution of New Sensors

It is usually easier to give examples of qualitatively new functions than examples of qualitatively new structures. In our opinion, the most salient examples of the creation of new primitives involve the biological evolution of new sensory capabilities. Where previously there may have been no means of distinguishing colors, odors, or sounds, eventually these sensory capacities evolve in biological lineages. From a set of primitive sensory distinctions, one can list all combinations of distinctions that can be made with those primitives, but there are always yet other possible distinctions that are not on the list. For example, we cannot combine information from our evolution-given senses (sight, hearing, smell, etc.) to directly detect low intensity electrical or magnetic fields in our midst (as is achieved by electroceptive fish and some migratory birds, respectively). Creation of the ability to sense these fields through biological evolution, or artificial construction of measuring instruments (magnetometers, field strength sensors), thus adds new primitives to the set of perceptual distinctions that can be made.

Artificial Sensor Evolution

Artificial devices that create new perceptual primitives have been built. A perspicuous example is a electrochemical device that was constructed by the British cybernetician Gordon Pask in the late 1950's (Cariani, 1993; Pask, 1958, 1959, 1960, 1961). Its purpose was to show how a machine could evolve its own "relevance criteria." The structure of the heart of the analog device itself was hopelessly ill-defined. Current was passed through an array of platinum electrodes immersed in an aqueous ferrous

[3] A well-known paper by theoretical biologist Howard Pattee was entitled "How does a molecule become a message?" Dev. Biol. Suppl. 3:1-6, 1969.

sulphate/sulphuric acid equilibrium, such that iron dendritic filaments grew to form bridges between the electrodes. By rewarding iron structures whose conductivity contingently varied with environmental perturbations, the set of structures could be adaptively steered to improve the sensitivity of the whole. Pask's device acquired the ability to sense the presence of sound vibrations and then to distinguish between two different frequencies. In effect, the device had evolved an ear for itself, creating a set of sensory distinctions that it did not previously have. Albeit, in a very rudimentary way, the artificial device automated the creation of new sensory primitives, thereby providing an existence proof that creative emergence is possible in adaptive devices.

Evolvable Cybernetic Systems

Pask's device is a special case of a broader class of devices that are capable of modifying their own internal structure in open-ended ways. One can formulate a taxonomy of possible cybernetic devices and their creative capacities (see Cariani, 1989, 1991, 1998). These robotic devices consist of sensors and effectors coupled together by means of computational coordinative modules with well-defined internal symbolic states (Fig. 4). These devices have an evaluative part that directs the construction and modification of the hardware that subserves faculties of perception, cognition, evaluation & reward, and action. This hardware includes sensors, effectors, and the internal computational mechanisms that mediate sensorimotor coordination by implementing particular percept-action mappings. The evaluative part contains memory, learning, and anticipatory mechanisms for measuring performance, changing percept-action mappings, and adaptively modifying internal structures to improve performance. A methodology has been developed to distinguish between these functionalities and to determine when a new measurement, computation, or action is created. We believe they capture the basic operational structure of the observer-actor.

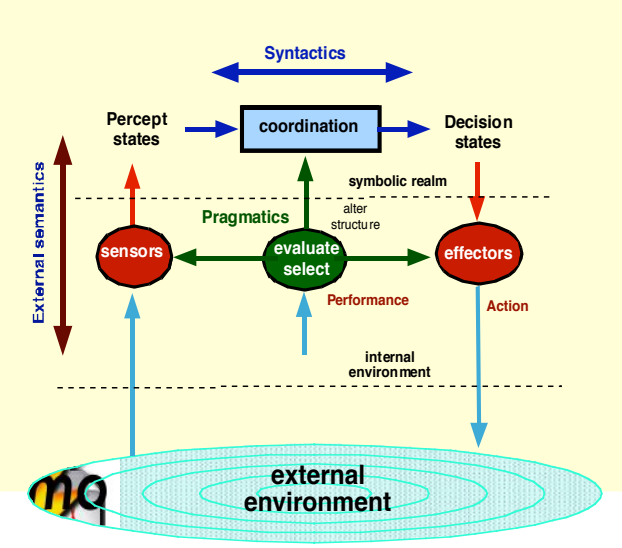


Figure 4. Self-modifying cybernetic devices.

Such cybernetic systems can be described in terms of semiotic categories: syntactic, semantic, and pragmatic dimensions. Syntactics describes rule-governed linkages between signs that are implemented in computational, coordinative portions of devices. Semantics involves the relation of signs to the external world, i.e. causal linkages between internal symbolic states and the world that are mediated by sensors and effectors. Finally, pragmatics involves the purposes for which signs are used: their relation to embedded goal states. Pragmatic relations are implemented by internal evaluation-reward mechanisms that adaptively steer or modify internal device linkages to better achieve embedded goals.

Within such a framework one can envision devices with both mechanisms that switch between existing sets of possible internal states (combinatoric emergence) or mechanisms that adaptively construct new hardware (e.g. new sensors, effectors, internal states) capable of creating new functional primitives (creative emergence). Table I summarizes possible types of adaptivity vis-à-vis combinatoric and creative emergence. In the syntactic realm, creative emergence produces new signs (symbols, internal states). In the semantic realm it produces new observables and actions that make new

Dimension	**Primitives**	**Stable systems** Fixed structure	**Combinatoric systems** Search/optimize existing possibilities	**Creative systems** Add possibilities Evolve
Syntactic	States Computations	Deterministic FSA's (fixed machines)	Change computations (trainable machines)	New states & rules (growing automata)
Semantic	Measurements Actions	Fixed sensors, effectors	Search combinations of existing sensors & effectors	New measurements and/or actions (epistemic autonomy)
Pragmatic	Goals	Fixed goals	Search combinations of existing goals	New goals (creative self-direction)

Table I. Combinatoric and creative emergence in cybernetic devices

contingent linkages between internal states and the outer world. In pragmatic realm, it produces new evaluative criteria (new goals).

Each functionality (sensing, effecting, coordinating) can be either be fixed, subject to combinatorial search, or capable of *de novo* creation of new primitives (Table I, above). In this scheme, combinatoric creativity involves new combinations of pre-existing input and output states, sensors, effectors, and goals. Creative emergence requires going outside of the set of existing functionalities to modify material structures ("hardware") in a manner that can create new states, new sensors and effectors, or new goals.

To the degree that a system has control over its own structure and functions, it attains a degree of freedom vis-à-vis both its environment and its own history. When a system can add to its own states and state-transitions, as in a growing automaton, it achieves some degree of computational autonomy. When a system can construct its own sensors, it attains a degree of epistemic autonomy. When it can construct new effectors it attains a greater autonomy of possible actions. Finally, when the system can construct its own set of evaluations and embedded goal states, it becomes self-directing.

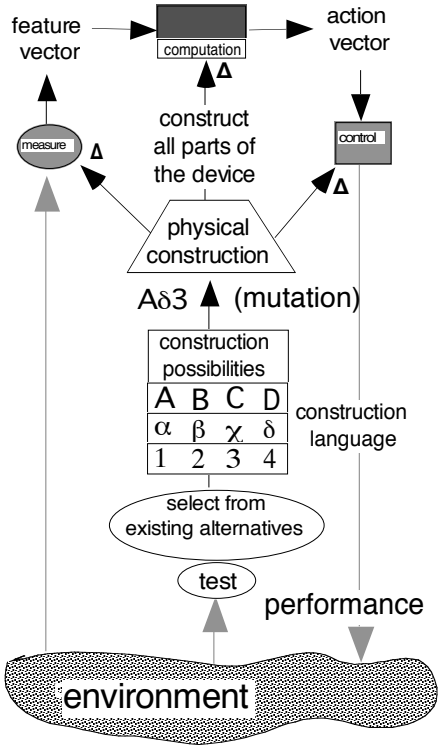


Figure 5. Evolutionary construction of cybernetic devices.

Genetic Construction and Closure

As in biological organisms, adaptive self-construction in these devices can be guided by genetic plans (Fig. 5). In these systems a genetic plan directs the construction of the material hardware of a device. This hardware consists of sensors that implement measurement operations, coordinative parts that implement computational mappings between sensory feature vectors and motor action vectors, effectors that carry out actions on the environment ("control" operations). The construction system also constructs itself and the evaluative sensors that determine which set of construction possibilities is actually realized. Thus, the construction system consists of a set of genetic plans that codes for a pattern grammar of possible material structures that will constitute the hardware that will subserve all the functionalities of the device.

The discrete genetic plans and the analog material hardware of these devices complement each other (Pattee, 1972). This functional organization of symbolic plans that constrain rate-dependent material processes utilizes both the combinatoric possibilities of discrete symbol systems and the open creative possibilities of analog dynamics. The symbolic part is well-defined, steerable, and inheritable but it is bounded by a set of fixed primitives, as was the case with Ashby's homeostat. The analog dynamics of the physical hardware are capable of creating new attractor basins that can subserve new functional states and operations. Pask's analog electrochemical device certainly had a richness of functional possibility, but there were no inheritable plans that could reliably save the process of constructing useful ferrous structures – each assemblage was a one-of-a-kind that had to be grown *de novo*. Genetic plans solve the problem of how to reliably access the rich possibilities inherent in the physical dynamics of matter.

These conceptual examples suggest strategies for open-ended design that involve coupling digital plans with analog dynamics. One needs a physical system that has rich dynamics with a large set of stable accessible states that can subserve useful functions of one sort or another. Means of steering the dynamics such that functional states can sometimes be obtained, are needed. Finally, reliable means of replicating the search for functional states need to be found, and these means themselves need to be controllable through inheritable, symbolic steering mechanisms. Once reliable control/construction structures are in place, then these can in turn be coupled to evaluative mechanisms that can steer the system towards particular goals. Once goals are connected to reliable construction/control processes, then one has an adaptive, self-organizing

system. If the system can be made self-replicating, then adaptation can also take place in parallel, amongst populations of systems over many generations. Closing the self-reproduction loop dramatically speeds up the search through genetic and phenotypic spaces. We should note, however, that natural selection itself does not create more variety; it alone does not expand the space of possible genetic sequences or phenotypic structures.

Although spaces of genetic possibilities are well-defined and closed in these systems, spaces of the phenotypic, hardware structures and attendant functions may nevertheless still be open if we have an incomplete description of their environments. In lieu of an exhaustive model of the environment and possible functions within it, phenotypic function spaces are almost always open because of the relational, contextual, environment-dependent nature of functions. If genetically-directed construction occurs independent of external contingencies, then the space of constructed phenotypic structures is closed (1:1 mapping of phenotypes to genotypes). However, if unknown environmental conditions co-modulate the "genetic expression" construction process ("epigenetics"), then the space of possible phenotypes becomes ill-defined and potentially open (>1 phenotype per genotype).

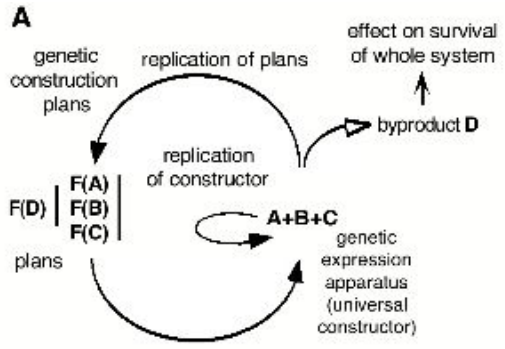


Figure 6. Schematic for von Neumann's kinematic self-reproducing automaton (von Neumann, 1948).

One can ask the analogous question of whether (or in what senses) biological evolution is "open" or "closed." John von Neumann's kinematic self-reproducing scheme (Fig. 6) captures the essence of relations between symbolic, inheritable plans, F(A)…F(D), and material products A…D as well as distinguishing those prodcuts involved in self-construction (A, B, C) from those "byproducts" that are not (D). While the set of possible genetic strings is finite and closed, epigenetic processes can open up somewhat the space of their associated gene-product structures. As with sensors and effectors, the space of intermolecular interactions and possible molecular functions is ill-defined and open-ended, at least until an exhaustive theory of biology is attained. In the meantime, we can reasonably regard biological genomes as closed symbolic realms capable of combinatoric novelty, and biological phenomes as partially-defined, material realms capable of producing both combinatoric and creative novelty in an open-ended fashion.

References

Ashby, W. R. (1960). *Design for a Brain*. London: Chapman and Hall.

Bonner, J. T. (1988). *The Evolution of Complexity*. Princeton: Princeton University Press.

Cariani, P. (1989). *On the Design of Devices with Emergent Semantic Functions.* Unpublished Ph.D., State University of New York at Binghamton.

Cariani, P. (1992). Emergence and artificial life. In C. G. Langton, C. Taylor, J. D. Farmer & S. Rasmussen (Eds.), *Artificial Life II. Volume X, Santa Fe Institute Studies in the Science of Complexity* (pp. 775-798). Redwood City, CA: Addison-Wesley.

Cariani, P. (1993). To evolve an ear: epistemological implications of Gordon Pask's electrochemical devices. *Systems Research, 10*(3), 19-33.

Cariani, P. (1997). Emergence of new signal-primitives in neural networks. *Intellectica, 1997*(2), 95-143.

Cariani, P. (1998). Towards an evolutionary semiotics: the emergence of new sign-functions in organisms and devices. In G. Van de Vijver, S. Salthe & M. Delpos (Eds.), *Evolutionary Systems* (pp. 359-377). Dordrecht, Holland: Kluwer.

Dawkins, R. (1987). *The Blind Watchmaker*. New York: Norton.

de Latil, P. (1956). *Thinking by Machine*. Boston: Houghton Mifflin.

Fodor, J. (1980). On the impossibility of acquiring "more powerful" structures: fixation of belief and knowledge acqquisition. In M. Piatelli-Palmarini (Ed.), *Language and Learning:The Debate Between Jean Piaget and Noam Chomsky* (pp. 142-162). Cambridge, MA: Harvard.

Morgan, L. (1931). *Emergent Evolution, 3rd Ed.* (Third ed.). New York: Henry Holt.

von Neumann, J. (1948). Re-evaluation of the problems of complicated automata -- problems of hierarchy and evolution. In W. Aspray & A. Burks (Eds.), *Papers of John von Neumann on Computing and Computer Theory (1987)* (pp. 477-490). Cambridge: MIT Press.

Pask, G. (1958). The growth process inside the cybernetic machine. In *Second International Conference on Cybernetics* (pp. 765-794). Namur, Belgium.

Pask, G. (1959). Physical analogues to the growth of a concept. In *Mechanization of Thought Processes, Vol II.* (pp. 765-794). London: H.M.S.O.

Pask, G. (1960). The natural history of networks. In M. C. Yovits & S. Cameron (Eds.), *Self-Organzing Systems* (pp. 232-263). New York: Pergamon Press.

Pask, G. (1961). *An Approach to Cybernetics*. New York: Harper & Brothers.

Pattee, H. H. (1972). Laws and constraints, symbols and languages. In C.H. Waddington (Ed.), *Towards A Theoretical Biology. 4 Essays.* (pp. 248-258). Edinburgh: Edinburgh University Press.

Rosen, R. (1985). *Anticipatory Systems*. Oxford: Pergamon Press.

On the ability of Swarms to compute the 3-coloring of graphs

Blanca Cases, Carmen Hernandez, Manuel Graña, Alicia D'Anjou[1]

[1]Grupo de Inteligencia Computacional
www.ehu.es/ccwintco
UPV/EHU

Abstract

Swarm Intelligent Systems are computational models of the spatial evolution of populations explaining it as a global behavior emerging from locally controlled movements, which are guided by decisions taken on the basis of local information. The increase of agents' cognitive capabilities, endowing them with memory and with the ability of selecting the rules of movement depending on an internal state allows the application of Self-Organizing Particle Systems (SOPS) to heuristic problem solving. Our focus in this work is on the complex emergent behavior arising when endowing the individuals with another elementary cognitive ability: the perception of the affinity of another individual. Each individual agent perceives other individual agents as friends or enemies. The first class is attractive while the second is repulsive. Metaphorically, the first class is associated with amity, security and comfort while the second is associated with danger, enemies and things to avoid. This local individual perception produces the emergence of teams and classes at a global level. This behavior produces an spatial distribution that can be interpreted under the appropriate metaphor as solving a particular computational problem. Applying this metaphor, we have found empirically that Self-Organizing Particle Systems can be designed to perform the task of 3-coloring graphs with the same precision as the Brélaz coloring heuristic, which is the best greedy heuristic known for this purpose.

Introduction

Emergent cooperation in biological systems is a central concept in Artificial Life from its very beginnings, researching for the ways in which a whole population of simple organisms is able to collectively perform a task (Nitschke, 2005). The natural social phenomena that inspired artificial life systems are swarms (Eberhart et al., 2001): flocking birds, fish schools, ant colonies, hives, or the pursuit and evasion behavior of predators and preys. The principal idea that underlied these works is the design of biologically inspired models, analyzing the emergence of collective behavior in terms of the local decision rules that govern the action of agents.

Emergent cooperation interest is not restricted to the domain of Artificial Life. Distributed Artificial Intelligence (Russell and Norvig, 1995) has gone in the direction of developing multi-agent systems able to solve problems by its collective behavior.

Swarm Intelligence (also called Self-Organizing Particle Systems (SOPS)) elements are agents geographically situated in a virtual environment. The emergent behaviors of interest for researchers are the ones showing collective navigation abilities or spontaneous clustering. These interests remain invariant from the first works of Reynolds (Reynolds, 1987, 1999) in computer graphics animation or the early applications to the navigation of teams of robots (G. et al., 2006; Lerman et al., 2001). On the other hand, Distributed Artificial intelligence is more focused in the local mechanisms of logical reasoning and conflict resolution over abstract spaces for knowledge representation.

Behind the approaches of Distributed Artificial Intelligence and Artificial Life to the design and simulation of (biologically inspired) intelligent social systems, Theoretical Computer Science has developed mathematical tools for the complexity analysis of collective emergent behavior. This field of Grammar Systems (Csuhaj-Varju et al., 1994) deals with a mathematical theory of agent cooperation arising from communication protocols modeled as grammars. Grammar Colonies (Kelemen and Kelemenov, 1992; Kelemenová and Csuhaj-Varjú, 1994) is a development in the framework of Grammar Systems closely related with Swarm Intelligence. It has been proved that a society of individuals equipped of a grammar generating finite languages are able to generate context dependent languages as if they possessed a collective "mind".

Recent research trends in Swarm Intelligence go towards the convergence of Artificial Life and Artificial Intelligence applying Self-Organizing Particle Systems to Problem Solving, by means of a mechanism that is basically the same used in Grammar Colonies: endowing each agent with a finite state machine that governs its inner flight rules depending on the current state. Adding a short term memory of visited positions is enough to design a system of two competing teams that collect minerals from some deposits transporting them to their respective homes (Rodríguez and Reg-

gia, 2004). This approach is called "designing for computing" in the self-design individual swarm-like agents whose problem-solving collective capabilities are proportional to the size of the population.

The research question guiding this work is the following one: what are the minimal cognitive capabilities that allow the emergent behavior of Swarms to solve NP-complete problems, such as the classical ones dealt with by classical and heuristic Artificial Intelligence algorithms, without mediating an explicit knowledge representation. We show in this paper that the simple distinction between friends (we) and enemies (them) in a population of boids is enough to produce an emergent behavior that can be interpreted as solving the problem of graph coloring. And they do it with a performance comparable to that of "traditional" algorithms.

We use the swarm metaphor to model the graph coloring problem as follows: agents correspond one-to-one with the nodes of the proposed graph. The graph topology defines the agent affinities as follows: the agents whose nodes are directly connected are "enemies", agents whose nodes are at graph distance[1] 2 are "friends". Agents are attracted to friends while try to fly from or to avoid enemies. The colors for the graph coloring correspond to specific attraction spatial regions. All agents are attracted to stay in these regions. Figure 2 shows the virtual space where the boids are moving around. The graph coloring solution is given by the distribution of the agents over the color attraction regions. When all the agents are placed in one of the color attraction space regions, the system configuration can be interpreted as defining a a complete coloration of the graph. When some agents are outside these regions, the system configuration corresponds to a partial solution to the coloration problem. To shake the system out from local optimal configurations corresponding to partial solutions, the agents are endowed of an aggressive instinct that allows them to overcome fear or repulsion to the enemies and to try to displace them from the privileged space regions.

We found that SOPS perform the task of 3-coloring graphs with comparable and sometimes better precision than the Brélaz coloring heuristic (Weisstein, 2008), which is the best greedy heuristic known for this purpose.

In the following sections we will first introduce Reynolds model. Next section describes in detail our metaphor to model the solution of the NP-complete graph 3-coloring problem through the swarm behavior. Next section summarizes some analytical results of the approach, trying to shed some light on the problem of determining the minimal cognitive capabilities that may have the agents to solve the problem of 3-coloring. Then we give some computational experiment results over a sample of hard colorable graphs. We end up with some discussion, conclusion and venues for further research.

[1]Graph distance means the length of the shortest path between two nodes. Unconnected nodes have infinite graph distance.

Description of Reynold's model

The idea of emulating of the movements and behaviours of societies of living beings from simple local rules that steer the individuals, giving rise to more complex global behaviours is a growing field of research, with applications in quite different domains. (Reynolds, 1987, 1999) was one of the pioneers in the simulation of the flight of flocks of birds.

According with Reynolds, each individual exhibits a very simple behaviour that is specified by a few simple rules that guide them to get along with the collective motion of the flock. The global behavior of the flock emerges from these individual decisions. We will stick to the birds metaphor, so that in the following, we call boids to the agents that compose a flock.

Each boid is aware of an spatial region around it, its neighbourhood. Given a set of n boids, the steering rules for i-th boid b_i, at time instant $t+1$ are defined as a function of the position p_j and the velocity y_j of the neighbouring boids at the previous instant t. The set of boids dwelling inside the neighborhood of the i-th boid is denoted:

$$\partial_i = \partial\left(b_i\right) = \left\{\, b_j \middle|\, dist\left(p_i, p_j\right) < \theta \right\}$$

where $dist$ is the euclidean distance. Let $|\partial_i|$ denote the number of boids in the neighbourhood.

The steering basic rules, used in our model, are the classical of Reynolds model: alignment, separation, and cohesion. Combining these rules, the flocking birds are able to flight co-ordinately avoiding collisions. The flocking rules for the boid b_i are formalized as follows:

- **Separation**: steer to avoid crowding local flockmates.

$$v_s = -\sum_{b_j \in \partial_i} \left(p_j - p_i\right)$$

- **Cohesion**: steer to move toward the average position c_i of local flockmates

$$v_c = c_i - p_i \;\text{ where }\; c_i = \frac{1}{|\partial_i|} \sum_{b_j \in \partial_i} p_j$$

- **Alignment**: steer in the direction of the average heading of local flockmates.

$$v_a = \frac{1}{|\partial_i|} \sum_{b_j \in \partial_i} v_j - v_i$$

Together with the elementary steering rules, the seek and flee rules, and the rules that define the behaviors of attraction towards friends and evasion from enemies.

- **Seek and flee**: seek attempts to steer a vehicle so that it moves toward a static goal. Here $\|p\|$ denotes the norm of

a vector p, and $f_{\text{maxvelocity}}$ is a non-negative parameter that limits the norm (the length in the Euclidean distance) of vector v_{seek}.

$$v_{seek} = v_{goal} - v_i \text{ where}$$

$$v_{goal} = \frac{p_i - p_0}{\|p_i - p_0\|} \times f_{\text{maxvelocity}}$$

Flee velocity is definided simply as the opposite of seek,

$$v_{flee} = -v_{seek}$$

- **Pursue and evasion**. These rules are a generalization of the seek and flee rules with the only difference of that the goal is non-static, and the boid moves toward or escapes from some of the flock mates.

Self-organizing particle systems for 3-coloring of graphs

Reynolds model is the basis of Self-Organizing Particle Systems (SOPS) for problem solving, introduced by (Rodríguez and Reggia, 2004). Following the authors, SOPS self-organization refers to the fact of that global behaviour emerges from the concurrent local interactions between particles in such a way that the whole population acts like an organism. Their contribution consists in incrementing the cognitive capabilities of agents by a Finite State Machine (FSM) controlling the boid flight rules. In this way, the system improves past SOPS, in which the boids do not have a significant intelligence. This is an interesting approach in that it represents an overture to Distributed Artificial Intelligence from the perspective of Artificial Life.

The paper by Rodríguez y Reggia presents a model formed of two teams of agents exploiting mineral deposits, carrying minerals to their respective homes. The agents have the following types of knowledge:

- Geographical: inherited from the Reynolds's model.
- Social: being able to recognize the neighbours as belonging to a category: friend or foe.
- Internal: the set of rules that control the movement of the agent, which are represented in a FSM, and a stack of visited positions.

From the point of view of Theoretical Computer Science, each individual has internally the elements that define a pushdown automata, which are able to generate flight trajectories in the virtual space. This enhancement significantly increases the power of SOPS to solve classical NP-complete problems. We focus on the 3-coloring of graphs.

Let $G = (V, E)$ be a graph (Borodin et al., 2005), composed of a set of vertices V and a set of non-oriented edges $E \subseteq V^2$. A *k-coloring* of G is a function that maps each vertex in $V = \{b_1, \ldots, b_n\}$ to a colour in the set $C = \{1, 2, .., k\}$ in a way such that two connected vertex have different colours. The problem can be formulated alternatively as minimizing the number of nodes that are badly coloured and even the number k of colours used. The chromatic number of a graph is the minimal number of colours needed for a coloration. The problem of k-coloring is NP-complete for all $k \geq 3$. We have selected 3-coloring as a benchmark problem because of the great difficulty that presents in spite of the simplicity of its formulation.

The representation proposed here is composed of a board of dimensions X_{max}, Y_{max} closed, as a torus, and $k \leq 4$ goals inscribed in a regular polygon representing each goal a colour in the set $C = \{1, 2, 3, 4\}$. Goals are static and attract the individuals with a pseudo-gravitational force.

A graph G represents a population of flocking birds. Each node b_i is an agent (a boid) whose initial position and velocity are drawn at random from a uniform distribution defined over the board. The graph represents the social network of the population. Two nodes $i, j \in V$ are enemies iff they are connected in the graph G, that is $(i, j) \in E$.

Hence, from the point of view of boid b_i, the set of boids actually inside its neighbourhood is partitioned into two subsets: the set of Enemies ∂_{E_i} and the set of friends, with an Amity relationship, ∂_{A_i}.

$$\partial_i = \partial_{E_i} \bigcup \partial_{A_i}$$

where $\partial_{E_i} = \{b_j : (b_i, b_j) \in E\}$. The Amity relationship is defined asbeing the "enemies of my enemies":

$$\partial_{A_i} = \{b_k : \exists j \ (b_i, b_j) \in E \wedge (b_j, b_k) \in E \wedge (b_i, b_k) \notin E\}$$

Velocity parameters

The model of 3-coloring has been implemented in Matlab 7. The velocity of each boid depends on three strengths that modulate its current velocity:

- **The Neighbourhood strength**: determined by a **Radius** around the boid with default **weight**$= 1.0$, pushing the boid toward the friends and away from enemies.
- **The Goal strength**: The seek strength, that attracts the boids toward the goals. The agent can seek for all the goals or only for the nearest goal with a default **weight**$=$ 1.0 in both cases.
- **The Attack strength**: if a boid can not reach a Goal or be sharing it only with friends after a number of steps it becomes "despaired" and attacks the enemies driving them from their positions:
  - **Internal attack**: displaces at random enemies occupying the same Goal it is lying in.
  - **External attack**: It is outside all of the Goals, since it has enemies inside all of them. The agent selects at random a Goal to head to and displaces an enemy from it.

A Matlab implementation of the process can be obtained from the authors and will be made public at http://www.ehu.es/ccwintco/. The magnitude of the velocity vector is globally limited by a parameter **Limitation of velocity** such that all the boids move with the same length step, set by default to 1.0.

The Social velocity We call Social velocity to the boid velocity component due to its repulsive and attractive interactions with the boids actually inside its neighbourhood. It is the sum of two components:

- **Enemity velocity**: taking as input the neighbouring enemies ∂_{E_i} we calculate a velocity

$$v_{E_i} = -w_a^{E_i} \times v_a^{E_i} + w_s^{E_i} \times v_s^{E_i} + w_e^{E_i} \times v_e^{E_i}$$

  where w's represents the strengths and a means alignment, s is separation and e evasion. This is the velocity term that corresponds to the alignment in opposite direction to enemies, separating and moving away from them.

- **Amity velocity**: taking as input the neighbouring friends ∂_{A_i} calculates a velocity

$$v_{A_i} = w_a^{A_i} \times v_a^{A_i} + w_c^{A_i} \times v_c^{A_i} + w_p^{A_i} \times v_p^{A_i}$$

  where p means the behaviour of pursuing its friends. There exists a strength of alignment, cohesion or pursuing the boids that compose the amity group.

The Goal seeking velocity This term models the need to get a colour for the node. In some experiments we do not activate it (see application interface in figure 2). For the purpose of solving the graph 3-coloring, we restrict the model to have at most 4 goals. The coordinates (X, Y) of the goals are given as the vertices of a regular polygon. The goals have influence inside a Goal Radius, and they attract the boids with a strength of 1.0 by default. Depending of the settings selected in the interface, the Goal velocity term is defined as:

- If the velocity to the nearest goal has been selected as a velocity parameter, the goal velocity is:

$$v_{Goal} = \frac{p_i - g_0}{\|p_i - g_0\|} \times f_{\text{lim itvelocity}}$$

  where g_0 is the position of the goal nearest to p_i.

- In the case of all the goals were selected,

$$v_{Goal} = \frac{\sum_{m=1}^{4} p_i - g_m}{\left\| \sum_{m=1}^{4} p_i - g_m \right\|} \times f_{\text{lim itvelocity}}$$

  Being g_m, $m \leq 4$, the positions of the goals.

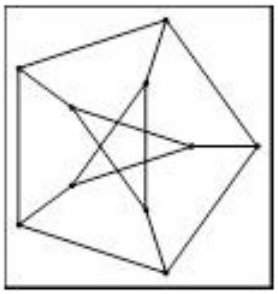


Figure 1: Petersen's Graph is 3-colorable. The system obtains a correct coloration with probability close to 1.0 for small graphs.

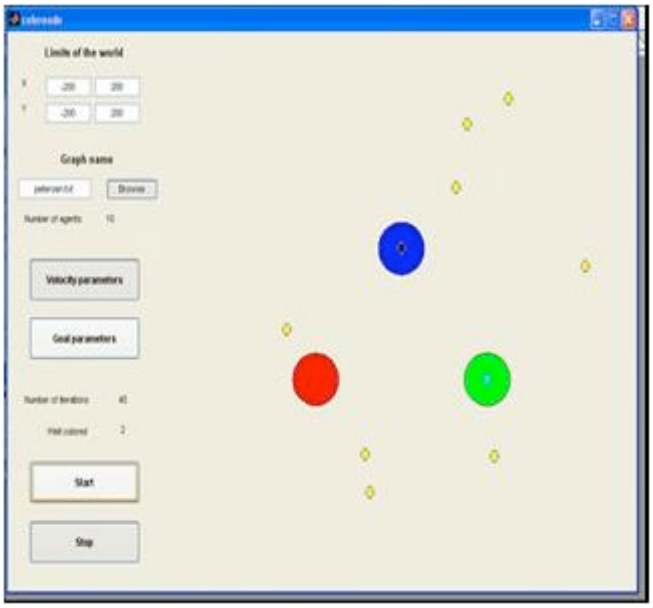


Figure 2: An snapshot of a run of 3-coloring SOPS on Petersen's graph.

Graph Coloring

The Matlab implementation allows to set the limits of the world and to load a file encoding a graph. Figure 2 shows an instant in the 3-coloring of Petersen graph, displayed in figure 1. This graph has 10 nodes. The interface shows an animation where the boids, which are represented as yellow small circles initially distributed at random, move toward the goals producing in this way a coloration.

Attack behaviour

The results produced by the application of the boids swarms to 3-colorable small graphs, like Petersen's graph, is successfull in the almost all of the cases when:

- The boid neighbourhood radius extends to the whole virtual world.

- The boids are attracted only to the nearest goal (instead of all the goals).

Without the attack mechanism, and using the default values for the Goal Radius and its strength of attraction, the system always converges either to an optimal configuration with all the boids situated inside the goals or to a sub-optimal one, with few boids wandering around of the nearest goal. This last situation occurs whenever the graph is non 3-colorable. Once a boid reaches a goal, it remains inside forever, being the goal a sink for agents' trajectories.

To shake the system away from (local minima) configurations that do not solve the coloring problem, we propose the Attack behavior, incorporated in a rule in the following deadlock conflict configurations:

- Internal: At least two enemies are situated in the same goal.
- External: An agent is wandering outside the goals because it has enemies in all of them.

To model attack we give to the agents an internal counter of the degree of "desperation" or "dissatisfaction" of the agent in a conflictive situation. Agents in a goal have an increasing degree of satisfaction over time. Whenever an agent enters in a conflict, the satisfaction level decreases as time goes, until the counter reaches a value below a given threshold, the aggressive behavior is activated and the boid attacks.

The attack consists in selecting randomly an enemy in conflict which is less desperate than the aggressor (its level of dissatisfaction is greater that the satisfaction of the assailant agent. The boid under attack is expulsed from the goal and the aggressor takes its place. We have introduced a noise term in the velocity that helps to generate mildly erratic trajectories for wandering agents.

Modelling agents as Finite State machines

Following the patter of the proposal by (Rodríguez and Reggia, 2004), we present our SOPS model of 3-coloring in a top-down manner.

First, the introduction of a satisfaction counter can be represented by a FSM as the one shown in fig. 3, being k the maximum value for satisfaction and 0 the minimum. We can represent the whole automaton of the figure as a single state with dash border labelled with the level s of satisfaction.

In figure 4 we give the FSM specification of a boid. We have abbreviated the states with satisfaction of level i simply as $S > 0$.

Initially, the agents are wandering starting at a randomly drawn position and with maximum level of satisfaction. If the agent falls within the area of influence of a goal, the Goal seek behaviour is activated and the boid tends to remain inside it. The boids only come out from the goal if they are involved in conflicts that make the satisfaction decrease.

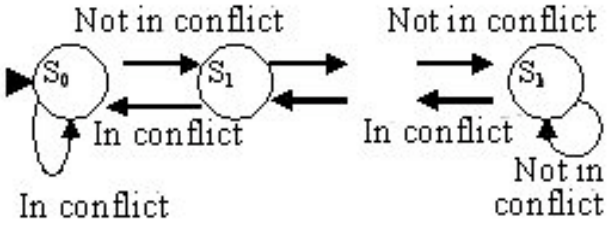


Figure 3: State S_0 means dissatisfaction while states S_i where $0 < i \leq k$ represent Satisfaction of level i.

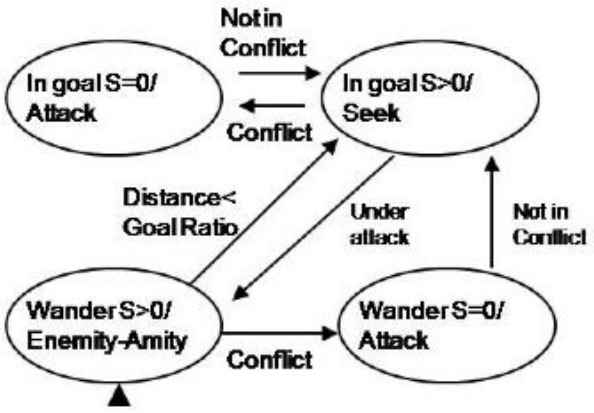


Figure 4: the FSM for a boid

If the agents suffers an attack, it passes to the wandering state, looking for a new goal.

If the agents reaches the state of desperation, $S = 0$, it attacks displacing another agent of the world and incrementing its satisfaction so long as conflicts disappear.

To obtain a faster convergence, we apply a cascade coloration strategy. Therefore, the execution of the program has two stages : First, the system attempts to find a 4-coloration of the graph situating 4 goals in the world. Once a coloration is obtained or after the maximum allowed time (1500 iterations) is elapsed, the second stage starts, eliminating the less populated goal. The individuals newly freed wander to seek a new goal until a 3-coloring is reached or the limit number of iterations (in this case 3500) are completed.

This procedure of cascading coloration is based on known works in reaction-diffusion particle systems (Turk, 1991) and is a way to extend the problem of 3-coloring of graphs. To find the chromatic number of a graph, i.e. the minimal number of colours that are necessary for a coloration is sufficient to start the process of colouring successively the graph

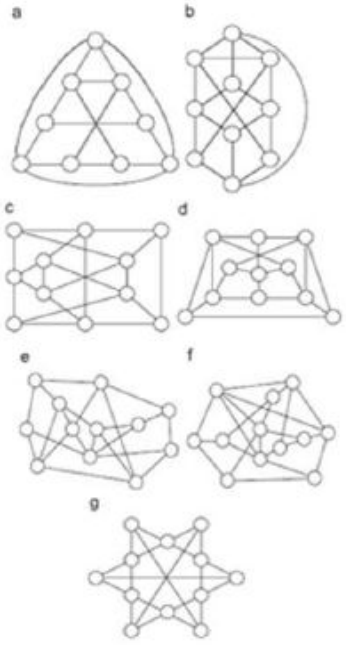


Figure 5: Several instances of hard 3-coloring graphs

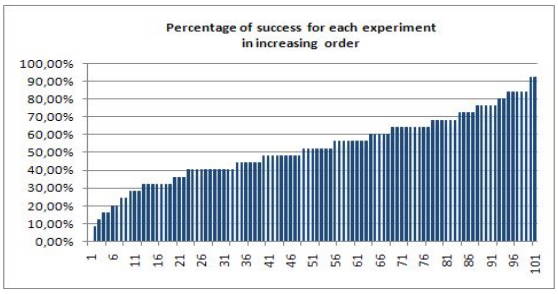


Figure 6: Success of SOPS 3-coloring

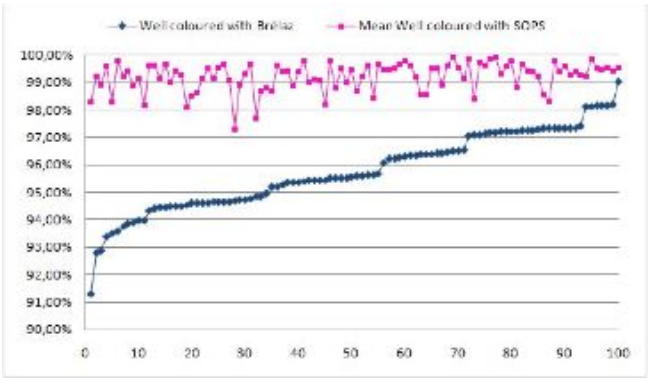


Figure 7: Percentage of well colored nodes for Brélaz versus Self-Organizing Particle Systems over the sample of benchmark hard graphs

with k, $k-1$, $k-2$,... colours until a minimal successful number of colours is reached.

Benchmarking Experiments: a comparison to Brélaz heuristic

The problem of 3-coloring of graphs has a very simple formulation but it is very difficult to solve. In 1979 Steinberg (Borodin et al., 2005) formulated a conjecture: every planar graph without 4 and 5-cycles is 3-colorable. In the last years, important advances has been made in the direction of proving Steinberg's conjecture. However, the problem of 3-coloring is NP-complete and the current research efforts focus on heuristics that may give good approximations to a global optimal solution in polynomial time. The best known heuristic for graph coloring is the Brélaz algorithm (Weisstein, 2008; Galinier and Hertz, 2006), which is a greedy algorithm that proceeds as follows: it performs first the coloring of the nodes with greater degree (number of connected nodes) and more constrictions (saturation), giving to each node the first available colour.

We have made some experiments to verify that SOPS algorithm is at least as precise as Brélaz algorithm, meaning that the chromatic number given by SOPS is less or equal than Brélaz chromatic number. It is well known that Brélaz heuristic works deficiently with some hard configurations for the 3-coloring (Mizuno and Nishihara, 2008). These authors present a graph building algorithm to construct hard coloring graphs. It performs graph embedding to combine basic hard configurations, given in fig. 5, into bigger hard graphs. For all of these graphs the Brélaz heuristic gives 4 as chromatic number, while all of them are 3-colorable. A sample of 100 graphs obtained from 10 random embeddings of the basic configurations were generated. For each graph, we have executed 25 runs of the SOPS algorithm registering the best configuration (we call this an experiment): Each run ends either when a 3-coloring solution is reached (success) or when 5000 iterations are completed in cascade.

The average results over all the experiments are: Mean number of nodes (boids): 110, Mean number of iterations: 3761, and Average of succeeding runs: 51%. In fig. 6 we have ordered the experimental graphs by the percentage of succeeding trials. This figure may serve as a model of the accumulative probability distribution of our algorithm obtaining a successful coloration over the sample of hard 3-colorable graphs. Note that if one execution of our algorithm obtains a 3-coloring, that constitutes a proof that the graph is 3-colorable. Note also that the Brélaz algorithm algorithm is deterministic, so that repeated trials have no sense for it.

For another look into relative performance we consider the following: if a node gets color number 4, then it is badly colored in Brélaz coloration. In SOPS, we register for each run the minimum number of individuals outside all the goals as the best configuration, and we say that this is the num-

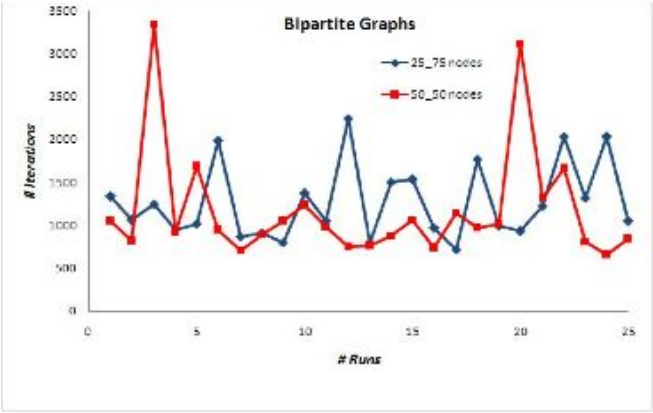


Figure 8: Number of iterations of 25 runs on the bipartite graphs

ber of bad coloured nodes for that execution. In each experiment (25 runs) the mean is taken. Figure 7 shows the percentage of well colored nodes for the whole sample. It can be appreciated that SOPS is always very close to the 100% well colored nodes, while some instances of Brélaz coloration are very poor. In order to discover if there exists a correlation between the variables, the sample has been ordered by increasing values of Brélaz algorithm. A correlation Pearson coefficient of 0.30 has been found and in consequence, correlation does not exists between the results. In average, Brélaz algorithm colorates well the 95,82% of the nodes with a standard deviation of 1.45%, while SOPS reaches a mean of the 99.17% and standard deviation 0.60%.

It is well known that Brélaz algorithm needs two colours for a bipartite graph, being particularly efficient in this case. To show that SOPS solves also correctly these problems, we have selected two complete bipartite graphs of 100 elements: the first with two classes of 50-50 nodes and the second with 25-75. In the 25 runs of each graph, the run was successful in both cases, being successful the 100% of the times. Regarding computing time measures, the mean number of iterations for graph 25-75 was 1266 and the minimum length of a successful run was 715. For graph 50-50 the average final step was 1172 being the minimum 654. Figure 8 shows the distribution of the number of iterations on the 25 runs of the SOPS algorithm for this graph.

Discussion

We have designed and implemented a Self-Organizing Particle System that may interpreted as solving the graph colouring problem. We addressed the problem of 3-coloration of graphs, but the cascading procedure of coloration presented before makes the extension to k-colorations be an immediate consequence. We chose the problem of 3-coloring graphs because of the important open questions around the problem: it is NP-complete and Steinbergs conjecture is giving arise an important research nowadays (Borodin et al., 2005).

A recent biologically inspired approaches to this problem has used the ant colony optimisation approach (Dowsland and Thompson, 2008), but we do not know of any other attempt to solve the problem using flocking birds. Their approach that identifies an individual in a population to a whole coloration of the graph, that is a tuple $(u_1, ..., u_n)$ where u_i is the colour of node i, losing in this way the biological inspiration if favour of cognitive abstraction. On the other hand, our approach to the coloration of graphs is mainly geometrical, attending to the representation of the nodes of a graph as a flocking bird situated geographically. The solution to the graph coloring emerges from the whole population configuration, which means a great economy of representation, and of computational power needed to implement the approach. The geometrical approach can be a source of experimentation and inspiration to improve sequential algorithms and heuristics for 3-coloration, which is important from the point of NP-completeness.

Second, we do not proceed in the direction of creating a model of colouring graphs from an existing model. Our aim was the research of the behaviour arising from endowing the individuals in a swarm with another elementary cognitive ability: the perception of the affinity of another individual. The individual perceives another individual as belonging to We or to Them. The first class is attractive while the second is repulsive. The first class is associated with amity, security and comfort while the second is interpreted as danger, enemies and things to avoid . We found that amity- enemity dynamics allows to model the solving process for coloring graphs, and not the other way around.

The third contribution of this paper has to do with the complexity of swarms, understood as the complexity of the behaviour of the emergent super-organism with respect to the computational capabilities of individuals. This work has been made in the last years in the field of theoretical computer science (Csuhaj-Varju et al., 1994; Kelemen and Kelemenov, 1992; Kelemenová and Csuhaj-Varjú, 1994). We have attempted to discover the lowest computational capabilities of individuals that allows the swarm to perform a coloration of a graph. Revisiting the work of Rodriguez and Reggia (2004) may lead a strong theoretical basis for furthers developments in the convergence with grammar systems.

The experimental results on hard coloring graphs with known chromatic number 3, show that the proposed approach can be very effective and competitive with state of the art algorithms. The Brélaz algorithm algorithm is the common benchmark algorithm. Our approach improves on it over a sample of hard graphs.

Acknowledgements

The MEC partially supports this work through grant DPI2006-15346-C03-03. The Basque Goverment has

awarded a *Grupo de Investigación Universitario* award to the Computational Intelligence Group.

References

Borodin, O. V., Glebov, A. N., Raspaud, A., and Salavatipour, M. R. (2005). Planar graphs without cycles of length from 4 to 7 are 3-colorable. *J. Comb. Theory Ser. B*, 93:303–311.

Csuhaj-Varju, E., Kelemen, J., Paun, G., and Dassow, J., editors (1994). *Grammar Systems: A Grammatical Approach to Distribution and Cooperation*. Gordon and Breach Science Publishers, Inc., Newark, NJ, USA.

Dowsland, K. A. and Thompson, J. M. (2008). An improved ant colony optimisation heuristic for graph colouring. *Discrete Applied Mathematics*, 156:313 – 324.

Eberhart, R., Shi, Y., and Kennedy, J. (2001). *Swarm Intelligence*. Morgan Kaufmann, 1 edition.

G., B., D., P., and S., N. (2006). Distributed coordination of simulated robots based on self-organisation. *Artificial Life*, 12, n. 3:289–311.

Galinier, P. and Hertz, A. (2006). A survey of local search methods for graph coloring. *Comput. Oper. Res.*, 33:2547–2562.

Kelemen, J. and Kelemenov, A. (1992). A grammar-theoretic treatment of multiagent systems. *Cybernetics and Systems. An International Journal*, 23(6):621–633.

Kelemenová, A. and Csuhaj-Varjú, E. (1994). Languages of colonies. *Theoretical Computer Science*, 134(1):119–130.

Lerman, K., Galstyan, A., Martinoli, A., and Ijspeert, A. (2001). A macroscopic analytical model of collaboration in distributed robotic systems. *Artificial Life*, 7:375–393.

Mizuno, K. and Nishihara, S. (2008). Constructive generation of very hard 3-colorability instances. *Discrete Applied Mathematics*, 156(2):218–229.

Nitschke, G. (2005). Emergence of cooperation: State of the art. *Artificial Life*, 11(3):367–396.

Reynolds, C. (1999). *Steering Behaviors for Autonomous Characters*. http://http://www.red3d.com/cwr/papers/1999/gdc99steer.html.

Reynolds, C. W. (1987). Flocks, herds, and schools: A distributed behavioral model. *Computer Graphics*, 21:25–34.

Rodríguez, A. and Reggia, J. A. (2004). Extending self-organizing particle systems to problem solving. *Artificial Life*, 10(4):379–395.

Russell, S. and Norvig, P. (1995). *Artificial Intelligence: A Modern Approach*. Prentice-Hall, New Jersey, USA.

Turk, G. (1991). Generating textures on arbitrary surfaces using reaction-diffusion. *SIGGRAPH Computers Graphics*, 25(4):289–298.

Weisstein, E. (2008). Brelaz's heuristic algorithm.

The effects of periodic and continuous market environments on the performance of trading agents

Satpal Singh Chaggar[1], Jason Noble[1] and Dave Cliff[2]
[1]School of Electronics and Computer Science, University of Southampton
[2]Department of Computer Science, University of Bristol
jn2@ecs.soton.ac.uk

Abstract

Simulation experiments are conducted on simple continuous double auction (CDA) markets based on the experimental economics work of Vernon Smith. CDA models within experimental economics usually consist of a sequence of discrete trading periods or "days", with allocations of stock and currency replenished at the start of each day, a situation we call "periodic" replenishment. In our experiments we look at both periodic and continuous-replenishment versions of the CDA. In this we build on the work of Cliff and Preist (2001) with human subjects, but we replace human traders with Zero Intelligence Plus (ZIP) trading agents, a minimal algorithm that can produce equilibrating market behaviour in CDA models. Our results indicate that continuous-replenishment (CR) CDA markets are similar to conventional periodic CDA markets in their ability to show equilibration dynamics. Secondly we show that although both models produce the same behaviour of price formation, they are different playing fields, as periodic markets are more efficient over time than their continuous counterparts. We also find, however, that the volume of trade in periodic CDA markets is concentrated in the early period of each trading day, and the market is in this sense inefficient. We look at whether ZIP agents require different parameters for optimal behaviour in each market type, and find that this is indeed the case. Overall, our conclusions mirror earlier findings on the robustness of the CDA, but we stress that a CR-CDA marketplace equilibrates in a different way to a periodic one.

Introduction

The Continuous Double Auction (CDA) is a market institution that plays a fundamental role in the world economy. It is the principal trading format for commodity markets, equity exchanges, foreign exchange, and derivatives markets. Real-world examples of CDA-based markets include the NYSE and the Chicago mercantile exchange. Although we have a great deal of observational data on these markets, it would be both difficult and illegal to manipulate them experimentally. Our understanding of how CDAs work has therefore been greatly enriched first by the discipline of experimental economics (Smith, 1962), in which human subjects participate in economic games in the laboratory. More recently CDAs have been studied using the methods of artificial life: in agent-based computational economics (see Tesfatsion, 2002, for a review) the behaviour of a simulated market emerges from the interactions of many relatively simple trading agents.

Our particular interest is in how the temporal structure of a CDA can affect both overall market performance and the optimal strategies for agents participating in that market. We look at two variant CDAs: one is an explicitly periodic market in which there is a discrete trading period with daily opening and closing points; we refer to this as the day-based or periodic-replenishment (PR) market. The second variant involves a non-periodic or continuous-replenishment (CR) market which allows for trading without interruption. We refer to the continuous-replenishment variant of the CDA as the CR market. These two types of CDA have important real-world exemplars: most stock exchanges are day-based, for instance, whereas the global foreign exchange markets are continuous-time. Intuition suggests that these markets are significantly different playing fields. Our goal is to use an agent-based model to find out how different these two CDA variants really are.

Experimental economics

The motivation of experimental economics is to model economic phenomena using human participants in controlled laboratory situations. Smith (1962) conducted pioneering studies in which a small number of inexperienced human traders participated in a CDA and were able to reach a competitive equilibrium price and equilibrium quantity of a traded commodity. Smith derived a qualitative indication of the relationship of supply and demand curves in producing equilibrating transaction prices and presented results suggesting the replication of classical microeconomic theory, all from a surprisingly simple model.

Smith's studies are recognized as the standard modelling framework for CDAs and the simplicity of Smith's concept has been integral to its success. Recent research has focused on establishing the robustness of Smith's general findings and examining the fidelity with which these experiments reproduce phenomena from real CDA markets. The reproducibility of economic phenomena is important as it

means that on the one hand market makers (e.g., a regulatory agency setting up a new marketplace) can use these experiments to develop fairer and more robust market mechanisms. On the other hand, traders (and the operators or regulators of financial institutions) can use results from experimental economics to identify and exploit strategic niches in their existing marketplaces.

Computational economics

If we take the human traders of the experimental economics paradigm and replace them with programs representing different trading strategies, we get agent-based computational economics (ACE) (Gode and Sunder, 1993; Cliff, 1997; Tesfatsion, 2002). An important aspect of this research has been finding the simplest algorithm capable of producing equilibrating market dynamics in a similar fashion to human participants. Cliff (1997) introduced the Zero-Intelligence Plus trading agent (ZIP) as an algorithm with minimal intelligence that nevertheless produced market behaviour that was very close to that of human traders. ZIP trading agents are a modified version of an earlier agent known as ZI (Zero Intelligence), created by Gode and Sunder (1993). ZI traders are simply stochastic agents that announce random prices for bids and offers. ZIP is able to model CDA price formation based on an intuitive heuristic "decision tree" algorithm coupled with elementary machine learning techniques (Cliff, 1997).

Computationally lightweight autonomously adaptive ("intelligent") trading agents (such as ZIP) are extremely significant given the emergence of virtual market-places. On the side of the market designer, iterative economic simulations using ZIP allow experiments to be conducted faster and yield significant results insofar as the ZIP trader can be seen as a realistic model. On the side of financial institutions that act within the market there is an incentive to replace human traders with automated trading agents. A fair chunk of work in ACE modelling to date concerns the use of agents inspired by the ZIP architecture in CDA markets. Studies have concentrated on evolving more robust agents and trading strategies. A basic ZIP agent acting in a periodic-replenishment (PR) CDA market with fixed supply and demand curves (as in the classic Smith experiment) has been used by a number of authors as the *de facto* benchmark for demonstrating equilibrating price formation with artificial agents.

The impact of replenishment in markets

Past work using intelligent agents in CDA markets has rarely explored the importance of the replenishment schedule within the market model. Round-the-clock 365-days-per-year environments are emerging at a fast rate in the real world, and yet continuous-replenishment modelas are perhaps one of the least discussed CDA variants (Cliff and Preist, 2001) in experimental economics. The standard Smith CDA model is conducted over discrete intervals known as trading days, and the dynamics of the market are centered around this day-trading structure. As not all real CDA markets are periodic the applicability of a day-based model to these variants is dubious. In what we believe to be the first human-based experimental economics studies to address this issue, Cliff and Preist (2001) explored the effect of removing periodicity from the standard CDA model by allowing continuous trading — i.e., switched from PR to CR CDA models. Cliff and Preist's general conclusion was that the ability of a CDA market to reach an equilibrium price did not seem to be affected by the switch from PR to CR. However, due to the inherent difficulties in human experimentation, the sample size in these experiments is really rather small.

Experimental aim

Our goal is to look at whether PR and CR markets produce different trading dynamics, and ultimately we would like to examine optimal trading behaviour across a wide range of different replenishment structures of the marketplace. In this paper we directly extend the work of Cliff and Preist (2001) by developing both continuous- and periodic-replenishment markets with ZIP traders instead of humans. We are especially interested in potential differences between the two market types that may have been too subtle to be detected given Cliff and Preist's limited sample size.

Method

We wrote computer simulations recreating the methods of Cliff (1997) and Cliff and Preist (2001), which are both adaptations of the experimental economics methods of Smith and Williams (1983). The method is a static model of a continuous double auction: i.e., supply and demand curves are fixed, and market participants ("traders") each privately know how many units they are willing to trade and the cost or value of each of their units, but not the allocations of any other traders.

There are 22 trader-agents in the simulated market: 11 buyers and 11 sellers. Each individual agent is allocated a private fixed *limit price*. The limit price specifies, for sellers, the minimum price at which they can sell, and for buyers, the maximum price at which they can buy. The difference between an agent's limit price and the actual transaction price they may achieve for the commodity is their utility — "profit" for sellers, "savings" for buyers. Limit prices for each of the agents are different, i.e., the agents vary in how much the commodity is worth to them. Limit prices range between \$0.75 and \$3.25 as shown in figure 1.

At the start of the experiment the 11 buyers and 11 sellers enter the market, with the sellers each in possession of one unit of the commodity, and the buyers each seeking to purchase one unit. We refer to these units as the agents' *entitlements* to buy or to sell. A single experiment — in the standard, periodic-replenishment case — consists here of a

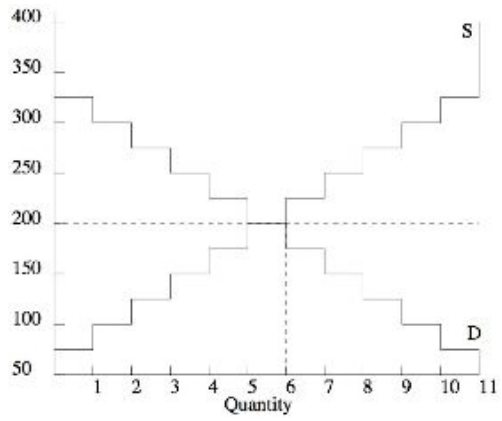


Figure 1: Stepped market Supply curve S and demand curve D, for 11 buyers and 11 sellers. Vertical axis is price in cents ($0.50 to $4.00); equilibrium price P_0 = $2.00. Supply and demand curves are fixed and symmetrical for all experiments. Figure is reproduced from Cliff (1997).

sequence of 20 trading periods, referred to as days. Each day is separated into 120 trading intervals (referred to as ticks). A tick is a discrete boundary of time at which a complete trading interaction can be executed (i.e., up to 120 attempted trades can take place during a day). Buyers and sellers normally have their entitlements reset to at the start of each trading day, by replenishing money to buyers and stock to sellers.

The arrangement of buyer and seller limit prices creates a stepped supply and demand curve for the imaginary commodity with a theoretical equilibrium price (P_0 = $2.00) and theoretical equilibrium quantity (Q_0 = 6) of units traded. Economic theory suggests that for rational agents participating in such a market, trading dynamics will show the competitive equilibration colloquially known as "the laws of supply and demand". In excess demand (trading taking place below the equilibrium) there is an incentive for the buyers to raise their bids to ensure they make a trade, and in excess supply (trading taking place above equilibrium) there is an incentive for sellers to lower offers to ensure a successful trade with a buyer (Cliff, 1997).

Trading process

With the market set up as described, buyers and sellers then engage in a CDA, in which they are free to announce and accept bids and offers for the commodity. The auction procedure is the same as that used by Cliff (1997).

1. At each tick a randomly selected agent quotes a price. This will be a *bid* if the agent is a buyer or an *offer* if the agent is a seller. The quoted price is made public to all agents from both communities and is the future transaction price for the trade. The agent's choice of price to quote is a function of its strategy.
2. Agents of the "contraside" (i.e., buyers responding to an offer, or sellers responding to a bid) make an assessment on whether dealing at the quoted price would be profitable for them. Again, this decision is a function of the agent's strategy. For ZIP agents, the decision will be influenced by their limit price but also by their current estimated valuation which is based on the recent history of successful trades in the marketplace.
3. If no willing agents are present in the market, i.e., the quoted bid is too low or the offer is too high, that tick-step is designated as a failed trade, and the market progresses onto the next tick.
4. If an agent decides that the shouted price is acceptable, it designates itself as a willing agent.
5. Prices of willing agents are arranged into a queue similar to NYSE rules (i.e., a trader makes a bid or offer at any time, but once made it is persistent until the trader alters it for a better price or it is accepted).
6. An agent is chosen from the queue, and the *quoted* price is the transaction price for the trade. The entitlements of both agents decrease by one and the profit and bank balances of the agents are adjusted according to the transaction price.
7. Finally, agents are assessed on their market activity state. Agents with no remaining entitlements to trade drop out of the market (although entitlements may later be reset, e.g., at the beginning of the next trading day).

A day's trading can be terminated prematurely if there are no active agents remaining in the market. Otherwise the market is open for 120 ticks, the duration designated for a trading day. For our markets we arbitrarily set the number of trading days to 20, to measure market performance over a reasonable period of time.

The periodic CDA

The replenishment schedule in a CDA market model effectively determines how and when the buying and selling entitlements of traders are reset. The periodic-replenishment (PR) variant is the default condition that has been described above; this is a replication of the Smith and Williams (1983) and Cliff (1997) models. The PR market forces the simultaneous and uniform renewal of all trading entitlements at the start of each day.

The continuous-replenishment CDA

For the continuous-replenishment (CR) CDA we recreate the market model of Cliff and Preist (2001) where there is no division of time into trading days. Once opened, the market continues for 2400 ticks until the end of the experiment. Every 120 ticks (the equivalent time frame for a day in periodic market) the entitlements for each agent are updated independently and with staggered phases. In short, the market is always open, and although agents temporarily drop out of trading after successfully buying or selling their single unit, they will return to trading at a randomly determined point in the future.

We have implemented two variations of the staggered renewal of agent entitlements, one referred to as *periodic continuity* or continuous(P) and the second referred to

Figure 4: Modelization of an example of the Gene Regulatory Network. A, B, C and D are 4 actions with their efficiency coefficient. The transfer coefficients are given by the arrows.

experiment consists in developing a system able to move substrates in the environment whereas the second one creates simple shapes like starfish or jellyfish.

To find the creature the most adapted to a specific problem, we use a genetic algorithm. Each creature is coded with a genome composed of three different chromosomes:

- The list of available actions, a subset of the environment possible actions. This list allows the cell to activate or inhibit some actions.
- The action selection system that contains a list of rule to apply actions.
- The gene regulation network that allows cell specification during duplication.

The creature is tested in its environment that returns the score at the end of the simulation. To increase the genetic algorithm power, we use a computational grid parallelized genetic algorithm. This parallelization allows the computation of hundreds of creatures at the same time.

Experiments

Developing a transfer system

The first experimentation consists in developing a simple organ : a transfer system. In other words, the cell structure must be able to transport substrate from one point to another. To do that, we imagine an environment composed of 2 substrates:

- The red is the substrate that must be moved by the organism. This substrate has the specificity not to spread in the environment, in order not to impact on the organism work.
- A gray that will be used by the cell as fuel and duplication material.

The cell can perform the following actions:

- duplicate (needs one gray substrate and vital energy),
- absorb or reject substrate (consume vital energy),
- transform one gray substrate in vital energy.

We place 10 red substrate units into a specific cross of the grid (at the top left of the environment) and diffuse gray substrate all over the environment. The creature's score is given by the squared sum of the red substrate distance to the goal point (at the bottom right of the environment). The parameters of the genetic algorithm are:

- selection: 7 tournament competition with elitism,
- mutation rate: 5%; crossover rate: 65%,
- substitution: worst individuals,
- population size: 500 individuals,

Figure 6 shows the convergence curve of the genetic algorithm. It shows the variation of the minimum, the average and the maximum fitness of the population for each generation. The genetic algorithm's aim is to maximize fitness, which is the creature score. A relevant organism appears quickly. After 3 generations, the organism is able to move the red substrates but not in the right direction. After 10 generations, it is able to move closer to the goal point. The genetic algorithm converges after 22 generations (the average fitness is close to the best).

Figure 5 shows the development of the best organism[1]. We can see that only the cells on the way from the initial point to the end point are created. Moreover, the organism uses absorption and rejection actions to transfer the substrate gradually. Cells that overtake the final point die quickly so as not to interact in the transfer. During the convergence of the genetic algorithm, it is interesting to observe the evolution of the organism strategy towards the best solution. The first step is to learn to survive in the environment, absorbing gray substrate and transforming it in vital energy. The next step is to learn to duplicate in the right direction. Intermediate solution organisms are able to transport the red substrate

[1] Videos of all presented creatures in this paper are available on the website http://www.irit.fr/∼Sylvain.Cussat-Blanc

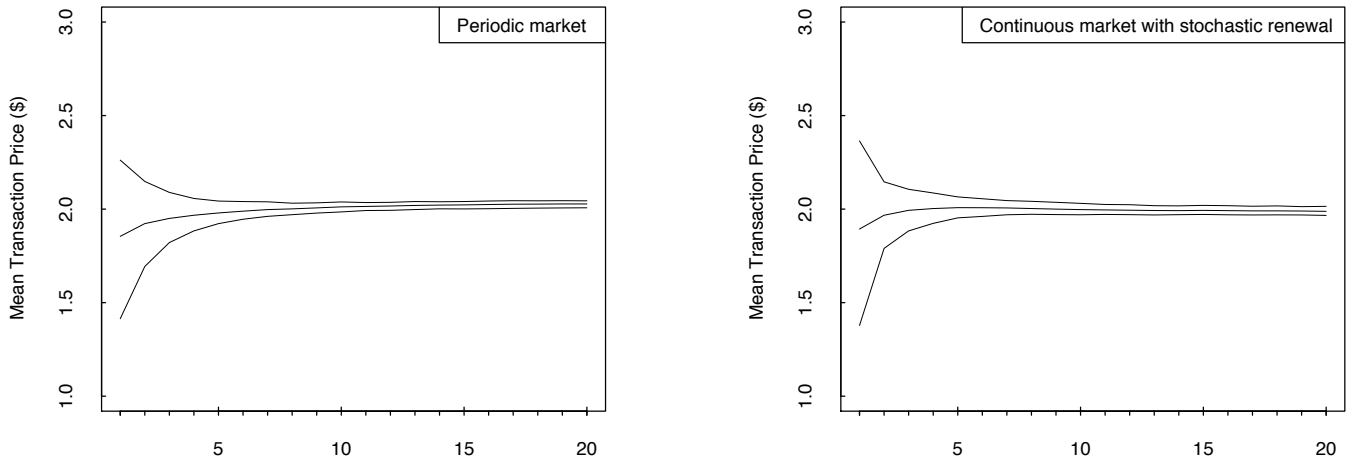


Figure 2: Mean transaction price over time (days or pseudo-days) for 500 ZIP experiments, for symmetrical supply and demand (P_0 = \$2.00) in a PR CDA (left), and a CR CDA with stochastic renewal (right). Dashed lines indicate the mean upper and lower transaction price boundaries at each day.

to occur in a range of prices around the equilibrium P_0 rather than convergence on the theoretical optimum price. Cliff and Preist (2001), in their continuous-time markets with human participants, found impressively low values of α that were below 0.1 within 600 seconds of the start of the experiment. Our overall data shows a failure to reach average α-values as low as 0.1, although we occasionally see single experimental runs with these low values.

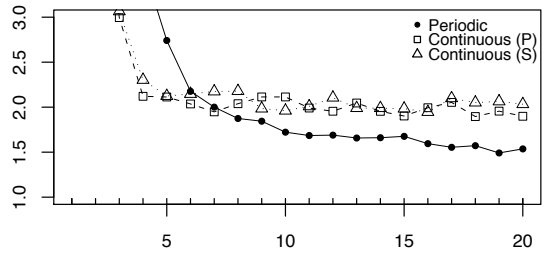


Figure 3: Mean α values over time (days or pseudo-days) for 500 ZIP market experiments: one periodic- and two continuous-replenishment CDA variants shown. Note that α values are very high (up to 20) in the earliest days of the experiments.

Hypothesis 3: CR markets exhibit greater experimental late-phase stability than PR markets. To investigate this question we split our data sets and looked only at results from the second half of each experimental run, i.e., days 11–20. This means we can look at market equilibration — effectively, long-term market efficiency — without the initial transients distorting the picture. We measure efficiency using both Smith's α and another measure, "profit dispersion". Gode and Sunder (1993) describe profit dispersion as the cross-sectional root mean squared difference between actual profits and equilibrium profits of an individual trader. For a group of n traders profit dispersion is given by $\sqrt{\frac{1}{n}\Sigma(a_i - p_i)^2}$ where a_i is the actual profit earned by trader i and p_i is p_0 for that trader. The more efficient the market, the lower the profit dispersion.

Figure 4 shows both mean α and mean profit dispersion for late-phase markets. Periodic-replenishment markets are consistently more efficient according to Smith's α, which means that transactions occur at prices closer to P_0 than in continous markets. There is also a very low variance in α-values for periodic markets. In terms of α performance the continuous markets with periodic renewal perform marginally better than the stochastic renewal version. In contrast, profit dispersion levels for all three market variants are approximately equal. This indicates that individual traders are not any more or less likely to trade at prices further from their personal equilibrium price in one type of market or another.

Hypothesis 4: Price formation in periodic markets is distributed around the opening of the market. We defined the "morning" period as being the first 25% of each trading day or pseudo-day, i.e., ticks 1–30. The trading volume during the morning period was approximately 3.5 times higher in periodic markets compared to continuous ones. This is not unexpected, as in the periodic CDA the entitlements of all traders are reset simultaneously as the market opens. This leads to an opportunity for many deals to be done immediately. More interestingly, despite the influx of entitlements to a morning market the transaction prices for periodic markets have a mean of 2.0147 ($\sigma = 0.037$). The transaction prices for both continuous markets in the morning period

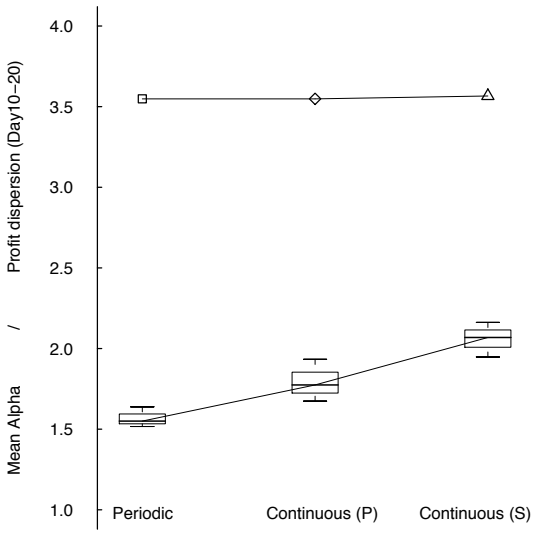


Figure 4: Mean convergence statistics for late-phase markets (days 11–20) with 500 ZIP experiments for each market model. Results for α indicate that the PR CDA is the most efficient, followed by the CR CDA with periodic renewal, and then the CR CDA with stochastic renewal. Average profit dispersion is roughly equal for all three types of replenishment.

are at a mean of 1.72 ($\sigma = 0.047$). Morning trade activity in periodic markets is very close to the equilibrium price despite the higher volume of trading. Approximately 79% of all experimental transaction occurs in the morning for a periodic market model, whereas in continuous markets the "morning" period has no particular significance and so obviously it accounts for 25% of trading.

Hypothesis 5: The optimal parameters for trading agents will take on different values depending on the market type. The behaviour of ZIP agents depends on a number of different parameters. Several different variables dictate the speed with which a ZIP trader modifies its price in the market, but the two most important are the Widrow-Hoff momentum (γ) and the agent learning rate (β). Preist (1999) demonstrates the significance of these variables. We looked at the effectiveness of different γ and β values in both periodic and continuous markets by creating surface plots of market efficiency, measured by Smith's α, for homogeneous communities of ZIP agents: see figure 5. We find that the resulting profiles of market efficiency are different for periodic and continuous CDAs. In other words, if I am a ZIP agent, the optimal settings for my core parameter values will depend on the market type I am in. In a periodic market, a value of $\beta = 0.2$ will produce the most efficient equilibrating market performance. Momentum γ, which acts to damp the oscillations for heuristic adjustments, can then vary across the range of 0.1–0.5 and this makes little difference to performance if $\beta = 0.2$ (see figure 5, left panel). In continuous-replenishment markets, the best example of market equilibration results from ZIP traders with $\beta = 0.1$. Continuous markets react more than periodic markets when γ is varied over the range 0.1–0.5. In a continuous market ZIP agents with lower momentum result in more efficient market behaviour (figure 5, right panel).

Intuitively we might expect that fast learning (a high value of β) and strong damping of adjustment oscillations (a high value of γ) would produce ZIP agents with more efficient market behaviour. Instead the trend for both markets is the opposite. Of course, we should be aware that ZIP parameters are not limited to γ and β. Our rationale for using these variables was not to find the most efficient ZIP trading strategy, but merely to illustrate that market replenishment style affects the way a ZIP trader should best operate.

It is also noteable from these results that some of our combinations of fixed γ and β ZIP variables produce markets that are almost 50% more efficient than those of the populations of ZIP agents used in the main set of experiments that featured the random assignment of parameters. This evidence is suggestive that there may be a market efficiency gain if all traders are uniform agents and consequently can be said to share the same idea of rational behaviour.

Discussion

Our experiments are, as far as we know, the first studies conducted with adaptive artificial trading agents operating in a simulation of a continuous-replenishment CDA. We have demonstrated the robustness of the CDA institution in fair price formation, by showing that groups of ZIP trading agents can consistently converge to the competitive equilibrium price and quantity governed by the supply and demand curves of the market. These results validate the observation of Cliff and Preist (2001) that both periodic and continuous markets can reach an equilibrium price. The use of simulation methods allows us to examine price formation variables more easily than in human-based experiments and we have therefore compared and contrasted the two CDA variants in more detail than was possible for Cliff & Preist in 1998.

Firstly, we found that profit dispersion between markets is almost identical in the later phase of the market for all three of our CDA variants. Secondly, we examined the α statistic over time, which calculates the divergence of market activity from the competitive equilibrium price. A comparison between the α values of periodic and continuous markets over time suggests that periodic markets equilibrate more efficiently over the long run than do continuous-replenishment markets. Comparing markets in late-phase allows measurements that are free from the effects of initial market turbulence, and thus facilitates a fair comparison between peri-

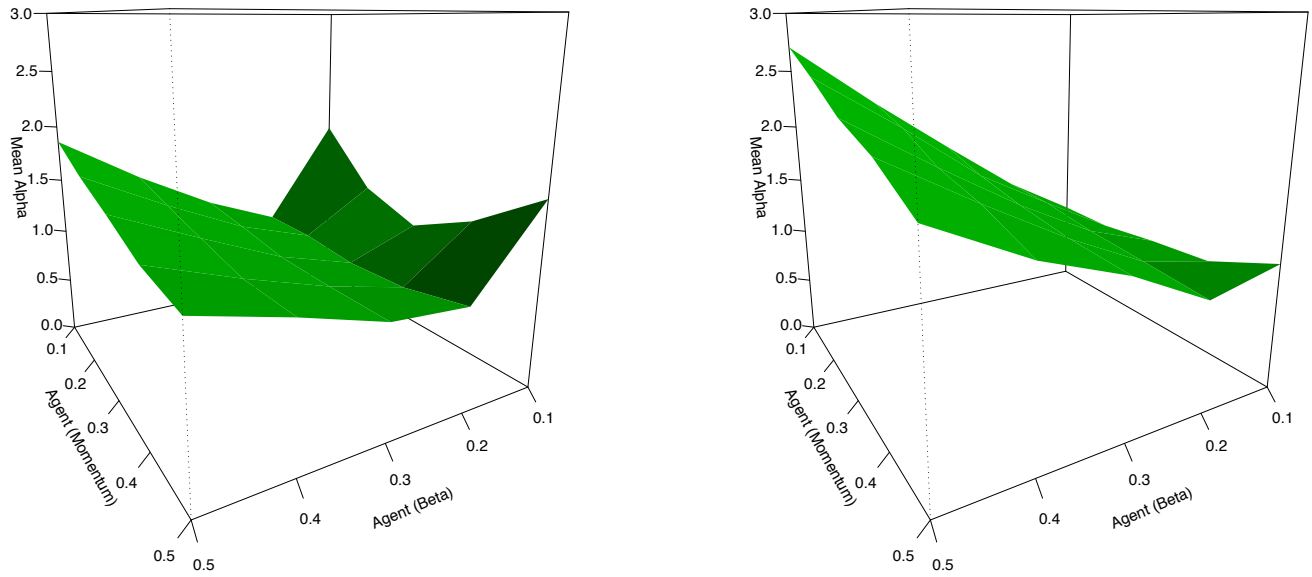


Figure 5: Surface plot of α against different values for γ (momentum) and β (learning rate) in a homogeneous ZIP agent community. The periodic market is shown on the left and the continuous market on the right. Results were generated over 500 experiments with all other agent parameters remaining at default values for ZIP version 1 (Cliff and Bruten, 1998). Note that the two surfaces are quite different, indicating that the two market types produce different optimal strategies.

odic and continuous markets. With no difference in profit dispersion across the three market types but with periodic markets achieving the most impressive (i.e., lowest) α values, this suggests that periodic markets represent a (near) Pareto-optimal solution to the problem of market design, with respect to our two measures of market efficiency.

Our original intuitions about the likely relationship between market efficiency and temporal structure were, in fact, the direct opposite of our results. We expected that the reoccuring event of an opening and closing of the market for our periodic variant would be enough to bring about a mini-fluctuation in the movement of opening prices each day and that possibly this pattern of trading would lead to oscillations around the equilibrium price at daily intervals. For CR markets, our intuition was that competitive price formation would occur early on and be maintained without such interruptions. Our original expectations can be summed up by the analogy that an engine that is continually restarted runs less smoothly than one that only starts once.

While it is not immediately clear why periodic markets over time deviate less from the competitive equilibrium price when compared to continuous markets, we can illustrate one reason for this behaviour from the perspective of the proportion of active agents within the market. The aggregate movement of price formation towards transactions at the equilibrium price only occurs if an agent is active within the market. For PR markets there is no potential delay in an agent being active for any given day, as by default all agents are deemed active at the start of each day. In a CR market, agents may in theory wait for a maximum time period equivalent to two days before being active within the market. An agent can only make meaningful contributions to the movement of the current trading price when it is active. Therefore in periodic markets, in which all agents start the day as active participants in the market, the collective action of all agents in reaching the equilibrium price will be maximally efficient. It may be that this activity being concentrated in time leads to the improved α values of the periodic market in comparison to the continuous ones.

Our average α-values for both the CR and PR variants of the market compare poorly to the reported α-values obtained by Cliff and Preist (2001) with human traders. This may well indicate relative inefficiency on the part of our ZIP agents, but it is also possible that the α-values reported by Cliff and Preist were the result of a regrettably small sample size.

The majority of PR market transactions occur within the "morning" period (i.e., the first 25% of the trading day), whereas in CR markets the trading activity is unsurprisingly spread across the trading day as the morning has no special significance. After the rush of morning trading, the remainder of the day in a periodic market is an empty trading environment, although quotes are still continuously made. In a sense, our PR markets "waste" most of the time of their participant traders, as (in these experiments) there isn't enough market surplus to fulfill the desired shouts; and so on average

our PR markets were nonuniformly — and arguably inefficiently — used over the duration of each day. In contrast, the CR market successfully facilitates continuous trading. Many of these dynamics may be attributable to the assumption that each trader makes only one trade per day. However, even if agents traded many units per day we believe that a concentration of trading volume in the morning would remain characteristic of periodic markets as opposed to continuous. Empirically testing this belief remains a topic for further work.

How does periodicity of replenishment affect the agent? Our results suggest that groups of agents with uniform trading heuristics perform differently in each market. Therefore, each market requires a different trading strategy to produce the greatest efficiency or to extract the greatest utility. From the agent perspective, these two styles of market replenishment create two different playing fields. Results show that each market is capable of reaching the equilibrium price with intelligent trading agents, but it is important to emphasize that the greatest market efficiency is achieved by different agent strategies in the different marketplaces.

Questions concerning which of PR or CR as a market model is more efficient and which model offers the fairest profit distribution are hard to clarify. Indeed, if these questions were easy to answer, we assume that all real-world CDA markets would have converged to the optimal market model. The distinction between market types exists because each possesses different practical features in their own right.

Further work

While the results presented in this report illustrate new work on the CR market model, there are still many ways in which our experiments could be extended. Firstly, we limited our ZIP agents to handling only a single trade per day. Cliff and Preist worked with traders with multiple entitlements per day, who were also able to buy or sell multiple units in one transaction. The rationale for allowing our ZIP traders multiple daily entitlements would be to look at whether more sophisticated trading takes place, based on accumulated entitlements being filled at a later time in a continuous market.

We have kept our models of agents and markets simple in the interests of clarity. However, there are numerous features of the trading agent behaviour that could be improved. ZIP agents are unable to formulate a decision process that considers waiting in the market and making full use of continuous time (i.e., they cannot make a decision as to whether waiting is better than buying now). The ZIP agents used here are the original 1997-vintage "Version 1.0", now referred to as ZIP08 (Cliff, 2008). One consideration would to implement an optimising ZIP60 agent (Cliff, 2008) based on a genetic algorithm, to properly observe how different the optimised variables would be in each market. This would be a full extension of Hypothesis 5. Additionally a ZIP agent could also be made sensitive to CR markets by receiving more informative signals on how long the market has been running, and through greater temporal awareness being able to exploit strategies such as delaying the sale of a commodity in order to exploit a shortage and higher prices later on.

We could also be more rigorous in creating a framework that is completely free from synchronous behaviour. This is obviously desirable because of the asynchronous nature of real markets. The rate at which our agents update their price information is synchronised in our models, at each tick. It is possible that experimenting with an asynchronous and varied update rate for each agent could capture the asynchronous intelligence of real-world populations of traders.

References

Cliff, D. (1997). Minimal-intelligence agents for bargaining behaviors in market-based environments. Tech. Report HPL-97-91, Hewlett Packard Labs, Bristol, UK.

Cliff, D. (2008). ZIP60: Further explorations in the evolutionary design of trader agents and online auction-market mechanisms. *IEEE Transactions on Evolutionary Computing*. In press, also available from http://eprints.ecs.soton.ac.uk/14078/.

Cliff, D. and Bruten, J. (1998). Market trading interactions as collective adaptive behavior. In Pfeifer, R., Blumberg, B., Meyer, J.-A., and Wilson, S. W., editors, *From Animals to Animats 5: Proceedings of the Fifth International Conference on Simulation of Adaptive Behavior*, pages 417–426. MIT Press, Cambridge, MA.

Cliff, D. and Preist, C. (2001). Days without end: On the stability of experimental single-period continuous double auction markets. Tech. Report HPL-2001-325, Hewlett Packard Labs, Bristol, UK.

Gode, D. and Sunder, S. (1993). Allocative efficiency of markets with zero-intelligence traders: Market as a partial substitute for individual rationality. *Journal of Political Economy*, 101(1):119.

Preist, C. (1999). Commodity trading using an agent-based iterated double auction. In *Proceedings of the 3rd Annual Conference on Autonomous Agents (AA'99)*, pages 131–138.

Smith, V. (1962). An experimental study of competitive market behavior. *Journal of Political Economy*, 70(2):111–137.

Smith, V. and Williams, A. (1983). An experimental comparison of alternative rules for competitive market exchange. In *Auction, Bidding and Contracting: Uses and Theory*, pages 307–334. New York University Press.

Tesfatsion, L. (2002). Agent-based computational economics: Growing economies from the bottom up. *Artificial Life*, 8(1):55–82.

Measuring the robustness of a developmental system based on sequential growth rules

Michael Cohen[1], Mark Miodownik[2] and Buzz Baum[1]

[1]MRC-LMCB, University College London, UK
[2]Material Research Group, King's College London, UK
m.cohen@ucl.ac.uk

Abstract

Understanding how complex structures emerge from localised interactions in a robust way is essential to unraveling the mechanisms that underlie developmental processes in both biological and artificial systems. This study investigates the effects of genome complexity on robustness using a simple, evolved developmental system in which cellular automata (CA) rules are applied in sequence in order to generate a 1D pattern of cells. The system employs a 1D two state CA with 128 distinct nearest neighbour update rules. Each developmental run is initiated with a single cell. The cell update rules adopted by every cell at each time-step are allowed to change sequentially at different times according to the instructions contained in a 'genome'. In order to generate a set of productive developmental programs for this analysis, a genetic algorithm was used to select for individuals whose cell states, after a fixed number of time steps, match a set of pre-defined target patterns. This was repeated for genomes of different sizes. The robustness of evolved and randomized CA patterns were compared by systematically applying single cell state perturbations during pattern development. This analysis revealed that in these evolved systems genome size has a positive effect on robustness by freeing the system to generate patterns using a relatively unbiased set of rules, which have very different individual properties. In contrast, smaller genomes are frequently forced to rely on complex patterning rules to generate complex patterns, which amplify damage and hence reduce their robustness. In addition, pattern size (the number of cells) was found to be a major factor in the measured robustness in this system. This is because the cumulative damage induced by developmental perturbations does not scale with pattern size. As a result, increasing pattern size reduces the percentage damage following perturbations and improves overall robustness. In conclusion, we have shown that pattern robustness is an additive effect of the ability of individual rules to propagate and heal defects resulting from environmental perturbation in this simple CA system, and is potentially increased by increasing pattern size and genome size. These results have implications for our understanding of robustness in biological and artificial systems.

Introduction

Both natural and artificial developmental systems are known to generate physical forms that are self-regulating and as such are highly robust to perturbations of many kinds including artificial wounding or cell removal (Wolpert, 2002; Kumar and Bentley, 2003). Understanding how complex structures emerge from localised interactions in a robust way is essential to unraveling the mechanisms that underly developmental processes. In biological and artificial developmental systems, development is often governed by cellular interactions. Fundamentally different processes may occur in sequence at different developmental stages in order to generate overall pattern and form (Wolpert, 2002). This paper explores how using different numbers of rules in sequence during the development of a cellular automata (CA) pattern affects the overall robustness of the patterning process.

Robustness to cell perturbation (or 'wounding') and self-regulation of developed patterns or 3D forms has previously been observed as an emergent property of evolved developmental CA systems (Andersen et al., 2006; Miller, 2004; Basanta et al., ress; Devert et al., 2007; Federici and Downing, 2006; Grajdeanu and Kumar, 2006; Streichert et al., 2003). However, because these systems are complex it is unclear precisely how the implicit developmental rules in these systems lead to a robust developmental program. In order to create a simple system in which the processes underlying the evolution of developmental robustness could be simply and rapidly analysed in detail, we developed a 1D model, in which CA rules are applied in series. CA rules are known to produce characteristic patterns relating to their dynamical properties and overall system stability (Wolfram, 2002) but it is not immediately apparent how such properties may contribute to the effect of cell perturbations during their development. In particular, no previous study has established how using CA rules in temporal sequence may effect the overall system stability and hence the robustness of patterning. Using this system, we explored the roles of evolution and genome complexity on developmental robustness. In each case, we compared results from evolved genomes with those obtained using an equivalent set of random genomes.

Method

The experiment uses a 1D two-state CA of the type defined by Wolfram (Wolfram, 2002). This system consists of a line of *cells* in one of two states; black or white. (The lines are

effectively infinite to avoid edge effects.) In this paper a black square is referred to as a cell a white square represents an empty space. At each time-step in the running of the CA, each location is updated according to a set of conditions dependent only on its previous state and the state of its two adjacent neighbours. The complete set of conditions defines an update rule, which operates on all cells in the system at any one time step. Here, a sub-set of 128 rules are used which exclude those rules whereby a cell can emerge from an empty neighborhood of cells. These are labelled according to Wolfram's numbering scheme and comprise the even numbers between 0 and 255.

The CA are developed for 51 time-steps at which time the 1D pattern generated is referred to as the 'end-state pattern'. In this system the rules are allowed to vary over different time periods, as shown in figure 1, where in this case 6 distinct rules are implemented in series. The particular rule applied to every cell at each time-step is contained in the 'genome for each individual run of a CA. The whole population in any single experiment has the same genome length, or number of genes, n. The specific case illustrated by figure 1 is represented by the n=11 genome: 10 50 174 242 230 122, 9 15 24 32 45. Here the first six numbers represent the set of rules (R1-R6). The remaining five numbers represent the transitions times (T1-T5) at which the rules change. The transition times are constrained to occur in evenly distributed fractions of the total 51 time-steps. For example. in the n=11 case shown, the 5 transition times occur in bins of 10 time-steps. Where the CA patterns are directed by artificial evolution, the fitness function (defined subsequently) is applied at time-step 51, where the end-state pattern of cells is compared to a pre-defined target pattern (shown in grey).

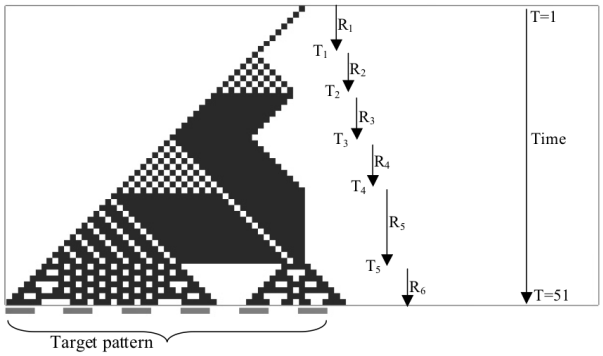


Figure 1: A screen shot of an individual CA run. The end-state pattern at time-step 51 is developed according to the cell update rules. Six rules (R1 to R6) are applied to the system over six different time periods; the transition points of which are labelled T1-T5. The light grey pattern below the box shows the target pattern, P1, towards which the system may be evolved.

Evolving patterns

To test the behaviour of the system under specific types of directed patterning the CA were evolved using a Genetic Algorithm (GA) (Davis, 1991; Mitchell, 1998). This was applied as follows. A population of size N=500 individual genomes was created and these were each developed in accordance with the CA program. Genes were initially seeded by a random number generator. The rule defining genes were selected randomly from the complete set and the time values were randomized within the time period constraint as previously described. A fitness function scored each individual according to the similarity of their end-state pattern, at time-step 51, with a pre-defined target pattern. The target patterns used are shown in figure 2. These were selected to test the effects of varying pattern regularity, symmetry and breadth of distribution. The first six patterns, P1-P6, are the same size, 30 cells, to enable direct comparison, whilst patterns P7 and P8 are 60 cells in size to control for the effects of pattern size.

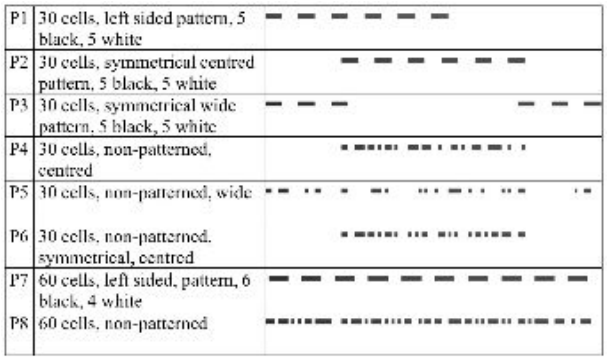


Figure 2: Target patterns selected to test for pattern regularity, symmetry, distribution and size.

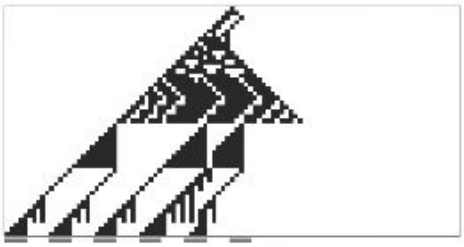
The fitness function sums the number of cells that differ in their location between the target pattern and end-state pattern of the developed CA. This is equivalent to the 'Hamming distance' between the two bitwise pattern encodings (Hamming, 1950). Thus the most 'fit' individuals have the lowest 'fitness score' and a perfect correlation scores zero. Tournament selection was used to determine which individuals pass to the next generation; whereby, two individuals are randomly chosen and the fitter individual selected. Crossover was not found to benefit the GA and was not used. The genomes of the next generation were mutated by randomly selecting either new CA rules from the complete set or transition times from within the constraints previously described. The mutation rate, per genome, used at each genome size, n=3,11 and 23, were; 0.6, 0.8, and 1.0 respectively.

The process of selection and mutation leads to a new generation after which the whole process is repeated. Throughout the experiment a fixed population of N=500 was used

and the system was evolved for 1000 generations for target patterns P1 to P8 as well as for an extended 5000 generations for pattern P1 (this set of data is referred to in the results as P1+). Ten evolutionary runs were carried out for every genome size and target pattern. These parameters were all optimised prior to the experiment and were found to be sufficient to achieve stable average fitness scores of low variance. Genomes of sizes 3,11 and 23 were used to compare the effects of genome complexity in this system.

Robustness Testing

Evolved solutions and unevolved, randomly generated genomes were tested for their robustness to cell perturbations. Each single (black) cell was systematically perturbed (cell state changed to white), one at a time, during the pattern development. The emergent end-state pattern after each cell perturbation was compared with that of the unperturbed CA (see figure 3). The damage caused by each cell perturbation was measured in terms of the Hamming distance between the perturbed pattern and the original end-state pattern. This difference was then expressed as a percentage of the original pattern size (the total number of black cells in the end-state pattern). The overall developmental robustness of a particular individual was regarded as being inversely proportional to the averaged percentage damage caused by all developmental cell perturbations. Mean data from 750 randomized genomes of each genome size was compared with mean data from the 10 evolutionary runs at each target pattern.



(a) Original n=23 solution consisting of 25 (black) cells.



(b) A single cell perturbation (from black to white) causes a shift in the end-state pattern such that 10 (black) cells are in a different location. Equivalent to a damage score of 40% of the final pattern size.

Figure 3: Measuring the effects of cell perturbations.

Results

A set of CAs with genomes of different sizes (n=3, 11 and 23) were evolved under a genetic algorithm by selecting for their ability to match a set of 8 pre-defined target patterns. The genomes contained instructions for the transient update of the CA rules. In this section the evolved solutions are investigated with regard to their relative success in matching target patterns, the developmental methods adopted to try meet those target patterns and their robustness to developmental cell perturbations.

Pattern Characteristics

Examples of the evolved solutions are shown in figure 4. There were variations in the 10 solutions obtained at each evolutionary run and the subset shown here are intended to illustrate some of the generic differences between the target pattern types and genome sizes. Most immediately striking is the difference in the developmental profiles (that is all the cells at each time-step leading up to the end-state pattern) among the different genome sizes. The n=3 solutions have very distinct profiles characterised by the two different rules applied to meet the target pattern. In contrast the n=23 developmental profiles share a common feature of branching or segmentation at the transition between the 12 rules comprising their genome. There is a complexity of patterning that arises as a result of these rule transitions. The n=11 solutions reflect an intermediate case. It is immediately apparent that the n=11 and n=23 genomes are good at matching the more regularly spaced target patterns but bad at matching a highly distributed random target pattern such as at P5. For the larger patterns, P7 and P8, all individuals of the 3 genome sizes rely on rules that cause an expansion or growth in the number of cells present, as might be expected.

Whilst the target patterns P1-P6 all consisted of 30 cells, the evolved end-state patterns varied in size between 8 and 35 cells. Among randomly generated genomes there was also a significant variation in pattern size. In data obtained from 750 random genomes of each genome size, the average end-state pattern size for n=3, 11 and 23 was 14, 7 and 3 cells respectively. Although the average size was seemingly, relatively low, significantly larger patterns of over 60 cells were also generated by the random samples. The size of the end-state patterns was found to have a significant effect on the robustness of the CA, as is shown later in these results.

Fitness of Evolved Solutions

The GA is designed to identify solutions that match the target pattern. This was shown to be the case, since for all genome sizes the GA yielded patterns with an improved fitness score. The average scores obtained for each of the genome sizes are shown in figure 5. Average scores are given for the champion individuals at the first and last generations. Overall the larger genomes show slightly less fit

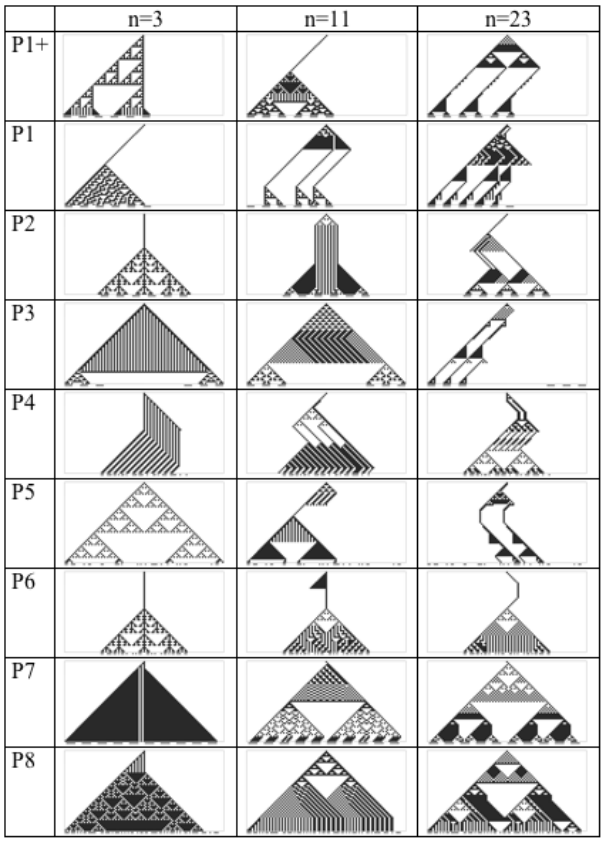


Figure 4: Examples of evolved champion solutions obtained at the last generation of evolutionary runs carried out for each genome size at each target pattern. The pattern at each time step is shown with developmental time represented in the vertical axis.

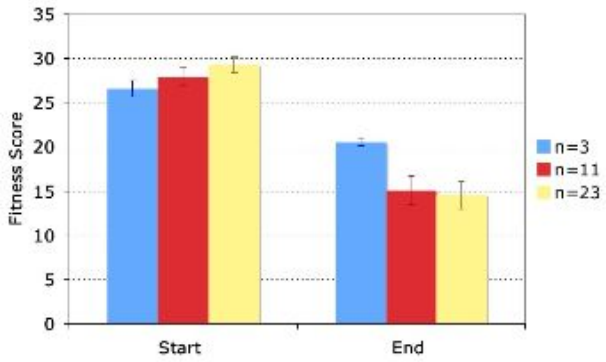


Figure 5: The average champion scores attained by each genome size. The data compares the lowest fitness scores from the first generation (labelled 'Start') and the last generation (labelled 'End'), averaged over all target patterns for all 10 evolutionary runs. Error bars show the 95 percent confidence intervals for the mean values.

(higher) scores at the start of the evolutionary runs, thus indicating that a random population is less likely to match the target patterns. After evolution the n=11 and n=23 genomes achieve very similar average scores both significantly fitter than for the n=3 case.

There were identifiable differences between the target patterns. The n=11 and n=23 genomes consistently outperformed the n=3 genome except in the case of one target pattern, P6. In general for the two larger genome sizes the regularly spaced target patterns P1, P2 and P7 achieve the fittest relative scores. Where more complex arrangements of cells were encountered these systems did less well in matching the end-state patterns.

In order to further qualify the relative evolvability at each genome size the fitness scores obtained by evolution were compared with those of a randomly generated population of 500,000. This is the equivalent number of individuals that are searched by the GA evolving a fixed population of 500 individuals over 1000 generations. The n=3 evolved solutions never out-performed the random search solutions. In contrast, for the n=11 and n=23 genomes all of the evolved solutions outperformed the random search.

Robustness to developmental cell perturbation

To analyse the effects of genome size on developmental robustness in this system cell perturbations were made to both evolved and unevolved individuals (see method for details). Figure 6 shows a plot of this data. Here, the average percentage damage score has been plotted against the size of the end-state patterns. For the random genomes each individual data point is plotted together with a trend line indicating the population mean and associated confidence intervals for this value. For the evolved solutions, data points are plotted showing the mean value obtained over the 10 evolutionary runs with associated confidence intervals.

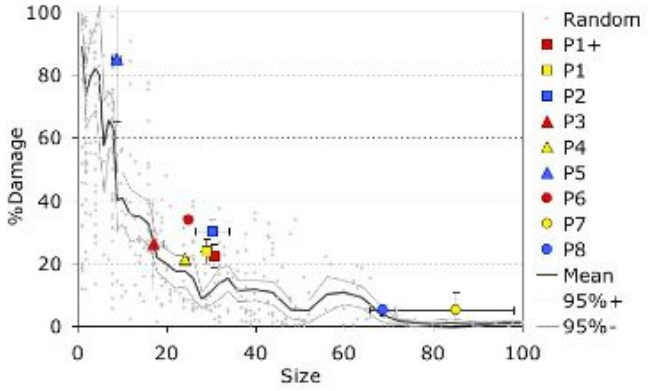
The evolved solutions for the n=23 genome all sit on the same trend line as for the random genomes and the range of random data in this case is much more constrained than for the n=3 and n=11 genomes. In contrast, for the n=3 and n=11 genomes the distribution of the random data is larger than for n=23. For some target patterns the mean robustness of the evolved solutions patterns is different to the mean random data of equivalent size. The n=3 evolved solutions for target patterns P1, P2, P3, P5 and P6 all have a mean robustness that is significantly lower than for the random data (evolved individuals show higher percentage damage scores within an equivalent pattern size range). For n=11, the solutions at target patterns P1 and P2 are significantly less robust than the average data. This would suggest that evolution towards these specific target patterns has repeatedly selected for combinations of rules and transition times that are less robust than the average random sample. Part of this loss of robustness may be attributed to the fact that these individuals sometimes show a sustained period without pattern

growth that is inherently weak to any perturbation; as can be observed in the examples shown in figure 4 where a single cell is maintained over a number of time-steps before any larger pattern finally develops. A perturbation during this early period without growth will remove the entire pattern. In contrast, the n=23 solutions consistently employ periods of growth and patterning throughout the pattern development. Another factor underlying the loss of robustness of some of the evolved solutions may be a selection for individual rules that are inherently sensitive to perturbations. This will be analysed further in the discussion section.

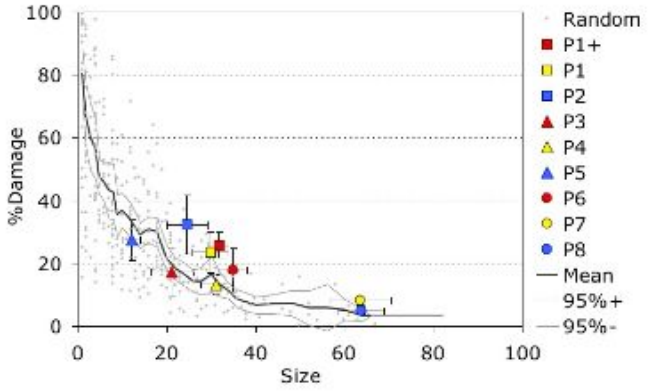
For all three genome sizes the predominate factor determining robustness is the end-state pattern size itself. To further investigate the effects of end-state pattern size as well as genome size, the mean trend lines from the randomized data are plotted together in figure 7. The curves from each genome size all follow the same trend and there is no significant variation in robustness. Thus it can be concluded that the use of a greater number of rules does not translate into a change in robustness in this system.

The real data are contrasted with curves that represent the effects of altering the state of 2, 4 and 6 cells in the end-state pattern; that is, a theoretical plot in which for each cell perturbation the end-state pattern is altered by a fixed amount. The curves obtained from the randomized genomes all follow a trend very similar to that of a fixed 4 cell perturbation. Only for very low pattern sizes, below approximately 10 cells, do the curves align more closely with a fixed absolute damage of 2 cells. This would suggest that regardless of the size of the pattern generated (and thus the average rate of growth of black cells) the average, absolute damage caused by cell state perturbations remains fairly constant over a wide number of randomized genomes. It is important to note that this is an average quantity. The effect of a cell perturbation early in development, where there are fewer cells, causes significantly more absolute damage than one very late in development (where there are likely to be many more cells). What is suggested here is that, averaged over developmental time, the absolute damage caused by a perturbation is largely independent of the ultimate pattern size. The effects of a cell perturbation do not scale in accordance with the rate of pattern growth and end-state pattern size, as might be expected. Hence, the percentage damage caused by a single perturbation rapidly decreases with increasing pattern size as the curves here demonstrate.

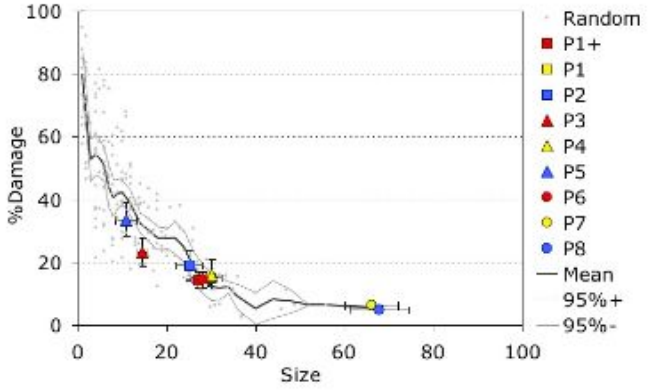
The results have shown that for all three genome sizes, there is very similar trend between the average robustness of randomly generated CA and their end-state pattern size. For evolved CA, the average robustness is shown to differ among solutions obtained at the different target patterns. Whilst the variation in evolved robustness can principally be explained by differences in evolved pattern size, some evolved solutions show a lower than average robustness than was obtained for CA derived from random genomes of equiva-



(a) n=3



(b) n=11



(c) n=23

Figure 6: A plot of end-state pattern size against cell perturbation damage expressed as a percentage of size. Data was obtained from 750 randomly generated genomes of each genome size. The trend line shown the mean of this data with associated 95 percent confidence intervals (derived from data bins across ranges of sizes). For the evolved solutions, mean values and confidence intervals derived over 10 evolutionary runs are shown.

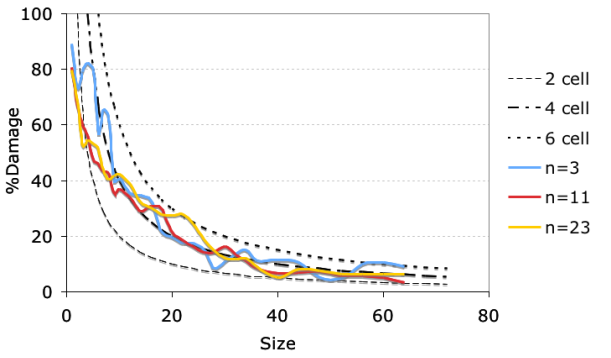


Figure 7: The mean trends of end-state pattern size against the average cell perturbation damage expressed as a percentage of the original pattern size. The data was derived from 750 randomly generated genomes of each genome size. This is contrasted with model curves representing a fixed absolute damage, at all pattern sizes, of 2, 4 and 6 cells.

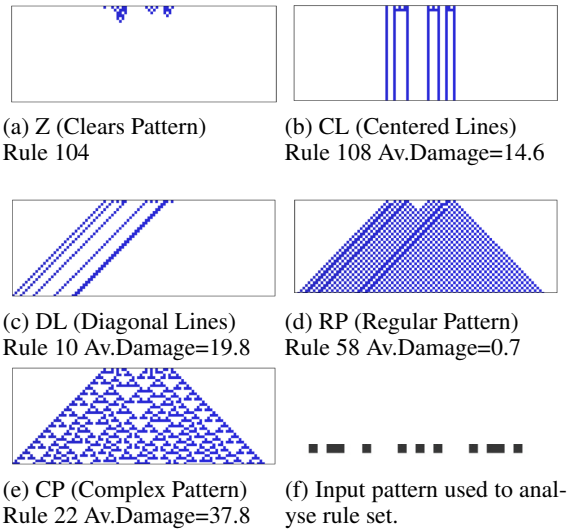


Figure 8: Rules classified according the defined criteria. Shown here are examples of each rule 'type'. The rule number is quoted along with average percentage damage score for that particular rule when each cell was systematically perturbed.

lent pattern size. Therefore, it can be inferred that in order to match targets the evolutionary algorithm is repeatedly selecting for particular combinations of rules that degrade overall robustness in these particular cases.

In order to better understand the effects of individual CA rules on robustness in this system, the rules were categorized and analysed in isolation. Figure 8 demonstrates how the individual rules were categorized. The figures show the behaviour of each rule after input from an arbitrary pattern comprising 11 cells in 9 discrete blocks at time-step one. Each was run for only 40 time-steps to account for the additional 'width' of the input pattern. This 'input' pattern was selected to illustrate the behaviour of the rules at some time into the development of a pattern, as distinct from seeding by a single cell.

A measure of the end-state pattern size and the average percentage damage caused by cell perturbations was made for each individual pattern. These are plotted in figure 9. This shows how the regular patterning (RP) rules are significantly more robust to cell perturbation than the complex patterning (CP) rules, regardless of the pattern size. The emergence of a regular pattern of growth from the irregular input pattern indicates that the system has a stable attractor state that is largely insensitive to initial conditions. Thus perturbing the system later in development has a similarly low effect on the emergent pattern. There is a self organization inherent in these types of rules. For the complex patterns the system is more sensitive to the initial conditions and forms complex pathways in the development of the pattern, with subsequent interactions when pathways intertwine; this results in the nested triangles characteristic of a complex pattern developmental profile. In this case information about previous cell states is transmitted throughout the CA in such a way that cell perturbations have an escalating effect on the emergent patterns at subsequent time-steps. The DL and CL rules that produce substantially less pattern growth show a perturbation response that scales very sharply with pattern size.

The mean trend line gives an indication of the average damage at each size for all the individual rules. It is interesting to note that when contrasted with the curves shown in figure 7 the mean trend among the individual rules closely follows the mean trend for the randomized genomes. This suggests that the average robustness of each of the combinatorial rule systems is essentially the same as the average robustness of the individual rules themselves. This reinforces the finding that the genome size has no intrinsic effect on the average robustness. In addition, it appears that the approximation towards a constant absolute damage (of approximately 4 cells), that was noted previously, can be attributed to a combinatorial effect of the different types of rules. Individually the different types of rules have quite distinct relationships in regards to pattern size and robustness. However the trend line shows that their aggregated relationship closely mimics that of a system with a fixed average response to perturbations in regard to pattern size.

It should be noted that the classification scheme adopted here is not concrete and there are a few rules that generate pattern that appear to be on the border between these types of classification. There is a correspondence with Wolfram's classification system for this type of CA, such that CL and DL are Class 2, RP Class 1 and 2, and CP Class 3. Rules that fall between between RP and CP are Class 4 systems (Wol-

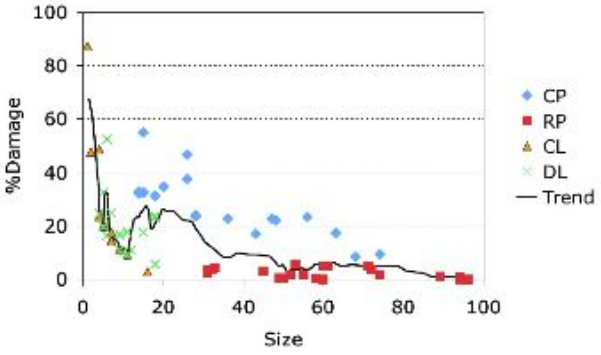


Figure 9: The robustness of individual CA rules of each classification type. The average percentage damage caused by cell perturbations is plotted against the end-state pattern size at time-step 40. The trend line shows a rolling mean average of all the data.

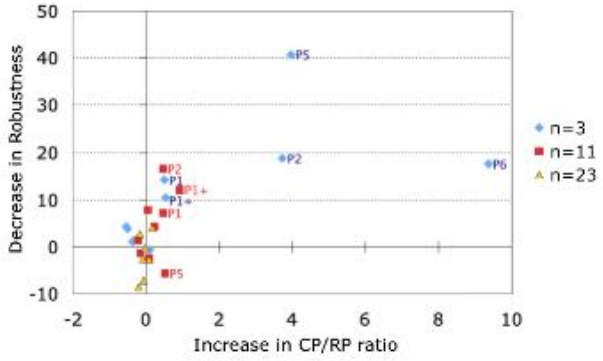


Figure 10: The effect of complex patterning rules on the robustness of evolved solutions. The x-axis shows the average difference in the evolved CP/RP ratio with the rule-set ratio. The y-axis shows the average difference between the evolved robustness scores (expressed as an average percentage damage due to cell perturbation) and the mean robustness of random data of equivalent pattern size (from figure 6). Data points located in the upper right quadrant reveal a correlation between complex patterning and a loss of robustness.

fram, 2002). This system of classification has convergence with other definitions relating to the dynamical properties of CA (Wuensche and Lesser, 1992). The principle distinction made here is that the RP rules are more dynamically stable than the CP rules.

To further investigate why the evolved solutions showed differences in their robustness in comparison to the randomized data, an analysis was carried out with regard to the proportion of rules adopted by the evolved genomes. For each of the evolved solutions the ratio of CP rules to RP rules was determined. The increase in this ratio, as compared with the actual rule set was then calculated. This value is plotted in figure 10 against the increase in the average perturbation damage score obtained by evolved solutions as compared to the mean randomized data of equivalent size (as illustrated in figure 6).

This analysis reveals that where CP rules have been used in high proportion, there is, in most cases, an equivalent decrease in robustness (increase in the percentage damage caused by cell perturbations). Therefore, it seems that in general a loss of robustness can be explained by an increased uptake in CP rules, which are required in order to match certain target patterns and are thus selected for by the GA. This generalization is true in all but one example, where for the n=11 genome at target pattern P5 the evolved solutions are seemingly more robust than average whereas the CP/RP ratio is higher than for the rule set itself. This may be attributable to the very small end-state pattern size that was adopted by these solutions, making them more robust than equivalently sized random patterns, even though a significant amount of their development was undertaken by complex growth rules.

Discussion

In summary, this analysis has demonstrated that there is no intrinsic emergent robustness as a result of increasing the number of sequential 'rules' in a CA system but there is a potential loss of robustness associated with evolved rule biases in smaller genomes. On average the two larger genomes were shown to evolve better (more fit) solutions than the smaller genome. The evolvability of the larger genome sizes is related to the size of the parameter space that they may select from. The greater complexity of the genome provides the means for complex adjustments in the patterning of cells that is not present in the individual rules themselves. Thus the n=3 genomes and to some extent the n=11 genome were more reliant on the use of specific rules for the generation of particular patterns and it was shown that when complex rules were used their robustness was degraded. Though the n=23 solutions were not inherently more robust to cell perturbations than the n=3 or n=11 genomes, they did not deviate from a random distribution in their selection of rules and so showed higher levels of robustness when the smaller genomes were forced to do so in order to achieve the required patterning.

Robustness, here, was explicitly defined as a percentage change in the phenotypic patterns as this was considered to provided the most informative comparison between different evolved solutions. It was shown that, on average, the propagation of a perturbation through development only effects a limited number of cells in the end-state pattern. This means that it is predominantly the size of a developed pat-

tern that contributes to its overall robustness such that the percentage impact of cell perturbations is reduced as size increases. In biological systems there may be a corresponding relationship between organism size and robustness such that larger organisms, containing a greater number of cells, may show less phenotypic response to both developmental and genetic perturbation. Research into the evolutionary adaption of size highlights the physiological or environmental constraints acting on an organism (LaBarbera, 1989). It may be that there is an underlying selective pressure to increase organism size for overall robustness.

There was no evidence for emergent robustness as a product of the GA itself. Adding stochasticity or noise to the CA development, by introducing cell death, may cause the system to evolve more robust solutions. In this scenario it may be that 'fit' solutions that can withstand developmental noise are more likely to be repeatedly selected for during evolution.

The particular CA used here update rules that operate at every site with relatively complex asymmetrical configurations. Rather than cell growth, it more closely represents a collection of established cells making internal decisions about their differentiation between states. Future work may explore how a similar system may be reconfigured to better represent more realistic cellular growth rules.

Conclusion

This study has provided a measure for the developmental robustness of evolved CA patterns in a simple one dimensional system. It has shown that there is no robustness intrinsically associated with using additional rules. However, increasing the complexity of a genome has a beneficial effect on robustness simply because it frees the system to generate patterns using a relatively unbiased set of rules.

For randomized genomes of each genome size individual cell perturbations, on average, produced approximately the same amount of absolute damage to the emergent patterns. This was shown to be equivalent to approximately 4 cells. Robustness, here, was explicitly defined as a percentage change in the phenotypic patterns. Hence, there was a strong correlation between robustness and pattern size.

It was revealed that the robustness of randomized genomes could be attributed to the aggregate effect of selecting from the complete rule set. Different types of rules demonstrated very distinct relationships between robustness and pattern size. However, a trend line showing the mean robustness of each of the individual rules approximated the trend one would expect given a fixed amount of absolute damage, regardless of overall pattern size. Thus the average robustness of the randomized genomes could be simply interpreted as a reflection of the average robustness of the individual rules.

In the analysis of individual rules it was shown that rules generating complex patterns were sensitive to cell perturbations. By contrast, the regular patterns in this system act as a stable attractor in which the state of cells enter a single homogenous state or a predictable cycle that is insensitive to changes in their input. Where genomes of the evolved CA patterns adopted a large proportion of 'complex' patterning rules, their robustness is shown to be reduced in comparison to the average robustness of patterns of equivalent size.

Acknowledgements

This work was supported by CoMPLEX, UCL.

References

Andersen, T., Newman, R., and Otter, T. (Fall 2006). Development of virtual embryos with emergent self-repair. In *American Association for Artificial Intelligence - Symposium on developmental systems*, pages 16–23.

Basanta, D., Miodownik, M. A., and Buzz, B. (in press). The evolution of robust homeostasis and the stem cell niche in artificial multicellular organisms. *PLoS Computational Biology*.

Davis, L., editor (1991). *Handbook of Genetic Algorithms*. Van Nostrand Reinhold.

Devert, A., Bredeche, N., and Schoenauer, M. (2007). Robust multi-cellular developmental design. In *Genetic and Evolutionary Computation – GECCO-2007, Part I*, pages 982–989.

Federici, D. and Downing, K. (2006). Evolution and development of a multicellular organism: scalability, resilience, and neutral complexification. *Artif Life*, 12(3):381–409.

Grajdeanu, A. and Kumar, S. (2006). A novel developmental system for the study of evolutionary design. In *Developmental Systems; AAAI Fall Symposium*.

Hamming, Richard, W. (1950). Error detecting and error correcting codes. *Bell System Technical Journal*, 26(2):147–160.

Kumar, S. and Bentley, P. J., editors (2003). *On Growth, Form and Computers*. Elsevier Academic Press.

LaBarbera, M. (1989). Analysing body size as a factor in ecology and evolution. *Annual review of ecology and systematics*, 20(97-117).

Miller, Julian, F. (2004). Evolving a self-repairing, self-regulating, french flag organism. In et. al., K. D., editor, *Genetic and Evolutionary Computation – GECCO-2004, Part I*, volume 3102, pages 129–139.

Mitchell, M. (1998). *An Introduction to Genetic Algorithms*. Masachusetts Institute of Technology.

Streichert, F., Speith, C., Ulmer, H., and Zell, A. (2003). Evolving the ability of limited growth and self-repair for artificial embryos. In Banzhaf, W., editor, *ECAL*, volume 2801, pages 289–298.

Wolfram, S. (2002). *A New Kind of Science*. Wolfram Media Inc.

Wolpert, L. (2002). *Principles of Development (2nd Ed.)*. Oxford University Press.

Wuensche, A. and Lesser, M. (1992). *The global dynamics of cellular automata*. Addison Wesley publishing company.

"Soft" Continuum Robots: the Interaction of Continuous and Discrete Elements

Lara S. Cowan and Ian D. Walker
Department of Electrical and Computer Engineering
Clemson University, Clemson, SC 29634, USA

Abstract

In this paper, we examine key issues underlying the design and operation of "soft" robots featuring continuous body ("continuum") elements. We contrast continuum and continuum-like robots created to date with their counterparts in the natural world. It is observed that natural continuum locomotors or manipulators almost invariably rely on hard/discrete elements (in their structure and/or operation) in their interactions with their environment. Implications for the successful operation and deployment of continuum robots are identified and discussed.

1. Introduction

There are innumerable alternatives available to the robot designer. However, only a small subset of these alternatives has been realized in hardware to date. Most modern industrial robots are (human) arm-inspired mechanisms with serially arranged discrete rigid links. This is fine for industrial work where the workspace is predefined and structured. However, robots are currently generally confined to such engineered and carefully controlled environments, and kept well away from humans and their world.

A robot that must interact with the natural world needs to be able to solve the same problems that animals do. Animals come in many shapes and sizes with widely varying specialized limbs suited to their particular everyday tasks. However, most robots are built according to "general-purpose" specifications with little attention to what they will ultimately be used for. The rigid structures of traditional robots limit their ability to maneuver in tight spaces and congested environments, and to adapt to variations in their environmental contact conditions.

In response to the desire to improve the adaptability and versatility of robots, there has recently been interest and research in "soft" robots [1]. In particular, several research groups are investigating robots based on continuous body "continuum" structures. Motivation for this work often comes from nature. If the body of a robot was soft and/or continuously bendable then it might emulate a snake or an eel with an undulating locomotion [2]. A slithering robot could navigate through a variety of terrains.

An alternative solution would be to have a continuous manipulator. A robotic continuum manipulator could be similar to a prehensile tail, an elephant's trunk, or an octopus's arm.

Several different types of continuum-like robots have been proposed. Robotic snakes have been built by a few different groups [3],[4],[5],[6]. These have almost all been built using multiple discrete links. These hyper-redundant robots can move in most of the ways snakes can, but they are not as conformable. Hyper-redundant robots, like the SnakeBot [7], represent a bridge between discrete links and continuous elements [8].

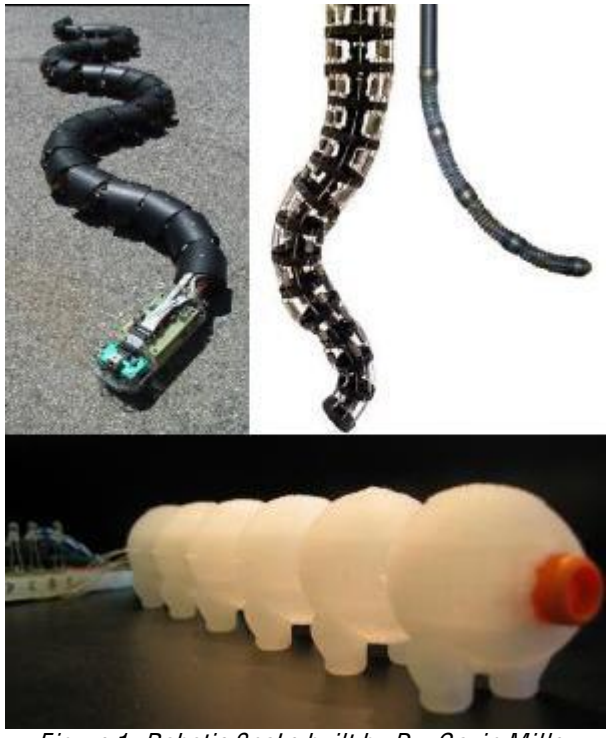


Figure 1: Robotic Snake built by Dr. Gavin Miller, Elephant Trunk Manipulator and Tendril by Clemson University, and Softbot built by Tufts University

True continuum robots, such as the Octarm [9] and the Tendril [10],[11] (Fig. 1), have continuous backbone sections which can conform around objects [12],[13],[14]. Soft robots, such as Softbot, are almost gel-like in their form [15],[16]. However, soft continuum

robots are hard to build, model and control [17],[18]. Management of the malleable and compliant properties which form a great part of their appeal is proving a major obstacle to progress in this emerging field [1].

There is an inherent tradeoff between continuous and discrete elements. For example, continuum structures can conform to their surroundings while discrete rigid links aid precise positioning. Interestingly, continuum structures in nature seem to synergize their activities with various kinds of discrete elements, as discussed in the following section. With this in mind, we argue in section 3 that with a judicious mixture of continuous/soft and discrete/hard elements, robots can be made to perform many tasks. We conclude that the structure of soft and continuum robots should depend strongly on the task the robots will be used for and the application environment.

2. Continuous Structures in Nature

Animals in nature have a wide variety of continuum structures. Arms, tails, tentacles, and various other appendages all have important functions they perform for the animal. In the following, we classify these functions into three main classes.



Figure 2: Animals using Prehensile Tails for Balance

2.1 Balance/Stability

There are many instances in the animal kingdom of single hyper-redundant or continuous limbs being used for balance, like the tail of a kangaroo or (most probably) that of a dinosaur [19]. Some gecko species use their tails for stability when they climb. Monkeys can use their prehensile tails to hold onto branches and improve their stability [20]. A prehensile tail is often wrapped around a stable solid object at a discrete location and used as an anchor for support (Fig. 2). A caterpillar is similar in that it will anchor part of its body while the top half moves around to eat. Many other creatures, such as opossums and seahorses, have prehensile tails. The tails can be used to balance on land, in the trees, or under the sea. In this sense natural continuum structures compensate for the complexity inherent in their "softness" by essentially *environmentally grounding themselves at discrete body locations*, typically coupling with hard environmental elements. Similarly, when an animal's tail is used for balance the complexity inherent in the structure is typically handled by adopting restricted classes of movement. One example of this is running. The tail compensates for the complexity of the balancing task by making simple cyclical movements or being swung out behind to counter the animal's movements [19]. Soft continuum robots could clearly benefit from adopting similar strategies.

2.2 Exploration/Sensing

Exploration and sensing are other key functions of natural continuum limbs. Snakes have many different ways to slither. (Generally slithering refers to snakes but also describes the movement of slugs and earthworms.) The four slithering types are lateral undulation, rectilinear locomotion, concertina locomotion, and sidewinding [6]. The type of motion a snake uses depends on its environment. Lateral side to side undulation is the main way snakes move [6]. Rectilinear locomotion is how large pythons and anacondas move using their belly scales [6]. Concertina movement is how snakes climb or move in limited surroundings such as tunnels [6]. Sidewinding is used to move in the desert over loose sand [6]. Under water, eels and sea snakes can wind their way through holes in the coral to find food.

Often natural continuum elements are used as both sensors and effectors. Garden eels, brittle stars, and basket stars all sway in the ocean current to detect food. When a brittle star senses food, it can fling its arm out in the general direction of the food. Then it will coil an arm around it and bring the food to its central mouth. Once again, this flinging is not arbitrary, but is simply controlled since the arm merely unfurls in the needed direction. A similar pattern of simple control, and combination of sensing and exploration, are adopted by plants such as vines (Fig. 3) [21].



Figure 3: Climbing Morning Glory Vine

Alternative natural sensing continuum appendages are whiskers and antennae. Many animals have whiskers to help with their spatial awareness. A catfish's whiskers are used to check the muck at the bottom of a river for food. The tentacles on a star-nosed mole are very sensitive, for example the animals can even smell underwater [22].



Figure 4: Octopus Opening a Jar with its Arms [23]

Here once again, it appears the natural soft/continuum elements are seldom used in isolation of discrete or hard elements. For example, an octopus will wrap its arm around an object but uses its suckers, located discretely along the arm, for fine sensing and manipulation (Fig. 4). Millipedes have a hyper-redundant body studded with numerous discretely positioned legs. Their bodies will conform to the obstacles that they crawl over while using the fine movements of their legs for adjustments. Large anacondas use their belly scales to crawl forward silently when stalking prey [6]. These three creatures all use a combination of soft and hard(er) elements. These hybrid continuum/discrete structures *incorporate discrete elements for fine resolution*, using discrete parts for fine work and their continuum anatomy for general purpose positioning.

A robot could use a continuum appendage with sensors to probe places its main body cannot reach. This would be very useful in exploration of hazardous areas.



Figure 5: Sting Ray, Komodo Dragon tail, and Bullwhip

2.3 Obstacle Removal/Grasping

Another way to use a continuum limb is to use it to remove obstructions and rapidly grasp/manipulate the environment. A whip-like structure can be flicked out to move an obstacle from the animal's path. The movement does not have to be particularly accurate since it often just needs to be cast in the correct general direction. Many animals use their tails as weapons. Komodo dragons will whip enemies and so will sting rays (Fig. 5). If considered as a weapons system, a scorpion's tail would make an interesting model. Continuous natural appendages are also used as weapons. The tentacles of a squid are used to dart out in the direction of prey [24]. Similarly, a brittle star can fling its arms in the general direction of food and then draw the arm in to feed itself.

Octopus arms, which are formidable weapons as well as effective manipulators, appear to be similarly discretely directed in the direction of objects of interest rather than having their shapes closely controlled [25]. Elephants also simplify control of their trunks by moving them within a plane oriented towards objects they desire to grasp [26]. Brittle stars manipulate objects in a similar manner as octopuses, but unlike octopuses the brittle star does not have strong suction cups on its arms. Each arm is like a snake's tail and can be used to wrap around objects. They can slither or crawl depending on the terrain. Their arms are quite dexterous and can be used to grab food and move it to the star's central mouth.

Humans can also be very effective when augmented with continuum tools. Whips, lassos, and chains are all flexible tools that can be used in a variety of ways. In the movies, Indiana Jones has used his whip to swing across gaps [27]. If a robot could do this, then it could transport itself to places it could otherwise never reach, or at least get there quicker. Ropes can be made into lassos to loop around objects. Cowboys use lassos to capture errant steers. A robot could potentially use a lasso to hook rock outcroppings to pull itself up a cliff. A grappling hook is a strongly related alternative.

A common element in all the above examples is once again *discrete control*, with the problem of close control of all degrees of freedom in the continuum structure sidestepped by making simplified motions (controlled by a *discrete* set of variables) in specific directions. In many cases, only the direction and speed need to be directly controlled. A continuum limb could similarly be used swiftly to fling obstacles out of the robot's path, or form quick but effective curling grasps.

3. Implications for Soft and Continuum Robots

The examples from nature in the previous section motivate a new look at soft continuum robots. Up to this

point, most development has been motivated by the desire to create "fully soft" continuum robot bodies with no hard or discrete elements, and to precisely control their shape through the continuum of possibilities, independent of their environment. However, it seems clear that many natural soft and continuum elements are successful precisely by incorporating discrete elements, simplifying their movements, or interacting in a way very specific to their environment. The key in all cases we have reviewed is complexity reduction, which leads to strong implications for robot development. Each of these issues is investigated in the following subsections.

3.1 Complexity Reduction

A key goal for soft continuum structures is adaptability: compliance to environmental constraints via an enhanced (essentially infinite dimensional) configuration- or shape-space. In robotics, almost all efforts so far have tried to achieve this via soft compliant bodies in controlled continuum contact with their environment. (The two main types of continuum manipulator today are tendon-driven [8],[28],[29] or pneumatically [13],[29],[30],[31] controlled.) However, the resulting decision space (and its requirements for sensing and planning) is vast. A key simplifying observation from the natural world is that *in nature, soft continuum limbs are used mostly for approximate positioning, strongly exploiting discrete elements in their structure, operation, or their environment to simplify and resolve their operation*. In all cases this allows complexity reduction: environmental contact and fine manipulation details are handled by discrete scales, legs, or suckers; the movement space is restricted to a given direction or plane, as in the movements of octopus arms and elephant trunks, or dynamic balancing of tails; imprecision due to environmental forces is alleviated via stabilization using tails, anchors, or tongues. All these concepts could be exploited in novel robotic counterparts.

Another issue which appears to have been rarely considered as a major issue in robotics, but which appears critical in nature, is that of the underlying nature of control. Continuous control (regulation of the system to an arbitrary shape throughout its workspace) enables precise operation. Continuous control in the above sense is the most commonly used form of control in conventional rigid link robots. This allows the control system to compensate for (indeed, take advantage of) the simplicity of the discrete rigid link structure to achieve the precise positioning desired in structured applications such as manufacturing. However, effective continuous control of continuum robotic structures is proving extremely difficult to achieve [9],[10]. The increased complexity in continuum structures is hard to either model well, or to provide sufficient actuator inputs for, to enable consistent control.

Nature however suggests an alternative approach to complexity reduction in control. If a continuous manipulator is controlled discretely (restricting the allowable shapes of the system to a finite set, or a shape set defined by a finite set of inputs) then it will be much easier to control. Clearly many, if not most, continuum structures in nature are controlled in a discrete (as defined above) manner, as discussed in section 2. Notice that in this case the compliance inherent in the continuum structure allows the system to adapt to compensate for the simplicity of the control. The concept of central pattern generators has been used to define the shapes and simplify the control of some snake-like robots [2]. An extension of these ideas to the wider class of continuum robots could enable practical control of behaviors similar to octopus arm or elephant trunk manipulation. Binary control (enabling "whip-like" movements similar to those discussed in section 2.3) has corresponding potential for continuous manipulators in dynamic tasks.

3.2 Design Implications

A common theme in the above discussion is the effectiveness of the combination of continuous and discrete elements. One direct way to achieve this synergy is by incorporating both types of structure on an overall robot design, a *hybrid continuum/discrete robot*.



Figure 6: Fictional Snake-Arm Robots (B-9, Sentinel, Doc Ock)



Figure 7: Real Snake-Arm Robots from OC Robotics [28]

Some hybrid continuum/discrete robot designs have previously been considered. One possibility is to have a continuous arm and simple gripper, like the trunk of an elephant which can pick up a peanut with its finger-like projections. A robot with a continuous arm and discrete gripper is generally called a snake-arm robot. There are numerous examples of snake-arm robots in science fiction, but few in real life (Fig. 6). Science fiction can serve as inspiration just as well as nature. For example, the flip-top communicators from Star Trek could have inspired the cell phone [32]. However, while there are multiple examples of fictional continuum robots, there are very few continuum robots in reality. Most real snake-arm robots are discrete, using many joints to become hyper-redundant [8]. Snake-arm robots are used in the nuclear industry and for robotic surgery [28],[33] (Fig. 7). The advantage of having a continuous arm with a discrete gripper is that it would be like having a tentacle with a hand on its end, providing impressive maneuverability with a simple, if not particularly dexterous, grasp (Fig. 8).



Figure 8: Discrete Arm with Continuous Fingers [34]

The question of whether to use discrete or continuous parts is an interesting one, with the answer depending on how the robot is desired to move and what its function will be. Let us consider an example consisting of an arm and a manipulator. When would it be best for the arm to be continuous (i.e. the snake arm approach)? Having a continuous arm would let the manipulator reach places that might otherwise be unreachable. The three most prominent continuum structures in nature are the octopus arm, elephant trunk, and tongues. Underwater animals can have soft continuum arms because they are affected little by gravity. Most tongues are short and stout so they can also ignore gravity. However, an elephant's trunk is affected by gravity and can be seen swinging as the elephant moves its head from side to side. Adding a discrete gripper onto the end of a continuum trunk would cause an even greater sag in the robot.



Figure 9: Giraffe Using its Tongue to Extend its Reach

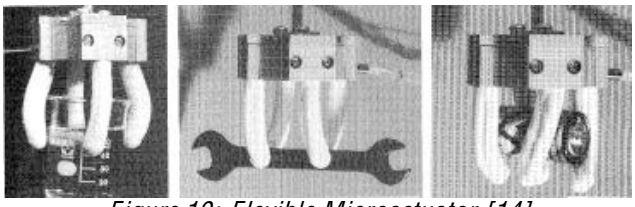


Figure 10: Flexible Microactuator [14]

An interesting alternative design approach would be to use a serial discrete link arm and a continuous end effector. This model is less frequently explored than the snake-arm robots, even in fiction. The giraffe is a natural example. The concept can be thought of as a discretely built neck with a continuous tongue as a manipulator. It could use its prehensile tongue to reach places it cannot fit its neck into (Fig. 9). Unlike the giraffe's tongue, most robotics end effectors are in the form of hands or simple grippers. One example of a hand with continuum elements is the AMADEUS dexterous underwater gripper [1]. The flexible microactuator built by the Toshiba Corporation is much smaller and could be used for more delicate tasks [14] (Fig. 10). This type of robot would be like having an octopus for a hand. It would be able to manipulate objects dexterously and do things that current discrete link manipulators can't. One issue with the manipulator is how many fingers it should have and how many joints for each. Four fingers is usually enough to manipulate objects in 3D. As with a continuous arm, continuous fingers would have sagging and torsion issues. However, this would be less than for a continuum trunk, and the continuum end effector could compensate for gravity and/or changes in the environment such as the movement of its goal, just like a giraffe's tongue can move to catch leaves blown by the wind. There are few examples of a discrete arm with a continuous end effector in nature. However, there are also few examples of the wheel and yet it is one of humanity's most useful inventions. Roboticists should not be limited by nature, but also look to their imagination for inspiration.

A third alternative design would be a non-serial hybrid continuum/discrete structure. These structures

might be ideal for fine manipulation. One natural model for a continuous end effector is the basket star (Fig. 11), which has similarities with the brittle star (Fig. 12). Rather than a brittle star's five limbs, the basket star has a fractal-like pattern of tentacles. It is almost tree-like in its form. A basket star would make a great manipulator if you could control it [35]. A manipulator with rigid linked fingers cannot conform to an object it intends to grasp, but continuum fingers can wrap around an object like the grasp of an octopus. This would result in a better grip with less chance of the object being dropped.



Figure 11: Illustration of a Basket star



Figure 12: Illustration of a Brittle star

A key question raised by the earlier discussion is how motions for soft continuum robots should be planned and controlled. Motivated by the examples from nature reviewed here, we argue that simplifications should be sought where possible, as discussed in the previous subsection. The strategy of restricting and controlling movements to a plane is appealing and clearly successful for many animals, and likely to be most practical for continuum robotic elements. For hybrid continuous/discrete robots, it would appear to be best for the discrete part of the robot to be controlled continuously (and vice versa) so that the discrete part is concerned with precision, and the continuum part with more global environmental accommodation. For example, the fractal-like pattern of the basket star end effector design would be hard to control continuously so discrete control of the continuum elements would be most appropriate.

Additionally, it seems clear that the structure of these new forms of robots with soft continuum elements robot should be dependent on the environment they will operate in. The traditional approach of building general-purpose robots has only been partially successful – while traditional robots are used for a variety of tasks in structured environments, typically those environments have been heavily engineered to fit the robots capabilities. Therefore robots have not significantly penetrated the inherently unstructured environments of the "real world". Soft continuum robots are explicitly intended to enter that world, and the lesson from their counterparts in the natural world is that success generally implies specialization and matching to the environment. We believe that, at least in the medium term, the same is likely to be true for continuum robots.

Finally, notice that there are other types of locomotion not discussed here for which soft continuum robots might be useful. Legged locomotion and slithering are the two main types of terrestrial locomotion, but some creatures can configure their bodies to roll around like wheels [36]. In nature the caterpillar of the Mother-of-Pearl moth and the stomatopod shrimp (*Nannosquilla decemspinosa*) are two of the few rolling animals [37]. There are many types of robots that mimic the legged locomotion of animals, but wheeled robots are more common and more practical at this time. Rolling is usually a secondary form of motion in nature with the primary form being legged locomotion. Rolling is complex to control and a non-wheeled rolling continuum robot would be hard to steer with no stable base for sensors. However, new types of modular and shape shifting robots might find this mode useful in the future.

4. Conclusion

We have discussed the design and operation of the emerging class of soft and continuum robots, contrasting the state of the art in robotics to date with the counterparts in the natural world. We note that natural continuum locomotors or manipulators almost invariably use design modifications or specialized "tricks" to simplify their operation. The complexity reduction achieved is usually based on synergy of soft/continuum with hard/discrete elements (in the structure and/or operation of the robots). We have discussed implications

for the design and successful operation of novel continuum robots. A key inference is that construction of a soft continuum robot should depend on the environment it will be used in. It also appears that appropriate combination of continuum and discrete, or soft and hard, elements is likely to significantly improve the performance of these robots.

Acknowledgements

This work is supported in part by NASA contract NNJ05HF74G and in part by NASA/EPSCoR grant NCC5-575.

References

[1] Robinson, G. and Davies, J. B. C. (1999). Continuum robots—A state of the art. In *IEEE International Conference on Robotics and Automation*, pages 2849–2854. Detroit, MI.

[2] Crespi, A. and Ijspeert, A. J. (2006). Amphibot ii: An amphibious snake robot that crawls and swims using a central pattern generator. In *9th International Conference on Climbing and Walking Robots (CLAWAR 2006)*, pages 19-27.

[3] Miller G. (2007). Snake Robots. <http://www.snakerobots.com/>.

[4] Ostrowski, J. and Burdick, J. (1998). The geometric mechanics of undulatory robotic locomotion. *International Journal of Robotics Research*, 17(7):683-701.

[5] Wright, C., Johnson, A., Peck, A., McCord, Z., Naaktgeboren, A., Gianfortoni, P., Gonzalez-Rivero, M., Hatton, R., and Choset, H. (2007). Design of a modular snake robot. In *IEEE International Conference on Intelligent Robots and Systems*, pages 2609-2614.

[6] Hirose, S. (1993). *Biologically Inspired Robots*. Oxford University Press. New York, NY.

[7] Lipkin, K., Brown, I., Peck, A., Choset, H., Rembisz, J., Gianfortoni, P., and Naaktgeboren, A. (2007). Differentiable and piecewise differentiable gaits for snake robots. In *IEEE International Conference on Intelligent Robots and Systems*, pages 1864-1869. San Diego, CA.

[8] Hannan, M. W. and Walker, I. D. (2001). Analysis and experiments with an elephant's trunk robot. *Advanced Robotics*, 15:847-858.

[9] McMahan, W., Pritts, M., Chitrakaran, V., Dienno, D., Grissom, M., Jones, B., Csencsits, M., Rahn, C. D., Dawson, D., and Walker, I. D. (May 2006) Field Trials and Testing of "OCTARM" Continuum Robots. In *IEEE International Conference on Robotics and Automation*, pages 2336-2341. Orlando, FL.

[10] Cowan, L. (2007). Azimuth, elevation, and coupling compensation for the Tendril. Technical Report, Department of ECE, Clemson University. <http://www.ces.clemson.edu/~ianw/Tendril.pdf>.

[11] Mehling, J., Diftler, M., Chu, M., and Valvo, M. (2006). A Minimally Invasive Tendril Robot for In-Space Inspection. In *IEEE International Conference on Biomedical Robotics and Biomechatronics*, pages 690-695. Pisa, Italy.

[12] Cieslak, R. and Morecki, A. (1999). Elephant trunk type elastic manipulator - a tool for bulk and liquid materials transportation. *Robotica*, 17:11-16.

[13] Tsukagoshi, H., Kitagawa, A., and Segawa, M. (2001). Active hose: an artificial elephant's nose with maneuverability for rescue operation. In *IEEE International Conference on Robotics and Automation*, pages 2454-2459. Seoul and Korea.

[14] Suzumori, K., Iikura, S., and Tanaka, H. (1991). Development of flexible microactuator and its application to robotic mechanisms. In *IEEE International Conference on Robotics and Automation*, pages 1622-1627.

[15] Tufts University (30 January 2007). Biomimetic Technologies Project Will Create First Soft-Bodied Robots. *ScienceDaily.* <http://www.sciencedaily.com/releases/2007/01/070128105355.htm>.

[16] Trimmer, B., Takesian, A., and Sweet, B. (2006). Caterpillar locomotion: A new model for soft-bodied climbing and burrowing robots. In *7th International Symposium on Technology and the Mine Problem*, Monterey, CA.

[17] Jones, B. and Walker, I. D. (2006). Practical kinematics for real-time implementation of continuum robots. In *IEEE International Conference on Robotics and Automation,* pages 1840-1847. Orlando, FL.

[18] Gravagne, I. A. and Walker, I. D. (2000). Kinematic transformations for remotely actuated planar continuum robots. In *IEEE International Conference on Intelligent Robots and Systems*, pages 19-26. San Francisco, CA.

[19] Raibert, M. (1986). *Legged Robots That Balance*. MIT Press.

[20] Covey, R. (2005). Prehensile tail use during feeding and foraging of white-faced capuchins, Cebus capucinus. Department of Anthropology, Ohio State University. <https://kb.osu.edu/dspace/bitstream/1811/5877/1/RyanCovey.pdf>.

[21] Darwin, C. (1888). *The Movements and Habitats of Climbing Plants*. Appleton, NY.

[22] Catania, K. (21 December 2006). Olfaction: Underwater 'sniffing' by semi-aquatic mammals. *Nature*, 444:1024-1025.

[23] 2008. [Online]. Available: http://www.tonmo.com/forums/showthread.php?t=11319

[24] Leeuwen, J. L. and Kier, W. M. (1997). Functional Design of Tentacles in Squid: Linking Sarcomere Ultrastructure to Gross

Morphological Dynamics. *Philosophical Transactions of the Royal Society of London*, 352:551-571.

[25] Sumbre, G., Gutfreund, Y., Fiorito, G., Flash, T., and Hochner, B. (September 2001). Control of octopus arm extension by a peripheral motor program. *Science*, 293:1845-1848.

[26] Martin, F. and Niemitz, C. (13 June 2003). How do African elephants (Loxodonta africana) optimise goal-directed trunk movements?. *Jahresversammlung der Dt. Zool. Ges. und der Dt. Ges. f. Parasitologie*, 96:159. Berlin, Germany.

[27] 2008. [Online]. Available: http://imdb.com/title/tt0082971/

[28] OC Robotics. (2007). <http://www.ocrobotics.com/>.

[29] Immega, G. and Antonelli, K. (1995). The ksi tentacle manipulator. In *IEEE International Conference on Robotics and Automation*, pages 3149-3154.

[30] Pritts, M. B. and Rahn, C. D. (2004). Design of an artificial muscle continuum robot. In *IEEE International Conference on Robotics and Automation*, pages 4742-4746.

[31] Wilson, J. F., Li, D., Chen, Z., and George, R. T. (1993). Flexible robot manipulators and grippers: Relatives of elephant trunks and squid tentacles. *Robots and Biological Systems: Toward a New Bionics?*, pages 474-479.

[32] Laytner, L. (2007). Star Trek Tech. *Edit International*.

[33] Simaan, N., Taylor, R., and Flint, P. (2004). A dexterous system for laryngeal surgery. In *IEEE International Conference on Robotics and Automation*, pages 351-357. New Orleans, LA.

[34] 2008. [Online]. Available: http://ad-blood.blogspot.com/2007/05/sanyo-xacti-octopushand.html

[35] Moravec, H. and Easudes, J. (1999). Fractal branching ultra-dexterous robots. *NASA Advanced Concepts Research Projects*. <http://www.frc.ri.cmu.edu/users/hpm/project.archive/robot.papers/1999/NASA.report.99/>.

[36] Biewener, A. (2003). *Animal Locomotion*. Oxford University Press, New York, NY.

[37] Armour, R. and Vincent, J. (2006). Rolling in nature and robotics: a review. In *Journal of Bionic Engineering*, 3(4):195-208.

From Single Cell to Simple Creature Morphology and Metabolism

Sylvain Cussat-Blanc, Hervé Luga and Yves Duthen
Institut de Recherche en Informatique de Toulouse
Université de Toulouse - CNRS - UMR 5505
2 rue du Doyen-Gabriel-Marty
31042 Toulouse Cedex 9, France
{sylvain.cussat-blanc, herve.luga, yves.duthen}@irit.fr

Abstract

In order to produce diversity in virtual creatures to populate virtual worlds, different techniques exist. Some of these use blocks or sticks. In this morphological approach, blocks and sticks can be considered as organs, which means body parts able to perform different functions. Another approach, artificial embryogenesis, consists in developing organisms from a single cell. In this paper, we propose a bridge between these two approaches : a model that will create creatures with a particular morphology and which is organized in organs. The creature development will start from a single cell. In this paper, we propose a unique model able to produce organisms that perform a specific function and to produce organisms with a user-defined morphology.

Introduction

Several models exist for creating artificial creatures. These models use different levels of abstraction to produce creatures of various shapes and sizes. Whereas the morphological approach produces relatively large creatures as in (Sims, 1994; Lassabe et al., 2007), embryogenic models produce creatures composed of hundreds of cells starting from a unique cell (Chavoya and Duthen, 2007; Dellaert and Beer, 1994; Stewart et al., 2005).

This paper details our model of cellular development, *Cell2Organ* (Cussat-Blanc et al., 2007). For the purpose of creating complete creatures composed of different organs, we propose a model able to produce organisms that perform specific functions. These organisms respect the biological definition of an organ. In other words, they are a "specialized cell regrouping that performs specific function or a group of functions". Our model contains an environment with a simple artificial chemistry (Rasmussen et al., 2003; Dittrich et al., 2001; Hutton, 2007; Ono and Ikegami, 1999) and cells that perform different actions. Cells are able to self-replicating and to specialize themselves to optimize specific actions instead of others. Moreover, we show that *Cell2Organ* can also produce simple creature shapes. The final aim of our project is to develop a complete creature starting from a unique cell.

This paper is organized in four sections. Section 2 presents related works about artificial creatures development, presenting artificial morphogenesis, cellular automata and already existing works about artificial embryogenesis. Section 3 presents our model of cellular development, *Cell2Organ*, starting with a description of the environment functioning and the mechanisms used by our artificial cell to interact with the environment. Section 4 presents different experiments using this model. The possibilities of the model are shown by the development of two types of organisms : a primitive organ able to move substrate in the environment and two creatures with particular morphologies. These experiments point to the possibility of simulating, in a simplified way, different approaches to organism growth. In the final section, we conclude by outlining different possible development paths for this model.

Related works

Artificial morphogenesis

Several projects have tried to generate artificial creatures well adapted to their environment. For example, in his famous works, Karl Sims (Sims, 1994) uses blocks with different properties such as size, shape, contact sensor positions or block layout. Komosinski also creates Framsticks creatures (Komosinski and Ulatowski, 1999) using an equivalent architecture: sticks replace blocks but creature functioning is comparable to Karl Sims' work: he uses a neural network to coordinate creature movements. Nicolas Lassabe improved Sims' work by using a more complex environment (Lassabe et al., 2007). Lassabe's creatures are able to climb a stairway or to practice skateboarding.

The aforementioned creatures use high-level components to create their morphology and their behavioral controller. A more biological-inspired approach was introduced by Dawkins in (Dawkins, 1986). Using simple rules to draw continuous segments, he developed a model able to create small graphic creatures. The addition of behaviors in these simple life forms allows the creation of a complex 2-D virtual world (Ventrella, 1998) where small filiform creatures co-evolve in an environment composed of energy sources.

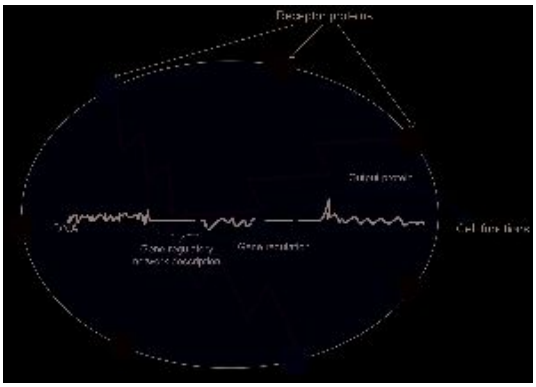


Figure 1: Scheme of the GRN action in cell duplication.

Each creature has a vital energy level and must survive in the environment, looking for food produced by the death of other creatures. This model produces a complete ecosystem with its own food chain. Creatures are also able to reproduce among themselves to create new life forms. EvolGL (Garcia Carbajal et al., 2004) is another 3D pond life project where creatures have different classes, such as herbivorous, carnivorous or omnivorous, which allows the emergence of survival strategies.

Using lower level components, cellular automata use neighborhood rules to evolve a cell matrix. The rules give the *t+1* state of each cell according to the cell neighbor's *t* state. Using this method, John H. Conway (Gardner, 1970) creates interesting patterns such as gliders, pulsars, etc.

Artificial Embryogenesis

One of the first works on artificial embryogenesis was that of Hugo de Garis (de Garis, 1999). Using a cellular automaton, he developed 2D shapes. The cellular automata rules were evolved with a genetic algorithm. The aim was to generate desired shapes like letters.

Another important goal of artificial embryogenesis is cell specialization. Different works on cell specialization already exist. In most cases, they use a Genetic Regulatory Network (GRN), just as in nature.

In nature, the organism's cells can have different functions, all of which are specified in the organism's genome and regulated by a Gene Regulatory Network (GRN) (Davidson, 2006). Cells get input signals from the environment thanks to receptor proteins. The GRN, described in the organism's genome, uses these signals to activate or inhibit the transcription of different genes in the messenger RNA, the future cell's DNA protein template. The expression of these genes will specify the cell's functions. Figure 1 shows (in a simplified way) the functioning of the GRN.

This nature inspired model was designed by Banzhaf in (Banzhaf, 2003). In this work, each gene beginning is marked by a starting pattern, named "promoter". Before the coding of the gene itself, enhancer and inhibitor sites allow the regulation of its behavior. In (Chavoya and Duthen, 2007), Chavoya and Duthen introduced another model in which the gene regulation system is encoded at the beginning of the genome. It consists of a series of inhibitor sites, enhancer sites and regulatory proteins. The production of each regulatory protein is conditioned by the inhibitor/enhancer sites. The concentration of this protein determines the cell function's activation or inhibition : if the concentration level is over a certain threshold, the gene is activated and so are the corresponding functions.

A different approach is the Random Boolean Network (RBN) first presented by Kauffman (Kauffman, 1969) and reused by Dellaert (Dellaert and Beer, 1994). A RBN is a network where each node has a boolean state: activate or inactivate. The nodes are interconnected by boolean functions, represented by edges in the net. The state of a node at time $t + 1$ depends on its particular boolean function applied to the values of its inputs at time t. The mapping to the gene regulatory network is simple: each node of the net corresponds to a gene and each boolean function represents the activity regulation of the gene. The cell function will be determined during the interpretation of the genome.

Eggenberger Hotz (Eggenberger Hotz, 2004) imagines a concept able to produce a simple creature with a user defined shape able to move in an environment just using a GRN. Cells rhythmically emit molecules that modify the adhesion properties between cells and between cells and the environment. He develops a simple simulator and produces a T-shape that grows and move in the environment.

The aim of our work is to make a bridge between artificial morphogeny and artificial embryogenesis to produce virtual creatures. We decide to use the hypothesis that blocks and sticks can be considered as organs, that is to say body parts of the creature able to carry one or more specific functions. Using developmental techniques of creature growth, we could create these organs starting from a single cell. In this way, the cell must be able to specialize itself into a cell more adapted to the environment. The cell organization in tissues (that is in cell groups that have the same function) and then the tissue organization will allow the creation of organs. After creating a library of organs, we will just have to assemble them to create a creature adapted to the environment with a morphological approach. This paper presents the embryogenic approach of the problem, and especially the creature shape development. The next section details the model, starting with the environment and, then, showing the cell mechanisms.

Cell2Organ : a cellular developmental model

The environment

To reduce the simulation computation time, we implement the environment as a 2-D toric grid. This choice allows an important decrease in the simulation's complexity.

The environment contains different substrates. They spread in the grid, minimizing the variation of substrate quantities between two neighbor crosses of the grid. This spreading is enacted in two stages, as illustrated by Figure 2

- First, the substrate spreads to the 4 cardinal points.
- Then, if the substrate quantity is sufficient, the substrate spreads to the diagonal crosses.

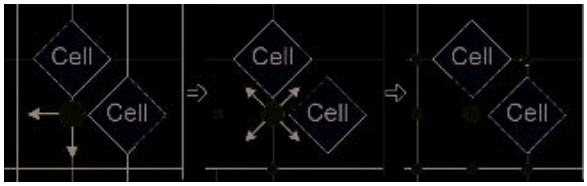


Figure 2: Example of spreading substrate in the environment.

Our model integrates a highly simplified model of artificial chemistry. Many works exist on artificial chemistry (Dittrich et al., 2001; Rasmussen et al., 2003). In these works, the artificial chemistry is highly developed and allows a good simulation of cell mechanisms. For example in (Ono and Ikegami, 1999), the cell division and the cell membrane formation and maintaining are highly realistic. However, the complexity of such a model is very great and does not support a high number of cells. In our model, the properties of artificial chemistry defined in (Dittrich et al., 2001) have been simplified.

Our molecules, named substrates, have different properties like diffusion speed or color, and can interact with other substrates. This interaction between substrates can be viewed as a typical chemical reaction: using different substrates, the transformation will create new substrates, emitting or consuming energy. For example, the transformation $2A + B \rightarrow C\ (+50)$ denotes that, using 2 units of substrate A and 1 unit of B, a unit of C is created, emitting 50 units of energy. To reduce the complexity at the maximum, the environment contains a list of available substrate transformations. The substrate reactions can only be triggered by cells. Then, in the previous example, from a biological point of view, C can viewed as waste from a cell which has the ability to convert A and B into energy.

To modify this environment, cells interact with the environment. They have different abilities and must perform a global action defined by the user. This action can be very diverse: harvest substrate, modify environment, create shapes or simply survive as long as possible. The next section describes cell functioning.

The cells

Cells evolve in the environment, more precisely on the environment diffusion grid. Each cell contains sensors and has different abilities (or actions). An action selection system allows the cell to select the best action to perform at any moment of the simulation. Finally, a representation of a GRN is inside the cell to allow specialization during duplication. Figure 3 is a global representation of our artificial cells.

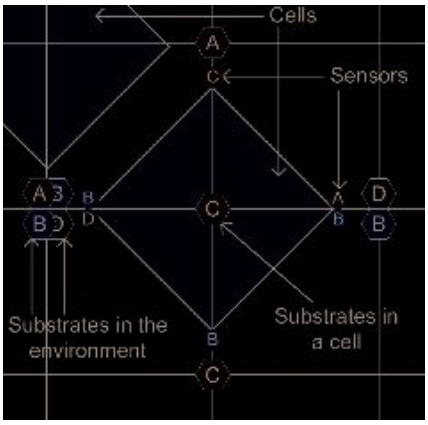


Figure 3: Scheme of a cell in an artificial environment. It contains substrates (hexagons) and corresponding sensors (circles)

Sensors Each cell contains different density sensors positioned at each cell corner. Sensors allow the cell to measure the amounts of substrates available in the cell's Von Neumann neighborhood. For each substrate in the environment, a corresponding sensor exists. Only this corresponding sensor can compute the density of the substrate. The list of available sensors and their position in the cell is described in the genetic code.

For example, in Figure 3, the cell has sensors for B and D substrates in the left corner. The results of the measure of the corresponding substrate densities are :

- 2 units for B substrate because of the presence of 2 units of B substrates in the left cross of the cell,
- 1 unit for D substrate.

Actions To interact with the environment, cells can perform different actions:

- The *substrate transformation* allows the cell to trigger a substrate reaction as previously described. To start, all the needed substrates on the left part of the equation must be present in the cell, that is, the needed substrates must be in the same intersection as the cell. In result of the reaction, the vital energy is increased or decreased (depending of the reaction properties), the needed substrates are destroyed and the new substrate is created.
- The cell can *absorb* or *reject* substrates in the environment. These actions allow the cell to move substrates

from one place to another. These actions, particularly the first , are important to trigger a substrate transformation.

- The *duplication* action allows the cell to create a new cell. We give details about this action in the next section.
- *Survive* is an action that allows the cell to wait for a signal from the environment to do something.
- *Apoptosis* allows the cell to autodestruct. This action can be useful to free a place for a more specialized cell for example.

The previous list is not final. Our model must be able to allow us to add new actions easily. Like sensors, all actions are not available for the cell: the genetic code will give the available action list.

Cells contain an action selection system. This system is inspired by classifier systems (Holland and Reitman, 1978). It uses data given by sensors to select the best action to perform. The selection system can be viewed as a rule database, where each rule is composed of three parts:

- The *precondition* describes when the action can be triggered. It is composed of a list of sensor value intervals that describe the best substrate densities in the neighborhood to trigger the action.
- The *action* gives the action that must be performed if the corresponding precondition is respected.
- The *priority* that allows the selection of only one action if more than one can be performed. The higher the coefficient, the more probable is the selection of the rule.

Action selection rules can be, for example :

$$\begin{array}{lll} (SensorA = 1) & and & (3 < SensorC < 7)\ and \\ (SensorB = 0) & \rightarrow & (ActionA)\ (23) \\ (SensorC = 3) & \rightarrow & (ActionB)\ (17) \\ & \rightarrow & (ActionC)\ (13) \end{array}$$

In this example, $ActionA$ will be performed if and only if $SensorA$ value is equal to 1 unit, $SensorB$ does not detect the presence of its associate substrate and $SensorC$ value is more than 3 units and less than 7. $ActionC$ does not contain a precondition. It means that this action can always be performed. The priority coefficients sort actions in the order $ActionA > ActionB > ActionC$ if different actions are possible.

In the list of possible actions, the cell can duplicate itself. We will now examine this action in detail.

Duplication The duplication is an action that can be performed by the cell if the next conditions are respected:

- The cell must have at least one free neighbor cross to create the new cell.
- The cell must have enough vital energy to perform the duplication. The vital energy level need is defined during the specification of the environment.
- A list of conditions can be added during the modelization of the environment. For example, some substrates can be needed to create a new cell.

The new cell created after duplication is completely independent and interacts with the environment. During duplication, the cell can be specialized to optimize a group of actions instead of others actions. In nature, this specialization is carried out by the GRN. In our model, we imagine a mechanism that plays the part of a GRN. Each action has an efficiency coefficient that corresponds to the action optimization level : the higher the coefficient, the lower the cost of vital energy. Moreover, if the coefficient is null, the action is not yet available for the cell. Finally, the sum of efficiency coefficients must remain constant during the simulation. In other words, if an action is optimized increasing its efficiency coefficient during duplication, another efficiency coefficient (or a group of them) has to be decreased.

The cell is specialized by varying the efficiency coefficients during duplication. A network built as follow gives the rules of these variations:

- the network's nodes represent cell actions with their efficiency coefficients,
- the network's edges are weighted. The edge's weight (a real number in the interval [0,1]) represents the efficiency coefficient quantity that will be transferred during the duplication.

Figure 4 is an example of our GRN. $(A, 35\%)$, $(B, 25\%)$, $(C, 17\%)$, $(D, 23\%)$ are cell actions with their associated efficiency coefficient. The edge between 2 actions represents the amount of efficiency coefficient that will be transferred during duplication. For example, the weighted edge between A and B means that after one duplication, 30 percents of the A action efficiency coefficient will be transferred to the B action. After four duplications, we can see that the actions B and C respectively have been optimized to the detriment of the actions A and D. According to this simple example, we can say that the cell function of the organism has been specialized during the duplication process.

We have implemented this model in Java using a multithreaded architecture: cells are coded as independent threads. Cells can communicate using the environment and substrate exchanges. We made such a choice because of the development of massive parallel computer architectures such as multi-processor machines, increasingly connected in computation grid. This parellelization allows an increase in the number of tasks executed at the same time.

Our model must be able to generate two types of artificial creatures: organs and user defined shapes. The next experiments show that it is possible to accomplish this. The first

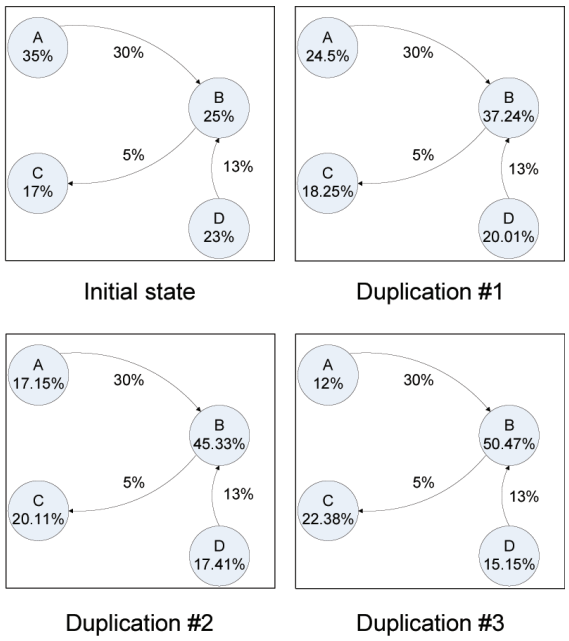


Figure 4: Modelization of an example of the Gene Regulatory Network. A, B, C and D are 4 actions with their efficiency coefficient. The transfer coefficients are given by the arrows.

experiment consists in developing a system able to move substrates in the environment whereas the second one creates simple shapes like starfish or jellyfish.

To find the creature the most adapted to a specific problem, we use a genetic algorithm. Each creature is coded with a genome composed of three different chromosomes:

- The list of available actions, a subset of the environment possible actions. This list allows the cell to activate or inhibit some actions.
- The action selection system that contains a list of rule to apply actions.
- The gene regulation network that allows cell specification during duplication.

The creature is tested in its environment that returns the score at the end of the simulation. To increase the genetic algorithm power, we use a computational grid parallelized genetic algorithm. This parallelization allows the computation of hundreds of creatures at the same time.

Experiments

Developing a transfer system

The first experimentation consists in developing a simple organ : a transfer system. In other words, the cell structure must be able to transport substrate from one point to another. To do that, we imagine an environment composed of 2 substrates:

- The red is the substrate that must be moved by the organism. This substrate has the specificity not to spread in the environment, in order not to impact on the organism work.
- A gray that will be used by the cell as fuel and duplication material.

The cell can perform the following actions:

- duplicate (needs one gray substrate and vital energy),
- absorb or reject substrate (consume vital energy),
- transform one gray substrate in vital energy.

We place 10 red substrate units into a specific cross of the grid (at the top left of the environment) and diffuse gray substrate all over the environment. The creature's score is given by the squared sum of the red substrate distance to the goal point (at the bottom right of the environment). The parameters of the genetic algorithm are:

- selection: 7 tournament competition with elitism,
- mutation rate: 5%; crossover rate: 65%,
- substitution: worst individuals,
- population size: 500 individuals,

Figure 6 shows the convergence curve of the genetic algorithm. It shows the variation of the minimum, the average and the maximum fitness of the population for each generation. The genetic algorithm's aim is to maximize fitness, which is the creature score. A relevant organism appears quickly. After 3 generations, the organism is able to move the red substrates but not in the right direction. After 10 generations, it is able to move closer to the goal point. The genetic algorithm converges after 22 generations (the average fitness is close to the best).

Figure 5 shows the development of the best organism[1]. We can see that only the cells on the way from the initial point to the end point are created. Moreover, the organism uses absorption and rejection actions to transfer the substrate gradually. Cells that overtake the final point die quickly so as not to interact in the transfer. During the convergence of the genetic algorithm, it is interesting to observe the evolution of the organism strategy towards the best solution. The first step is to learn to survive in the environment, absorbing gray substrate and transforming it in vital energy. The next step is to learn to duplicate in the right direction. Intermediate solution organisms are able to transport the red substrate

[1] Videos of all presented creatures in this paper are available on the website http://www.irit.fr/∼Sylvain.Cussat-Blanc

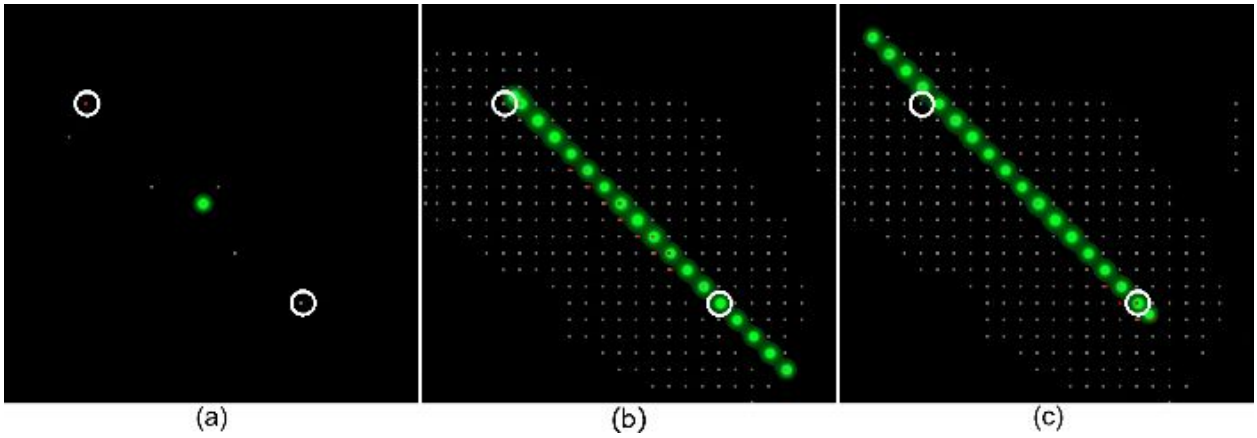


Figure 5: Our artificial transfer system. (a) Beginning of the simulation. (b) The creature develops itself to create the structure and begin the substrate transfert. (c) The creature transfers the substrate from the initial state (circle on top left) to the final state (circle on bottom right).

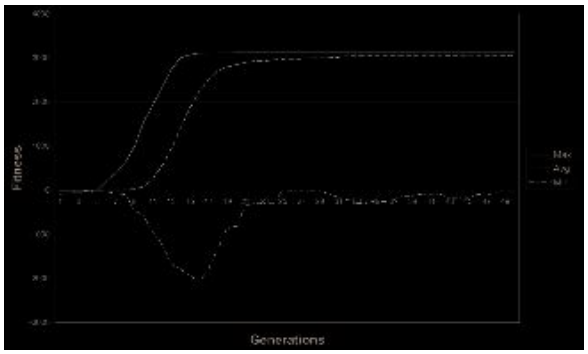


Figure 6: Smooth curve of the minimum, average and maximum organism fitness. The genetic algorithm must minimize the sum of the squared distance from the red substrate to the goal point.

from the initial point near to the goal. The organism also develops itself throughout the environment, scattering some units of the substrate in the environment. As shown in Figure 5, this organism deploys itself only on the best trajectory, decreasing the substrate scattering probability.

Creating simple shapes

In this experiment, we want to generate simple creatures with a user-designed morphology. The goal of such an experiment is to simulate the growth of more complex creatures, like those of Sims (Sims, 1994).

5 different substrates are needed to generate these shapes:

- *Water* gives energy to cells by transformation ($Water \rightarrow (+30)$). This substrate diffuses in the environment.
- Four different *morphogen* substrates, here named *NW*, *NE*, *SW* and *SE*, show four division directions to cells. These substrates do not diffuse in the environment so as not to interact with the simulation. The designer of the creature positions them in the environment.

4 different actions are associated to these substrates:

- *duplication* consumes energy and one unit of *Water*,
- *water transformation* allows the cell to trigger a transformation of one substrate of *Water* into vital energy,
- *water absorption* allows the cell to pick up water from the environment,
- *apoptosis* allows the cell to autodestruct if it wishes (for example if the cell is not in the desired shape).

To obtain the required creature morphology, the genetic algorithm fitness is calculated after a chosen simulation time and is given by the next simple formula :

- if the cell is inside the desired shape, the fitness value is increased by 2 units,
- if the cell is outside the desired shape, the fitness value is decreased by 1 unit.

The first simple morphology we try to develop using this environment is a starfish[1]. To do that, we place morphogens in the environment to lead the cell divisions. Figure 7 gives the result of the genetic algorithm. We can observe that the desired shape is obtained. It is interesting to study the action selection system rules produced by the genetic algorithm:

$$\begin{aligned} (SensorNE = 1) &\rightarrow (DuplicateNE)\ (6) \\ (SensorNW = 1) &\rightarrow (DuplicateNW)\ (5) \\ (SensorSE = 1) &\rightarrow (DuplicateSE)\ (4) \end{aligned}$$

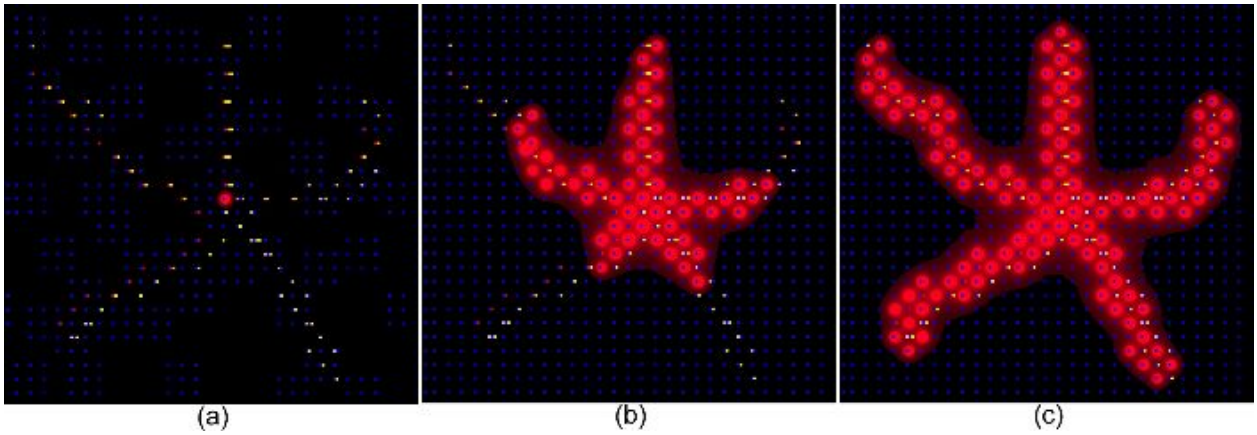


Figure 7: The starfish growth. (a) Beginning of the simulation. (b) The starfish develops itself following the morphogens. (c) The starfish stops its growth when the desired shape is obtained.

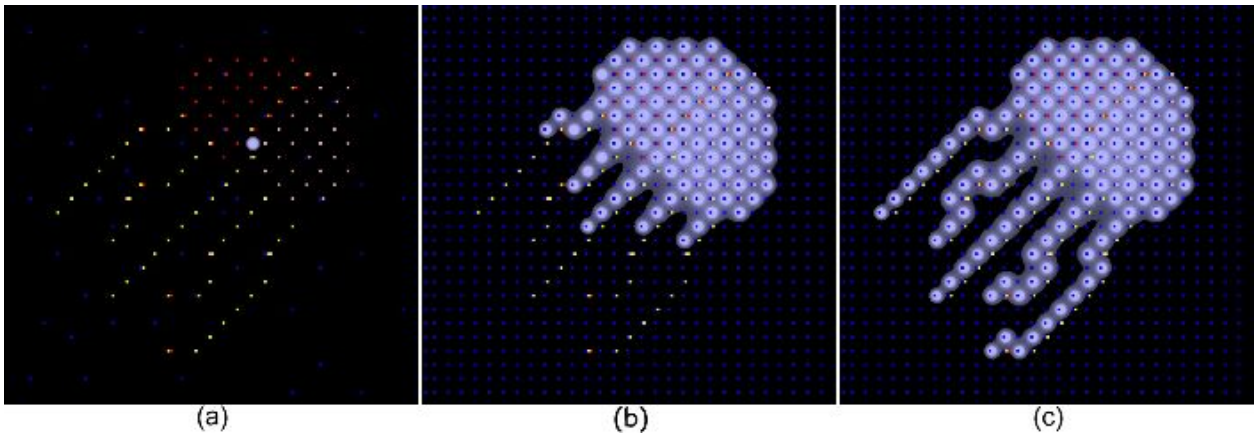


Figure 8: The jellyfish growth. (a) Beginning of the simulation. (b) The jellyfish develops itself following the morphogens. (c) The jellyfish stops its growth when the desired shape is obtained.

$$\begin{aligned} (SensorSW = 1) &\rightarrow (DuplicateSW)\ (3) \\ &\rightarrow (TransformWater)\ (2) \\ (SensorWater = 1) &\rightarrow (AborbWater)\ (1) \\ &\rightarrow (DoNothing)\ (0) \end{aligned}$$

This selection system shows that the genetic algorithm correctly uses the information given by the environment to follow the growth scheme given by the user. Moreover, duplications are always prior in relation to other actions to accumulate vital energy without using it. The last remark we can make about these rules is that the organism never uses *apoptosis* during growth. The organism assumes that morphogens give the correct growth direction.

Observing these rules, we notice that it could be possible to produce all desired creatures with the same genome. Indeed, the rules discovered by the organism allow it to follow any morphogen configuration. To verify the hypothesis, we decided to develop another simple creature: a jellyfish. To do that, we keep exactly the same environment architecture, with the same substrates and the same possible actions, and we only change the morphogen distribution in the environment. Using the starfish genome, we launch the simulation and we obtain the creature[1] shown by Figure 8.

Conclusion and future works

We propose a model of cellular development. This model is based on a marked simplification of natural development. We ignore the physics rules and the atomic and molecular interactions to focus on the cell abilities. Using a genetic algorithm and specific environment, we create an organism able to develop different organs with different functions. As we have shown during experiments, this model can produce various creatures with very different morphology or different functions.

The continuation of this work presents a wide field of de-

velopment. Developing new organs can be interested. For example, the next one could be an organ able to harvest different substrates and transform them into vital energy and dispose wastes at a specific position. Using different types of such an organ, the wastes of one used as energetic substrate by another, we will produce a complete creature composed of different organs. The different organs will be connected using the presented transfer system.

Another improvement may concern shape generation. For the moment, we use four different morphogens to obtain the creature morphology. We think that with only one morphogen and only giving the development main line, we could obtain the same creature and have an organ that develops itself correctly to produce this morphogenetic substrate. For example, in the case of the starfish, we could have a transfer system that moves the morphogenetic substrate from the center of the environment to the five branches of the starfish. In a second stage, the starfish will grow using the morphogen distribution.

A remark we can make when we watch the starfish growth is that all the branches do not grow at the same speed. The same fact can be noticed in jellyfish growth, where the bell-shape grows too fast in comparison with tentacle development. An idea to control shape development is to calculate fitness at different moments of the simulation. The best creature will then be the one that produces the best shape at each checkpoint.

After few experimentations, the model also seems to be able of self-repairing (Miller, 2004). Killing some cells of the starfish in different parts (center, middle of an arm or a complete arm), the starfish create new cells in these wholes. This self-repairing property must be confirmed by more experiments but are encouraging.

A final development path is the abstraction of this model. Starting from a unique cell, we grow shapes like the starfish or the jellyfish presented in the paper and, after a cell regroupement to different limbs, we want to put the creature in a physical simulator to make it move. The creature movements could be generated, for example, by a neural network, just like in Sims' works (Sims, 1994). We hope that this abstraction will allow us to have a complete creature development, from single cell to a creature able to move in its environment.

Acknowledgments Experiments presented in this paper were carried out using ProActive, a middleware for parallel, distributed and multi-threaded computing (see http://proactive.inria.fr), and the Grid'5000 French experimental testbed (see https://www.grid5000.fr)

References

Banzhaf, W. (2003). Artificial regulatory networks and genetic programming. *Genetic Programming Theory and Practice*, pages 43–62.

Chavoya, A. and Duthen, Y. (2007). Evolving an artificial regulatory network for 2d cell patterning. *Proceedings of the 2007 IEEE Symposium on Artificial Life*, pages 47–53.

Cussat-Blanc, S., Luga, H., and Duthen, Y. (2007). A developmental model to simulate natural evolution. *International Conference on Computer Graphics and Artificial Intelligence 3IA*.

Davidson, E. H. (2006). The regulatory genome: gene regulatory networks in development and evolution. *Academic Press*.

Dawkins, R. (1986). The blind watchmaker. *Longman Scientific & Technical*.

de Garis, H. (1999). Artificial embryology and cellular differentiation. In Peter J. Bentley, e., editor, *Evolutionary Design by Computers*, pages 281–295.

Dellaert, F. and Beer, R. (1994). Toward an evolvable model of development for autonomous agent synthesis. In *Artificial Life IV*, Cambridge, MA. MIT press.

Dittrich, P., Ziegler, J., and Banzhaf, W. (2001). Artificial chemistries - a review. *Artificial Life*, 7(3):225–275.

Eggenberger Hotz, P. (2004). Asymmetric cell division and its integration with other developmental processes for artificial evolutionary systems. In *Artificial Life IX*, pages 387–392.

Garcia Carbajal, S., Moran, M. B., and Martinez, F. G. (2004). Evolgl: Life in a pond. *Artificial Life XI*, pages 75–80.

Gardner, M. (1970). The fantastic combinations of John Conway's new solitaire game life. *Scientific American*, 223:120–123.

Holland, J. H. and Reitman, J. S. (1978). Cognitive systems based on adaptive algorithms. *Pattern-Directed Inference Systems*.

Hutton, T. J. (2007). Evolvable self-reproducing cells in a two-dimensional artificial chemistry. *Artificial Life*, 13(1):11–30.

Kauffman, S. (1969). Metabolic stability and epigenesis in randomly constructed genetic nets. *Journal of Theorical Biology*, 22:437–467.

Komosinski, M. and Ulatowski, S. (1999). Towards a simulation of a nature-like world creatures and evolution. In *ECAL '99*, pages 261–265, London, UK. Springer-Verlag.

Lassabe, N., Luga, H., and Duthen, Y. (2007). A New Step for Evolving Creatures. In *IEEE-ALife'07*, pages 243–251. IEEE.

Miller, J. F. (2004). Evolving a self-repairing, self-regulating, french flag organism. In *GECCO (1)*, pages 129–139.

Ono, N. and Ikegami, T. (1999). Model of self-replicating cell capable of self-maintenance. In *ECAL '99*, pages 399–406, London, UK. Springer-Verlag.

Rasmussen, S., Chen, L., Nilsson, M., and Abe, S. (2003). Bridging nonliving and living matter. *Artificial Life*, 9(3):269–316.

Sims, K. (1994). Evolving 3d morphology and behavior by competition. *Artificial Life IV*, pages 28–39.

Stewart, F., Taylor, T., and Konidaris, G. (2005). Metamorph: Experimenting with genetic regulatory networks for artificial development. In *ECAL '05*, pages 108–117.

Ventrella, J. (1998). Attractiveness vs efficiency (how mate preference affects location in the evolution of artificial ng organisms). *Artificial Life VI*, pages 178–186.

A Model Chemical Memory in an Evolved Animat

Kyran Dale
Centre for Computational Neuroscience and Robotics (CCNR)
Informatics
University of Sussex Brighton BN1 9QH, UK
kgd20@sussex.ac.uk

Abstract

This paper describes work carried out to investigate whether a classic simulated reaction-diffusion (RD) system could be used to control a 'minimally cognitive' animat during the course of a simple memory test. This test required the animat to remember an arbitrary signal and adapt its behaviour as a result. A further requirement was that the effects of the first signal be reversed by a second signal, returning the animat to its default behaviour. In this way the two signals combined to form a behavioural-switch, regulated by a memory-trace preserved in the homogeneous chemical substrate. The reaction-diffusion system chosen was that first described by Gray and Scott (Gray-Scott) and the minimally cognitive behavior of a class introduced by Beer et. al, involving the fixation and avoidance of a falling circular object by a whiskered animat. The parameters of this RD-controller were evolved using an evolutionary, or genetic, algorithm (GA).

Introduction

The study of memory in simple cognitive models has focussed primarily on the artificial neuron (AN) as a main component. In this paradigm networks of these ANs, connected by excitatory or inhibitory signalling links, mediate behaviour. The AN building blocks are usually heterogeneous, having variables such as activation time-constants or signalling thresholds which are adjusted on an individual basis. This paper deals with a different type of controller, which we call a reaction-diffusion controller (RDC), consisting of a one-dimensional array of cellular automata (CA) implementing a classic chemical model of reaction-diffusion. The constituent cells of this CA are homogeneous, sharing global defining variables.

Within Evolutionary Robotics the prominent model dynamical system is the continuous time recurrent neural network (CTRNN) (8; 9; 4). Many examples testify to the rich dynamics of which CTRNNs are capable (7; 9), such as generating the patterns to regulate legged robot gaits, and controlling such simple cognitive tasks as navigation and shape-discrimination. Many classic reaction-diffusion (RD) systems also display rich dynamics, manifesting the full range of classic qualities such as Hopf bifurcation, stable and unstable limit-cycles, chaotic boundaries etc.. The main motivation for the work described in this paper was to see whether the tried and tested technique of evolving neural-network controllers for simple robotic behavior could be adapted to harnessing some of the rich dynamics displayed by these RD systems. In this sense the interest was both methodological, to show that evolutionary algorithms could be used successfully with a different class of non-linear system, but also focused on exploring the ability of RD controllers. For example the ability to sustain spatio-temporal patterns suggests a role in controlling gaited movement but can systems be tuned to particular requirements? Given the difference between the essentially 'spaceless' CTRNNs and the necessarily spatial RD systems there is also the intriguing possibility that they might be able to complement one another. By placing artificial neurons in an excitable medium with which they can interact, CTRNNs might be able to exploit the spatio-temporal properties of the medium. Given the dynamical potential of these RD systems there has been very little work dedicated to exploring it (1; 2).

In place of the continuous time recurrent neural network used by Beer we used a one-dimensional ring of cells within which the concentration of two coupled chemicals changed according to two differential equations describing intra-cell reactions and inter-cell diffusion Fig. 1. Output from whisker-like proximity sensors was fed to the cells in the RD-ring via weighted links, perturbing the concentration of the two chemicals. Weighted links in turn allowed the concentration of particular chemicals in designated cells to specify motor activation, completing a sensor-motor loop. Links were made symmetrically about the animat's longitudinal axis. Parameters specifying the weighted links between cells, motors and sensors were evolved as were the values of a dimensionless feed rate and rate constant for the RD-system.

Reaction-diffusion Models

Perhaps the best known example of a reaction-diffusion model is that proposed by Alan Turing (6) as an attempt to explain cellular differentiation in early biological development. It is also one of the first examples of the use of a

computer to solve differential equations. Turing was trying to understand how the chemicals in arrays, in this case one-dimensional, of identical cells could, by reacting within the cells and diffusing between them, form stable patterns. He was able to show that by constraining the chemical reactions within cells and the relative rate of diffusion between them one could guarantee a stable pattern. Subsequent work has shown analogous systems responsible for leopards' stripes, patternation of nautilus shells and many other natural patterns.

Within the class of model reaction-diffusion systems defined by two coupled chemicals (two rate equations) Turing was interested in those tending toward a stable configuration. But by altering the governing reactions and diffusion rates many other systems are possible, displaying a wide variety of spatio-temporal properties. One of the most intriguing is that proposed by Gray and Scott in their 1984 paper (10) and extensively analyzed by Pearson in his 1993 paper (5). A variant of the autocatalytic Selkov model of glycolysis (5) the Gray-Scott model corresponds to the following reactions:

$$u + 2v \rightharpoonup 3v \tag{1}$$

$$v \rightharpoonup p \tag{2}$$

Both reactions are reversible so p is an inert product. A feed term for u introduces a non-equilibrium constraint with the feed process removing both u and v. This results in the following reaction-diffusion equations, expressed in dimensionless units:

$$\frac{\partial u}{\partial t}\& = \&d_u \nabla^2 u - uv^2 + F(1-u) \tag{3}$$

$$\frac{\partial v}{\partial t}\& = \&d_v \nabla^2 v + uv^2 - (F+k)v \tag{4}$$

where k is a dimensionless rate constant and F a dimensionless feed constant. d_u and d_v are the diffusion rates for the two chemicals (see Method section below for specific details). A trivial steady state of $u = 1, v = 0$ exists for all values of F and k. Gray-Scott proves a very robust simulation, showing no qualitative difference when implemented by forward Euler integration over a broad range of spatial and temporal scales (5).

When suitably perturbed Gray-Scott exhibits a large variety of spatio-temporal patterns that have to be seen to be appreciated. Pearson's paper is replete with beautiful images but the simulation is best appreciated in real-time with a two-dimensional simulation and a suitable colour-map. By fixing the diffusion rates of the chemicals and using F and k as control parameters Pearson was able to show that within suitable limits the two-dimensional phase-diagram described shows regions associated with specific spatio-temporal patterns, ranging from spot replication and stripes in a continuous transition to traveling waves and spatio-temporal chaos.

Visually-Guided Agents

The choice of an evolved animat model, for example to demonstrate the potential of a novel reaction-diffusion controller, should be informed by two key considerations. The behavior in question must be cognitively 'interesting' and there should be a reasonable expectation that resultant controllers can be analyzed and understood.

> The term 'minimally cognitive behavior' is meant to connote the simplest behavior that raises cognitively interesting issues.
>
> Generally speaking, visually-guided behavior provides an excellent arena in which to explore the cognitive implications of of dynamical and adaptive behavior ideas, since it raises a host of issues of immediate interest. ((8) p.422)

In keeping with Beer's thesis we chose for our memory test a visual-guidance task conforming to the requirements of 'minimal cognition'. After priming by an arbitrary signal **s+** a whiskered animat, capable of moving along the floor of a two-dimensional arena (in the xz plane), is required to orientate toward and track a circular object falling from the arena's ceiling with a large range of vertical and horizontal speeds. In the absence of **s+** or if **s+** is followed by the second signal **s-** the animat is required to avoid the falling object. (see Figs. 2 and 3 for details).

Beer evolved continuous time recurrent neural networks (CTRNNs) to control his animats, a control-system the author has some experience of (4). His subsequent analysis (9) of the CTRNNs' dynamics makes them probably the best understood of all animat controllers, evolved or otherwise. This represents a useful benchmark and an obvious model to emulate. The use of such canonical models to provide a common point of reference would seem to be an efficient way to exploit the resources available. Broadly speaking this work preserves the details of Beer's model while replacing the CTRNN controller with a novel one using a reaction-diffusion medium.

Evolving Controllers

The Gray-Scott model, in keeping with most reaction-diffusion systems, is highly non-linear, at least unintuitive and often counter-intuitive [1]. It is not immediately clear how one could 'hand-wire' such a controller, but it would require an intuition about the rich dynamics of the system which escapes us. In cases such as this, where we require a controller capable of exploiting even a relatively simple dynamical system, it would seem that the need is pressing to leverage the increasing computer power at our disposal and

[1] The speed of modern processors makes it possible to interact in real-time with 2D implementations of these reaction-diffusion systems. Having implemented and played with just such a model of, among others, Gray-Scott, we can attest to its counter-intuitiveness.

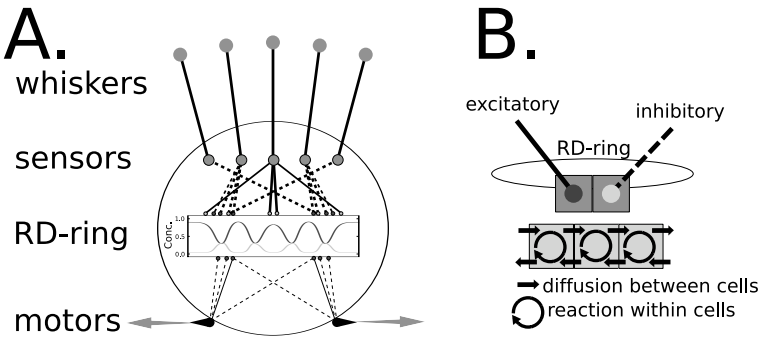


Figure 1: (A) The animat model. Output from the proximity sensors is fed, via weighted links, to the reaction-diffusion ring (*RD-Ring*) where it perturbs the cellular concentration of chemicals u and v. Solid links increase the chemical concentration in the cell while dashed links decrease it. The effects of any particular link are specific to one of the two chemicals u and v, this specificity being under evolutionary control. Following a number of reaction-diffusion cycles, the chemical concentration levels in designated cells are in turn fed via weighted links to activate the animat's motors. Activation at a motors is summed and multiplied by a constant (10) to produce an output. The combined output of oppositional left and right motors is used to move the animat. (B) Excitatory links from the sensors increase chemical concentration in the cell specified while inhibitory links (dashed) decrease it. In this way the whisker sensors affect the chemistry of the *RD-Ring* which in turn affects the motors.

automate the process of discovery. This approach is particularly appropriate to a robot that is intended to remain in-silico. The search algorithm employed here is a genetic algorithm (GA). A simplistic, but initially useful, way of understanding how a GA works is to picture the parameter space, describing in this case the details of our reaction-diffusion controller such as linkage points and weights, as a fitness landscape. Every point in this landscape describes an animat controller and height above ground corresponds to fitness. If the landscape is reasonably well-ordered it should be possible for the GA to find its way from low ground initially, corresponding to randomly-wired, poor performing controllers, to high, where the controllers are (much) better performing. This image leaves out important details, particularly the concept of neutral-networks [2], but the key detail is captured. From random parameters and allowing for a suitable encoding scheme, it should be possible to automatically produce good controllers by applying evolutionary pressure. The work described in this paper and elsewhere (8; 9) is testament to that fact.

Method

To a large extent details from Beer's earlier simulations (8) were preserved and the required behaviors essentially the same. The arena was 400 units long by 275 units high (Fig. 2) in all the experiments. The animat's five whisker sensors were 220 long and uniformly spaced over a 30^o spread. Activation of the whiskers was a simple linear function with a minimal value of 0 when the whisker was unimpinged and 1 when it was intersected at base.

[2] A complex subject highlighting our poor intuition of movement in higher-dimensional space

Fig. 1 shows a diagram of the animat. Activation from the sensors $\in [0, 1]$ was fed through weighted links $\in [-1, 1]$ to the one-dimensional reaction-diffusion ring (RD-ring) consisting of 128 cells subject to intra-cellular reaction and inter-cellular diffusion between near-neighbours (see the chemical reactions 1, 2 and rate equations 3 and 4). The weighted links were specific to either chemical u or v, this specificity being under evolutionary control.

The sensors, motors and input to the RD-Ring were updated using the forward Euler method with an integration step-size of 0.1. During this time-step each cell in the RD-ring was updated twice using the rate equations 3 and 4). Input via links to the cells perturbed the specified chemical's concentration by a simple multiple of time-step (0.1), sensor activation $\in [0, 1]$ and link weight $\in [-1, 1]$. The cellular concentration of u and v was bounded within the range $\in [0, 1]$.

The animat's motors received input from cells in the RD-ring. Input from a individual link was a product of link-weight $\in [-1, 1]$ and the concentration of the evolutionarily specified chemical in the cell. To update the animat's position, the activation of the oppositional motors was subtracted $(right - left)$ and the result multiplied by 10. This multiplier was fairly arbitrary, taking into account the need for the animat to move fast enough to catch objects with a maximal horizontal velocity around 5. It worked well enough but is probably too large. On reflection this value should probably have been an evolutionarily-specified parameter but given the fitness scores generated any gains could only have been very marginal.

Diffusion rates d_u and d_v were fixed at the standard values (5) of 2×10^{-5} and 10^{-5} respectively and the length of the RD-ring was 0.32. Each animat genotype specified

a value for the rate constant k and feed constant F (equations 3 and 4) which were seeded at values 0.055 and 0.02 respectively in the otherwise randomly generated initial populations. By moving through this F, k parameter-space evolution had some control over the properties of the reaction-diffusion system (see subsection *Reaction-diffusion Models* above).

The GA consisted of a population of thirty animat genotypes which were updated generationally according to rank-based selection. The genotypes were essentially a list of weighted, chemically specific links, describing the wiring of an animat controller. As the animat controllers were symmetrical, each link on the list corresponded to two links on the controller. At each generation these lists were converted into their respective animat controllers and assigned a fitness value according how well the controller performed its task. It was neither practical or desirable to have the genotype describe a fully connected controller (1408 links in all) so the number of links was pre-set. The starting number for the orientation experiment was 8 sensor→RD-ring, 4 RD-ring→motor making 24 symmetrically arranged links in all.

At the end of each generation a new generation was formed from the old and subjected to mutation operations. The numbers on the genotype were in the range $\in [0, 1]$, being mapped onto their respective controller parameters. Mutation consisted of the addition of a normally distributed random value with average 0 and standard-deviation 0.25. A second mutation operator was applied to each genotype with a probability of 10%, randomly deleting a link from or adding a link to the list. The link-addition operator allowed two links to share start and end points and chemical specificity.

The same fitness function $f(d_s, d_e)$ was used to evaluate all four trials (see Fig. 3) where the two values ds and de specify the absolute horizontal distance between animat and shape at the trial's start and end, marked by the shape reaching the arena floor, respectively:

$$f(d_s, d_e) = \begin{cases} 1 - \frac{d_s}{d_e} & \text{if } d_e < d_s \\ \max(\frac{d_s - d_e}{50}, -1) & \text{if } d_e \geq d_s \end{cases}$$

The value of this fitness function is highest if the animat fixates the object centrally and lowest if the animat avoids the object, to a maximum at distance 50. Where the trial required the animat to avoid the falling object (T1, T3, T4 in Fig. 3) the resultant fitness was multiplied by -1.0.

To evaluate the animat's performance at the memory task we used an amalgam of the fitnesses $F1 - 4$ over four trials $T1{-}4$. The animat was required to show the opposite behavior over $T2$ to the other trials, dependent on a prior stimulus. The four trials were alloted a fitness for orientation towards the circle and the total fitness calculated thus:

$$f(F1, F2, F3, F4) = (F1 - F2) \times (F3 + F4)$$

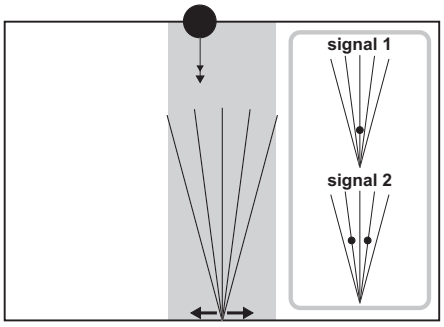


Figure 2: The memory experiment (to scale). An animat with five whiskers spread over a 30^o span is placed at the centre of the arena's floor. During a trial a circle was placed at the top of the arena within the grey drop zone on a straight downward trajectory of between 3 and 4 units per second. Prior to the circle-drop the animat received either signal 1 or signal 2 or signals 1 and 2 consecutively. The signals consisted of an arbitrary pattern applied to the animat's whiskers after which the system was allowed to settle. The animat was rewarded for its ability to reverse behavior on receiving signal 1, for example switching from a circle fixator to a circle avoider.

This function was designed to encourage a switching of behavior over T2 while avoiding evolutionary local minima [3] that might result by a simple adding of $F1 - 4$. The function does not specify whether $T2$ should show avoidance or fixation behavior, only that it is opposite to that seen in the other trials. To disambiguate these two possibilities the amalgamated fitness was multiplied by -1.0 in those cases where the animat showed aversion to the object in trial T2.

Training Protocol

A number of trials were conducted to assess the ability of individual controllers and allot their respective genotypes a fitness score. Fig. 2.a shows the trial set-up for the orientation experiment. The grey drop-zone delimits the possible circle trajectories during a trial. The object trajectories were constrained so as to ensure some whisker stimulation for the animat. [4]

Memory Previous work (3) has shown that motor feedback to the RD-ring can be used by the animat to stabilize behaviour. Purely diffuse controllers were able to use motor feedback in this way to maintain a memory trace. To prevent this the animats were not allowed to form links from motors to RD-ring over the course of evolution. A second consideration is the length of time between the animat receiving one of the priming stimuli **s+** and **s-** and its response

[3] A simple explanation...

[4] In keeping with Beer's original model (8) the simulation was noiseless, meaning that the symmetrical animat controller was incapable of breaking symmetry without stimulus from the whiskers.

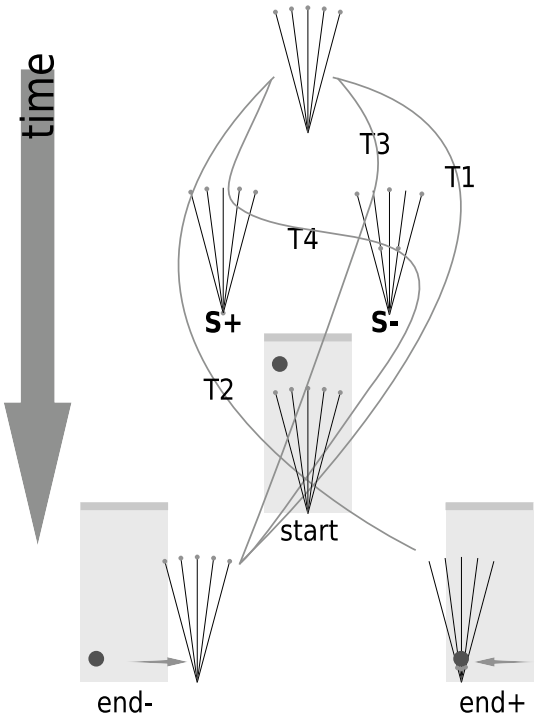


Figure 3: The four trials (**T1-4**) providing the components of the memory task's fitness function. **end+** and **end-** show the desired end positions of the animat relative to the falling object. The duration of all phases is the range $\in [400, 600]$ and the stimuli last for 10 time units. (**T1**) In this trial the animat receives no signal. The falling circle should elicit an aversion response. (**T2**) In this trial the animat receives the **s+** signal. As a consequence the animat should fixate the falling circle. (**T3**) In this trial the animat receives the reset signal **s-**. This should not affect the animat's aversion to a falling circle. (**T4**) In this trial the animat receives two signals, the priming **s+** followed by **s-**. The second **s-** signal should reset the animat's response, causing it to avoid the falling circle.

to the falling object. The limits on this random wait time $\in [400, 600]$ were set to oblige the animat to use the reaction between u and v to sustain a memory .

Fig. 3 shows the four trials T1-4 that comprised a single fitness test:

T1 No signal

T2 signal **s+**

T3 signal **s-**

T4 signals **s+** followed by **s-**

The animat was required to avoid the falling object in all trials but T2 where, after receiving signal **s+** it was required to fixate the circle.

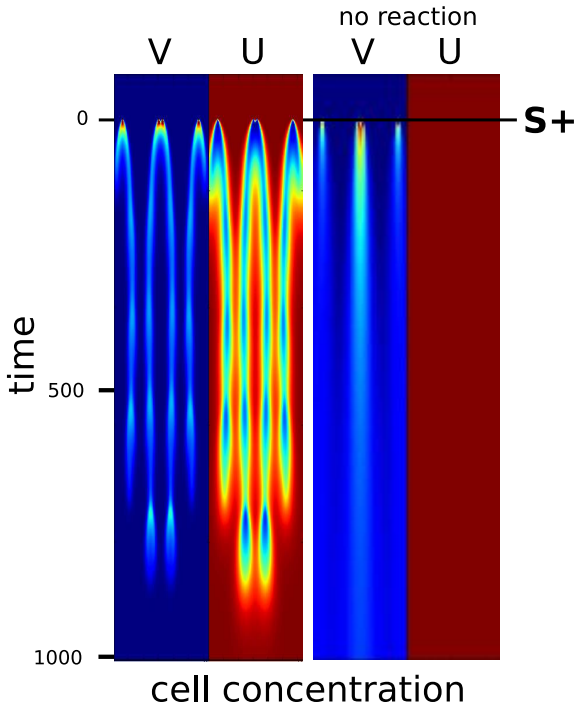


Figure 5: The colour-mapped (red high, blue low) changes in concentration over time of chemicals u and v in response to the signal **s+**. In the trace on the right the reaction component of the RD system has been disabled. When allowed to react the two chemicals maintain a strong, autocatalytic memory trace for approximately 750 time units, long enough to reliably remember the signal and score highly on the task. In the absence of reaction between u and v initially high concentrations of v diffuse away while u, unable to autocatalyse, shows no change in activity; the links to the animat's motors are all u specific so in the absence of reaction the animat is paralysed.

Results

In this section we focus on the best performing animat of a single evolved population. Populations with near-optimal solutions were readily evolved and the choice of this one is arbitrary but informed by pedagogic considerations. Similar mechanisms to those described were found in the large majority of those animats analyzed.

An important consideration in this kind of evolutionary modelling is that one be hopeful that the models produced will submit to analysis. As mentioned above, evolving a fully connected RD-controller would be infeasible, given the processing power available, but another objection might be that by encouraging evolution to distribute the animat's cognition within an overly complex structure one ends up with a model which is too complex to understand. One of the benefits of allowing network connectivity to be an evolutionary variable is that we can encourage evolution to search

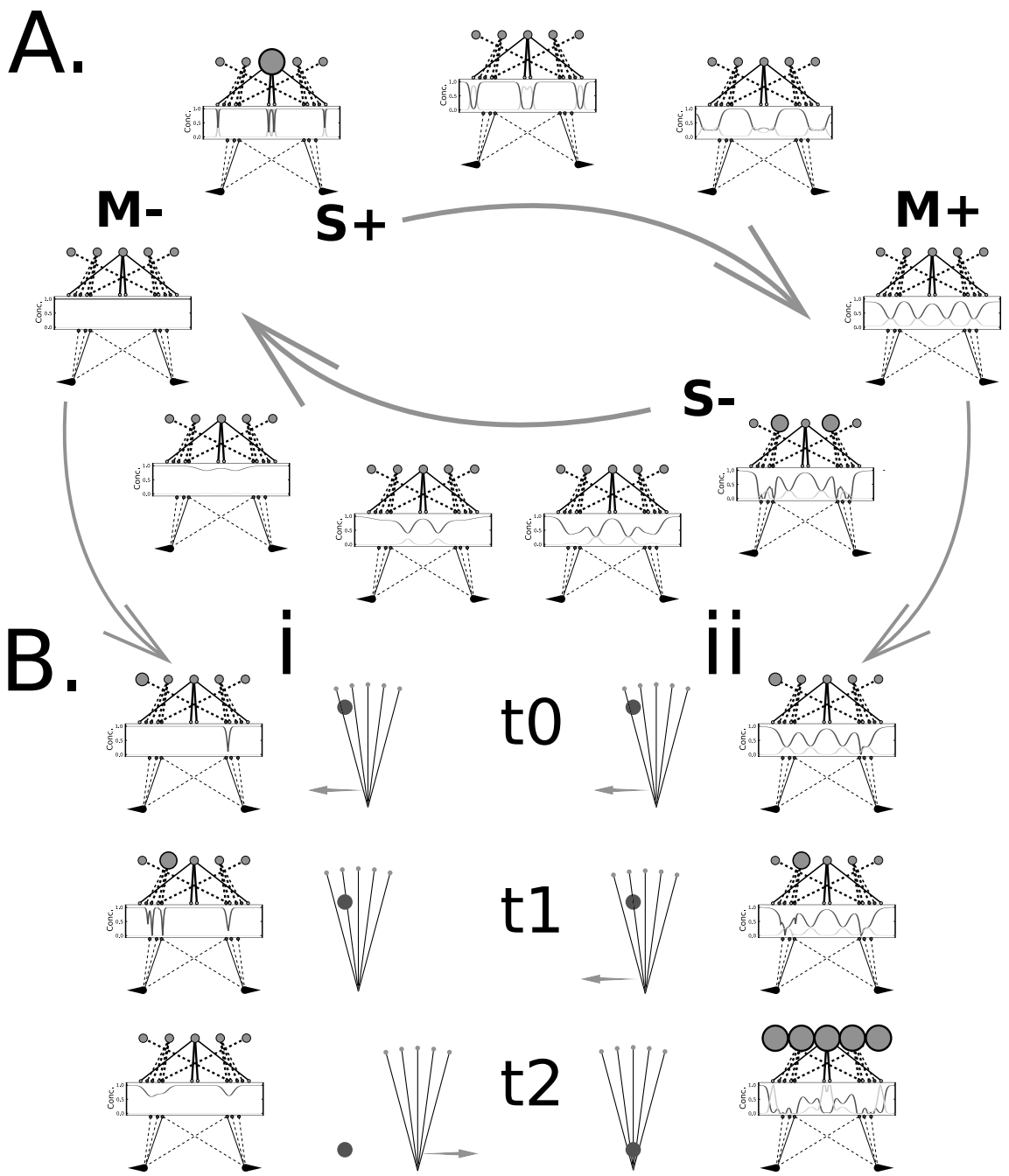


Figure 4: (A.) A chemical memory-switch. Moving clockwise from the default settled state **M-**, the animat receives the stimulus **S+** causing an auto-catalytic cycle which leads to the semi-stable state **M+**. While in state **M+** the resetting signal **S-** disrupts the RD-ring's structure, causing it to return to **M-**. (B.) shows the response of the animat, for states **M+** and **M-**, to a falling object. (i) In the default settled state **M-** the animat displays aversion to the falling circle. (ii) In the stimulated state **M+** the same falling circle elicits a fixation response.

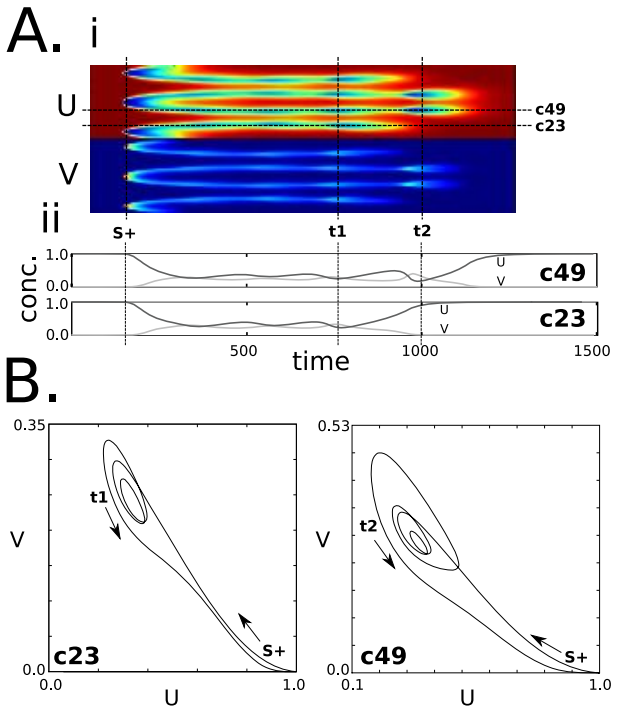


Figure 6: The change in U/V concentration over time of the two RD-ring cells **c23** and **c49** in response to the animat receiving stimulus **s+**. Two other significant times are marked, **t1** and **t2**, the return of **c23** and **c49** to default. (Ai) shows the colourmapped (red high, blue low) concentrations of chemicals U and V, with the position of cells **c23** and **c49** indicated, for U, by dashed horizontal lines. (Aii) the change in concentration over time for chemicals U (dark grey) and V (light grey) in cells **c23** and **c49**. (B) Plotting the activation of the two chemicals U and V against each other reveals the unstable attractor cycles at **c23** and **c49** that characterize the memory trace. The orbits, though unstable, are maintained throughout the course of the trial, allowing the stimulated animat to respond differently to the falling object.

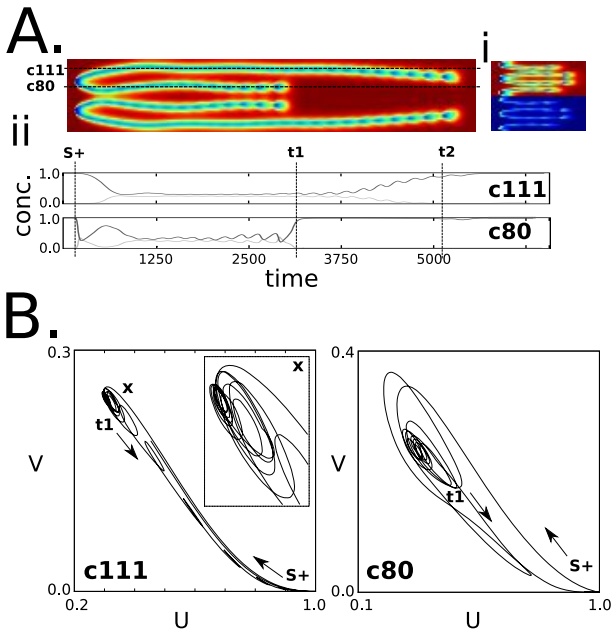


Figure 7: The reaction of two selected cells, **c80** and **c111**, to stimulus **s+** after a further period of evolution, wherein the animat was required to increase the length of its memory. The top plot of (Ai) shows, to scale, the previous memory trace following **s+**. (B) the attractor cycles are now more densely packed and greater in number.

for simpler solutions. With this in mind, the animat population of these results, having achieved a close to optimal fitness, was further evolved with a single change made to the GA. The probability of adding a link to the controller during mutation was set to zero, while the probability of deleting a link remained the same. In this way evolution is 'locked' from exploring bigger networks while being able to randomly wander through or test smaller ones. This technique has been found to rapidly reduce the size of networks while maintaining their fitness and has the added advantage that one does not have to introduce an arbitrary component into the fitness function to encourage simplicity.

A Chemical Switch

The evolved model, with 22 symmetrically arranged links, is summarized in Fig. 4. Fig. 4A shows the switching cycle which allows the animat to flip from state **M-** to state **M+** and back again in response to signals **s+** and **s-**. Clockwise from **M- s+**, a maximum stimulus of the central whisker, increases, via v-specific positive links, the concentration of chemical v at four cells in the RD-ring. This establishes autocatalytic waves, as u and v react and diffuse, which roughly stabilize at **M+**. The application of **s-**, a half maximum stimulus of the first proximal whiskers, at **M+** disrupts these waves, bringing the chemical system back to **M-**.

Fig. 4B shows the response of the animat to the same falling object trajectory whilst in default state **M-**(i) and primed state **M+**(ii). In **M-** stimulation of the leftmost whisker reduces the concentration of v via an inhibitory connection. Motor links, sensitive to changes in v's concentration, imbalance the left and right motors, drawing the animat towards the falling object. This behaviour at t0 is mirrored in **M+** (Bii). At time t1 the behaviours start to diverge, in response to stimulation of the whisker second from left. In Bi this engages a strong avoidance response in the animat, causing it to move quickly away from the object. The animat primed by **s+** does the opposite, moving the object towards its centre where, thus fixated, it remains through the course of the trial.

The dependence of the memory state **M+**, induced by signal **s+**, on interaction between chemicals u and v is highlighted in Fig. 5. In the absence of a reaction component the memory is not established. Purely diffusive controllers were thus unable to evolve a solution to this task.

The Stability of the Chemical Memory

As shown in Figs. 5 and 6 the memory trace produced by signal **s+** is unstable. The memory is long enough to enable the animat to pass the test requirements, being roughly defined by the upper bound of the randomized time between the application of a stimulus and the first engagement of the falling circle with the animat's whisker. In Fig. 6 we focus on the change in concentration of two cells over the course of the memory trace. These cells are roughly at the centre of two of the four symmetrical the peaks and troughs of v and u respectively (see Fig.4 **M+**). Fig. 6B shows that at these points in the chemical ring the interaction of u and v describe orbits around an area of the phase-space. At time **t1** this orbit cannot be sustained and the cell **c23** returns to default. The destruction of the **c23**'s orbit presages the destruction of the longer cycle of **c49** at time **t2**. The dependence of **c49**'s cycle on **c23**'s emphasizes the global nature of this memory.

Extending the Memory's Duration

In order to see whether the duration of the animat's memory-trace was coincidental to the requirements of the task, the animat population was further evolved (see subsection *Evolving Controllers* above), under standard conditions, while the time between the application of stimuli and the dropping of the object was gradually increased, requiring the animat to maintain a longer memory of the stimuli. Fig. 7 shows the successful result of an evolutionary run which required the animat to maintain a memory five times longer than that of the original task, whose trace is shown to scale in Ai. At this point the simulations became impracticably long but there was no indication in this or other of a hard constraint on the possible duration of memory. Note that the interaction of u and v now describe many more and tighter orbits in phase-space, maintaining the memory-trace for much longer. It should be stressed here that the necessary limits to spatial and temporal resolution in the simulated chemical RD-Ring probably play a part in the precise characteristics of the memory trace and that these results should be interpreted qualitatively.

Discussion

Although Cajal's neuron doctrine is predominant in cognitive studies and, by definition, neuroscience computational and otherwise, it does beg a very big question. How does a single-celled animal, of a set representing the larger biomass of the animal kingdom and those evolutionary precursors of all multi-cellular lifeforms including ourselves, negotiate its world and engage cognitively with it? Any explanation cannot involve neurons, single cells in themselves, but must explain how a seemingly homogeneous blob of chemicals can produce robust behaviour and exhibit the classical learning models. We would not suggest the models described in this paper hold any answers to the larger questions of animal cognition but the ability of these simple chemical systems to mediate simple cognitive tasks required memory is intriguing. Extending these model systems, used extensively and successfully in Biology to explain such phenomena as cardiac rhythmia, animal patternation, morphogenetical development etc. to the cognitive realm could prove fruitful.

Acknowledgments

Particular thanks must go to Phil Husbands for greenlighting the direction taken in this work, the paper's referees and to the EPSRC (grant GR/T11043/01).

References

[1] Breyer, J., Ackerman, J., McCaskill, J.: Evolving Reaction-diffusion Ecosystems with Self-assembling Structures in Thin Films. Alife 4: 25–40. (1998)

[2] Adamatzky, A., De Lacy Costello,B., Asai, T.: Reaction-diffusion Computers. Elsevier, (2006).

[3] Dale, K.: Evolving Reaction-diffusion Controllers for Minimally Cognitive Animats. Proceedings of Simulation of Adaptive Bahaviour 2006 498-509.

[4] Dale, K., Collett, T.S.: Using Artificial Evolution and Selection to Model Insect Navigation. Current Biology 11: 51–62. (2001)

[5] Pearson, John E.: Complex Patterns in a Simple System. eprint arXiv:patt-sol/9304003 (04/1993)

[6] Turing, A.M.: The Chemical Basis of Morphogenesis. Philosophical Transactions of the Royal Society of London. Series B, Biological Sciences, Volume 237, Issue 641, pp. 37–72 (1952)

[7] Beer, R.D.: On the Dynamics of Small Continuous-time Recurrent Nerual-networks. Adaptive Behaviour 3:469–510

[8] Beer, R.D.: Toward the Evolution of Dynamical Neural Networks for Minimally Cognitive Behavior In P. Maes, M. Mataric, J. Meyer, J. Pollack and S. Wilson (Eds.), From animals to animats 4: Proceedings of the Fourth International Conference on Simulation of Adaptive Behavior (pp. 421–429). MIT Press.

[9] Beer, R.D.: The Dynamics of Active Categorical Perception in an Evolved Model Agent. Adaptive Behavior 11(4):209–243

[10] Gray, P., Scott, S.K.: Autocatalytic Reactions in the Isothermal Continuous Stirred Tank Reactor Oscillations and Instabilities in the System $A + 2B- > 3B, B- > C$. Chem. Eng. Sci., **39**, 1087–1097 (1984)

Protein folding with stochastic L-systems

Gemma Danks[1], Susan Stepney[1] and Leo Caves[1]

[1]University of York, YO10 5DD, UK
gbd501@york.ac.uk

Abstract

Protein molecules adopt a specific global 3D structure in order to carry out their biological function. To achieve this *native* state a newly formed protein molecule has to fold. The folding process and the final fold are both determined by the sequence of amino acids making up the protein chain. It is not currently possible to predict the conformation of the *native* state from the amino acid sequence alone and the protein folding process is still not fully understood. We are using L-systems, sets of rewriting rules, to model the folding of *protein-like* structures. Models of protein folding vary in complexity and the amount of prior knowledge they contain on existing native protein structures. In a previous paper we presented a method of using open L-systems to model the folding of *protein-like* structures using physics-based rewriting rules. Here we present an L-systems model of protein folding that uses knowledge-based rewriting rules and stochastic L-systems.

Introduction

Protein molecules perform molecular functions in the cell that require a specific 3D structure. This *native* state of a protein is achieved only after a process of folding from an initially unfolded state that it adopts during synthesis on ribosomes in the cell. The thermodynamic hypothesis states that a protein folds to its lowest energy state (Anfinsen, 1973). The folding pathway(s) of a protein are unclear but it is known that the only information necessary to predict the native 3D structure of a protein is contained in its amino acid sequence. A protein cannot find its native state through random sampling as even for a small protein this would take in excess of 10^{27} years (Levinthal, 1969; Zwanzig et al., 1992). The energy landscape theory of protein folding (Onuchic et al., 1997) predicts a rugged funnel-like energy landscape biassed towards the native structure due to the effects of evolution. This theory predicts multiple pathways to the native state that an ensemble of unfolded protein molecules may follow. This is an opposing view to the classical view that there is a single defined pathway for each protein proceeding through a sequence of intermediate states.

Protein molecules are possibly the simplest example of a biological complex system and exhibit many emergent properties that have been selected for during the evolution of life on Earth. Proteins are composed of, and function at, many different levels. The folding of a protein may be viewed as an emergent phenomenon. It is governed by underlying physics involved in the interaction of amino acids that make up the protein chain. These local interactions together give rise to the changing conformation of the whole molecule in a way that leads to the native state. We use L-systems (Prusinkiewicz and Lindenmayer, 1990) to represent these local interactions as a set of rewriting rules. In a previous paper we described an open L-systems model of folding *protein-like* structures using simple physics-based rules (Danks et al., 2007). Here we describe a complementary approach using a stochastic L-systems model of protein folding with knowledge-based rewriting rules. We first give an overview of protein structure, then briefly describe the main aspects of modelling protein folding. We give an overview of L-systems and how we previously used them to model protein folding using physics-based rules. We then describe the development of a knowledge-based L-systems model of protein folding and our initial results.

Protein structure

There are 20 different naturally occurring amino acid monomers that make up proteins. These have the same NH_2-CαH-COOH backbone but differ in their side chain from the central carbon atom (Cα). These different side chains give amino acids different chemical properties. The genetic code specifies a unique linear sequence of amino acids that are covalently linked by peptide bonds during protein synthesis to form polypeptides. This is the *primary* structure of a protein. The length of a single polypeptide varies from around 70 amino acid residues to 1000s of residues. The conformation of the polypeptide chain is defined by the local conformation of each amino acid. Peptide bonds that link amino acids together are fairly rigid. This causes the CO of one amino acid and the NH of the next to lie in the same plane. The two backbone bonds N-Cα and Cα-C allow rotation - these rotations give each amino acid its backbone torsion angles ϕ and ψ respectively. These torsion angles

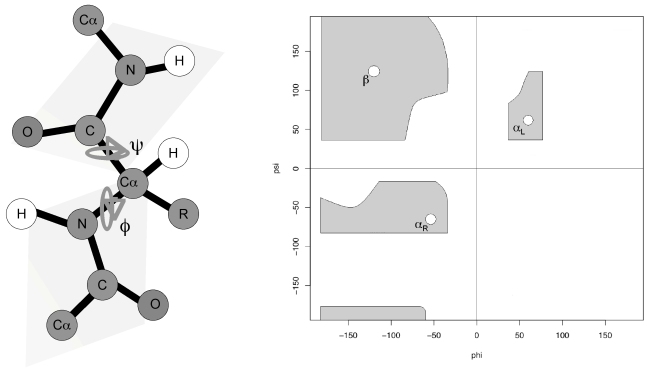


Figure 1: A dipeptide unit showing the structure of an amino acid (NH-CαHR-CO, where R is the amino acid specific side chain). Shaded areas show the atoms lying in the plane of the peptide bonds. The location of the two backbone torsion angles ϕ and ψ are shown and the distribution of sterically allowed values are shown as shaded regions in the ϕ/ψ plot. The two main areas of allowed torsion angles correspond to those that consecutive amino acids adopt to form the two main secondary structure units in folded proteins: the α-helix and β-sheet.

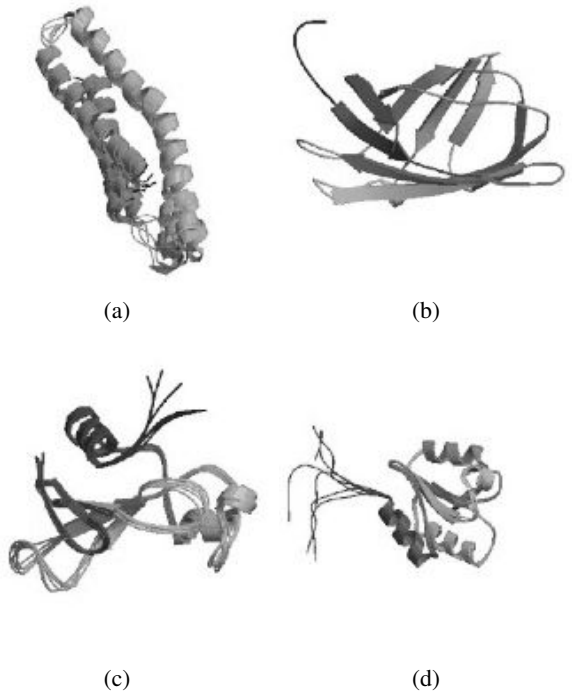


Figure 2: Examples of the four main SCOP classes. All images have been taken from www.rcsb.org. Structures are drawn using ribbons to represent secondary structure (arrows show the direction of a β-strand within a β-sheet). Multiple strands show different experimentally determined structures. (a) An all-alpha protein, PDB ID: 1aj3 (b) An all-beta protein, PDB ID: 1exg (c) Barnase (1bnr) an alpha and beta (a+b) protein - α-helices and β sheets are separated in the protein. (d) 2bjx, an alpha and beta (a/b) protein - α-helices and β-sheets are dispersed throughout the protein.

cannot adopt all possible values due to steric hindrance - some of the atoms branching from the backbone as well as the side chain atoms would collide if certain torsion angles were adopted (Ramachandran et al., 1963). The allowed torsion angles can be plotted to show the regions of ϕ/ψ space that each amino acid can occupy (figure 1). Two main regions of this ϕ/ψ, or Ramachandran, plot of torsion angles at the local amino acid level also correspond to the two main *secondary* structural elements found in native protein structures - the α-helix and the β-sheet. These are formed when a number of consecutive amino acid residues adopt the same torsion angles, and are stabilised by hydrogen bonding between the backbone N-H of one amino acid and the backbone C-O of another. The arrangement of these structural units gives the *tertiary* structure of a protein, i.e. the *native* state of a single polypeptide. The units are generally connected by turns and loops, smaller structural elements.

There are currently over 49,000 known protein structures. These can be divided into classes based on the arrangement and proportion of α-helix and β-sheet units (Murzin et al., 1995; Orengo et al., 1997). Two of the classes are 'all-alpha' and 'all-beta' containing mainly α-helices and β-sheets respectively. Two other major classes in the SCOP database (Murzin et al., 1995) contain a mix of α-helices and β-sheets: the α and β ($\alpha+\beta$) proteins contain segregated alpha and beta regions; the α and β (α/β) proteins contain alternating alpha and beta structures (figure 2). These classes are further subdivided into structurally related proteins. Proteins with similar structures often share a common evolutionary origin and will have a similar amino acid sequence. Comparative modelling uses related sequences with known structures to predict the fold of a new sequence (Ginalski, 2006). However some very different protein sequences can fold to similar structures and occasionally similar sequences fold to different structures.

Modelling protein folding

There are a wide number of existing models of protein folding (Duan and Kollman, 2001). These range in their representation of space (e.g. lattice or off-lattice, 2D or 3D) as well as the level of detail in the protein molecule itself (from all-atom models to those representing each amino acid as a single bead), which also largely defines the representation of interactions within the protein. Models also differ in their assessment of the *protein-like* nature of the final fold and the method used to sample conformations and find the native state. The simplest models can sample every possible conformation to find the most *native-like* state - usually the

lowest energy state where the free energy of the model is represented by the sum of interactions. For example the HP-model (Lau and Dill, 1989; Dill et al., 1995) represents the amino acids in a protein as two different kinds of beads on a string - H and P for hydrophobic and hydrophillic - confined to a 2D lattice with each bead on a point in a grid. The interactions between two H beads (i.e. two H beads next to each other on the grid but not in the string) are favourable and are summed for each conformation to give the energy. With proteins of a small number of beads it is possible to calculate the energy for every possible conformation and find the arrangement on the lattice of the native state. With a more detailed representation this is impossible and a sampling method must be adopted. Two main methods are used in these more detailed models. Monte carlo techniques, based on small random changes in conformation combined with an acceptance criterion using a Boltzman distribution, are widely used to find a conformations of progressively lower energy states (Hansmann and Okamoto, 1999). Molecular dynamics is also used extensively to model protein folding using Newton's laws of motion (Scheraga et al., 2007). However, calculating the forces between all atoms in a protein is computationally intensive and so it is not currently possible to model folding on biological time scales for any but the smallest and fastest folding proteins.

Alternatively, a move-set can be biased using knowledge of native protein structures. The most successful methods of protein structure prediction are those based on fragment assembly (Bujnicki, 2006). These model folding by alternating local conformations of the protein chain between different conformations of short fragments of native protein structures. The ϕ/ψ plot gives the conformations that a single dipeptide unit is allowed to adopt, sterically, which differs slightly between amino acid types. This places restrictions on many of the possible local conformations. However, the choice between allowed conformations can not be readily determined from such a local level. Further restrictions on local conformations are governed by the neighbouring residues and their local conformations (Fitzkee et al., 2005).

L-systems

L-systems were developed as a mathematical theory of plant development (Prusinkiewicz and Lindenmayer, 1990; Lindenmayer, 1968). The simplest L-system consists of an *axiom* containing an initial string of symbols together with a set of rewriting rules, or *productions*, one for each symbol. These rules are applied in parallel to each symbol in the string over a number of *derivation steps*. The current symbol, or *predecessor*, is rewritten by another symbol, or string, the *successor* as defined by the rule for that symbol. For example a simple rule might be:

$$a \rightarrow ab$$

This rule would be applied to every a that appears in the string.

Context sensitive L-system rules are applied only if the symbol is preceeded by and/or followed by a specific string. For example the rule:

$$c < a > d \rightarrow ab$$

is only applied to a if it is preceeded by c in its left context and followed by d in its right context.

Parametric L-systems allow each symbol to have one or more parameters associated with it. The rules can then incorporate conditions on these parameters. For example:

$$a(x) : x > 1 \rightarrow a(2)b(1)$$

will be applied only if the parameter associated with a is greater than 1.

Stochastic L-systems allow a number of different rules to match a certain predecessor. Each rule is applied with a given probability. For example using the rules:

$$a \rightarrow ab : 0.75$$
$$a \rightarrow b \ : 0.25$$

the predecessor a will be replaced by ab 75% of the time and by b 25% of the time.

Open L-systems (Mech and Prusinkiewicz, 1996) incorporate an interacting model of the environment. The L-system and an environmental program communicate using environmental query modules, $?E(...)$. Information is sent to the environment using the parameters of $?E(...)$. The environment uses this information to determine a response and communicates this information back to the L-system using $?E(...)$ parameters, which can be used in productions.

Using L-systems to model protein folding

The backbone conformation of a protein can be described using only the backbone torsion angles (ϕ, ψ) of each amino acid in the chain. The native state of a protein molecule has specific torsion angles associated with each residue. Secondary structure assignment is largely determined by torsion angles together with hydrogen bonding patterns. Folding of the protein involves the torsion angles within each residue changing to their native conformations. L-systems provide a natural way to model this process. Rewriting rules can be used to alter the ϕ, ψ angles in each residue in parallel across the whole molecule. This leads to the emergence of a global 3D fold as a result of local changes in conformation.

In a previous paper (Danks et al., 2007) we described the development of an open L-systems model of protein folding using physics-based rules. A brief outline is given below.

The axiom contains an amino acid sequence, using the single letter amino acid code, with initial backbone

torsion angles, ϕ and ψ, as parameters. For example, the first 4 amino acids in the protein barnase in a β-strand conformation (where ϕ is approximately $-120°$ and ψ is approximately $120°$) gives the following axiom

$$A(-120,120)Q(-120,120)V(-120,120)I(-120,120)$$

An initial derivation step is used to rewrite each symbol representing an amino acid with symbols that represent individual atoms, bonds, bond angles and torsion angles. An initial local conformation of each amino acid is formed by using the initial backbone torsion angles contained in the axiom. Each atom is associated with an environmental query module containing information on the atom that is communicated to an environmental program. At each subsequent folding derivation step an environmental step is performed where the L-system sends this information and the position of each atom in the protein to the environmental program. The environment processes this information and sends a response for each atom back to the environmental query modules in the L-system.

Two models were developed that use a different level of representation of interactions between atoms. One model calculates whether any of the atoms are colliding and another more detailed model calculates the forces between nearby atoms. This information is returned to the L-system. A rule set uses the collision or force information returned to each atom. The rules alter the backbone torsion angles of each amino acid depending on the interactions of atoms within that amino acid with any other atom in the protein that is spatially local. This is repeated over a number of derivation steps leading to a physics-based folding at the global level of the whole protein molecule. The resulting structures at each step were assessed for *protein-like* qualities that included a measure of compactness, which is characteristic of folded protein structures. We found that using local rules in this way to model folding, while not giving *native-like* folds, did lead to compact structures.

Developing a knowledge-based model of protein folding using stochastic L-systems

The physics-based L-systems models allow a protein to sample conformations by moving through time: forces between atoms determine the next conformation. We have used a different approach in developing a knowledge-based L-systems model. A protein alters in conformation over a number of derivation steps, but this is not representative of time. Instead of local moves based on physical forces, local conformations sample those that are most often found in native, i.e. fully folded, structures.

The backbone torsion angles that describe the conformation of a protein are used to assign secondary structure. Taking into account hydrogen bonding and the state of neighbouring residues each residue can be assigned one of seven different secondary structure states (Kabsch and Sander, 1983). These are: α-helix (H), extended strand (E), residue in isolated β-bridge (B), 3/10 helix (G), π-helix (I), hydrogen bonded turn (T) and bend (S). Residues not taking part in secondary structure units are not assigned a state. We have developed an L-systems model that uses these secondary structure states instead of individual torsion angles. We use stochastic rules with probabilities based on data obtained from the DSSP database (ftp://ftp.cmbi.kun.nl/pub/molbio/data/dssp) instead of using physics-based deterministic rules.

Obtaining frequencies of context dependent states

Data obtained from the DSSP database include backbone torsion angles, amino acid type and secondary structure state of each amino acid residue in 35,492 proteins from the protein data bank (www.rcsb.org). Each protein sequence was split into fragments using a window of 3 residues long. Fragments where secondary structure was not assigned were removed leaving 10,954,172 fragments.

Frequencies of each of the 20 residue types in each of the 7 secondary structure states were calculated in all possible contexts of one residue either side. Where R represents an individual amino acid residue, A is the amino acid type and S is its state, a 3-residue fragment contains the following information:

$$R_{i-1}(A_{i-1},S_{i-1})R_i(A_i,S_i)R_{i+1}(A_{i+1},S_{i+1})$$

For each unique combination of $A_{i-1}, S_{i-1}, A_i, A_{i+1}, S_{i+1}$ the frequency of each possible S_i is calculated. There are 20^3 possible 3 residue sequences and 7^3 possible state contexts. All possible 3 residue sequences (8000) appear in the data used here. However, of a possible 2,744,000 unique 3 residue sequence and state combinations only 230,250 appear in the data. Where there is no data for an amino acid in a particular 3 residue fragment in a specific conformation, that state is allocated a low frequency of 10^{-2}, rather than zero, to allow these states to be sampled with a low probability in the L-systems model.

Developing stochastic L-systems rules

An L-systems model has been developed to use the frequencies calculated from the data in stochastic rewriting rules. The axiom contains an amino acid sequence using the single letter amino acid code. An initial derivation step rewrites this code to replace each amino acid by the symbol R with parameters defining the amino acid type and its initial state. For example the first five amino acids in barnase, $AQVIN$, in an initial extended (E) conformation are replaced by:

$$R(A,E)R(Q,E)R(V,E)R(I,E)R(N,E)$$

where the first parameter represents the amino acid type and

the second parameter represents the initial conformation (numbers are used in the model).

Each R is also accompanied by an environmental query module containing the same information. At each subsequent derivation step the information on each residue is first sent to an environmental program. This stores all the residue amino acid types and states. Open L-systems are used here only to store and return specific frequencies from a matrix of values. For each residue, excluding the first and last, given the amino acid type of that residue and the amino acid type and state of one residue either side the frequency of that residue in each of the 7 secondary structure states is found. The first and last residues are given equal probabilities for each state. These 7 frequencies are returned, for each residue, to the environmental query modules in the L-system. A set of 7 stochastic rewriting rules, one for each state, use the corresponding frequency from the environment as its probability of being applied. These rules then rewrite the secondary structure state of each residue depending on the 3 residue sequence that it is within (constant) and the secondary structure state of the residues either side (variable). The form of the rewriting rules are as follows:

$$R(a,s) >?E(p0,p1,p2,p3,p4,p5,p6) \rightarrow R(a,E) : p0$$
$$R(a,s) >?E(p0,p1,p2,p3,p4,p5,p6) \rightarrow R(a,H) : p1$$
$$:$$
$$R(a,s) >?E(p0,p1,p2,p3,p4,p5,p6) \rightarrow R(a,S) : p6$$

where $p0, p1, p2, p3, p4, p5, p6$ are the probabilities of being in states E, H, G, I, B, T, S respectively. Each $R(a,s)$ is followed by an associated environmental query module $?E(p0,p1,p2,p3,p4,p5,p6)$ in its right context. This contains frequencies of each of the 7 secondary structure states, returned from the environment, for that residue while its neighbours are in their current states. Each $R(a,s)$ is then rewritten to change its state, s, to one of the secondary structure states with probabilities calculated from the frequencies in $?E(...)$. The environmental modules are also rewritten to again store the amino acid residue type and the updated state of the preceding $R(a,s)$ to send to the environment at the next derivation step. The states of the neighbours also change at each derivation step as it is a parallel rewriting process.

The aim of this model is to detect the emergence of any locally encoded secondary structure preference and to assess its ability to produce *protein-like* global features. The 3D protein structure is obtained by using *homomorphism* rules. These are applied after each derivation step but are used only for graphical interpretation and do not rewrite any symbols in the string. A rule for each amino acid type draws out the structure of that amino acid with amino acid specific ϕ, ψ angles for each of the seven secondary structure states. These angles were obtained from the data used to calculate the probabilities. Each secondary structure occupies a specific region(s) of the ϕ/ψ plot. As an approximation we took the most common ϕ, ψ angles for each residue type in each state.

Folding proteins using stochastic L-systems

The folding behaviour of four example amino acid sequences using the knowledge-based stochastic L-systems rules are shown in figure 3. Each sequence represents a protein from one of the four major SCOP classes: all-α, all-β, $\alpha + \beta$ and α/β. Each plot shows the change in state of each residue in the protein over 5000 derivation steps. Each protein starts in the same all-extended state. There is a marked difference in patterns of secondary structure, across all derivation steps, between different protein sequences. However, comparison to the native secondary structure states for each protein shows that the structures emerging are not necessarily *native-like*. The horizontal bands that are visible for some residues show that some local secondary structure preference is emerging using these local rules.

Secondary structure is one characteristic of protein structures. Another key feature of globular proteins is their compactness. The 3D structures of each protein at each derivation step was obtained by mapping secondary structure states for each residue type to typical ϕ, ψ torsion angles taken from the data. The radius of gyration (Rg) is a measure of compactness and this was calculated for each structure resulting from each derivation step. Figure 4 shows the change in Rg of one protein, barnase (1bnr), over 2000 derivation steps. This gives an indication of how *protein-like* the global structures are at each step in the L-systems model. The results of the physics-based L-systems model for barnase as well as the value of the native state are also shown. It is clear that the knowledge-based rules are not folding the protein to a very compact structure, and there seems to be little convergence to one structure over time. At most steps the radius of gyration is above the native state and consecutive steps may allow the protein to fold and unfold rapidly. This can also be seen by looking at the global conformations at a number of derivation steps (figure 5).

The physics-based rules seem to be forming more compact structures. There is no constraint on which states a residue may take at the next step in the knowledge-based model other than the probability of being in that state in the context of its neighbours. Torsion angles at subsequent steps could jump dramatically across ϕ/ψ space and this is causing the global structure to also change dramatically. There is also little convergence to a preferred global structure, although this appears to vary between protein sequences - those with more β-sheet conformations seem to maintain a more consistent pattern in the state images (figure 3). This problem is largely due to the fixed probabilities that drive the rules. For convergence to a preferred structure the prob-

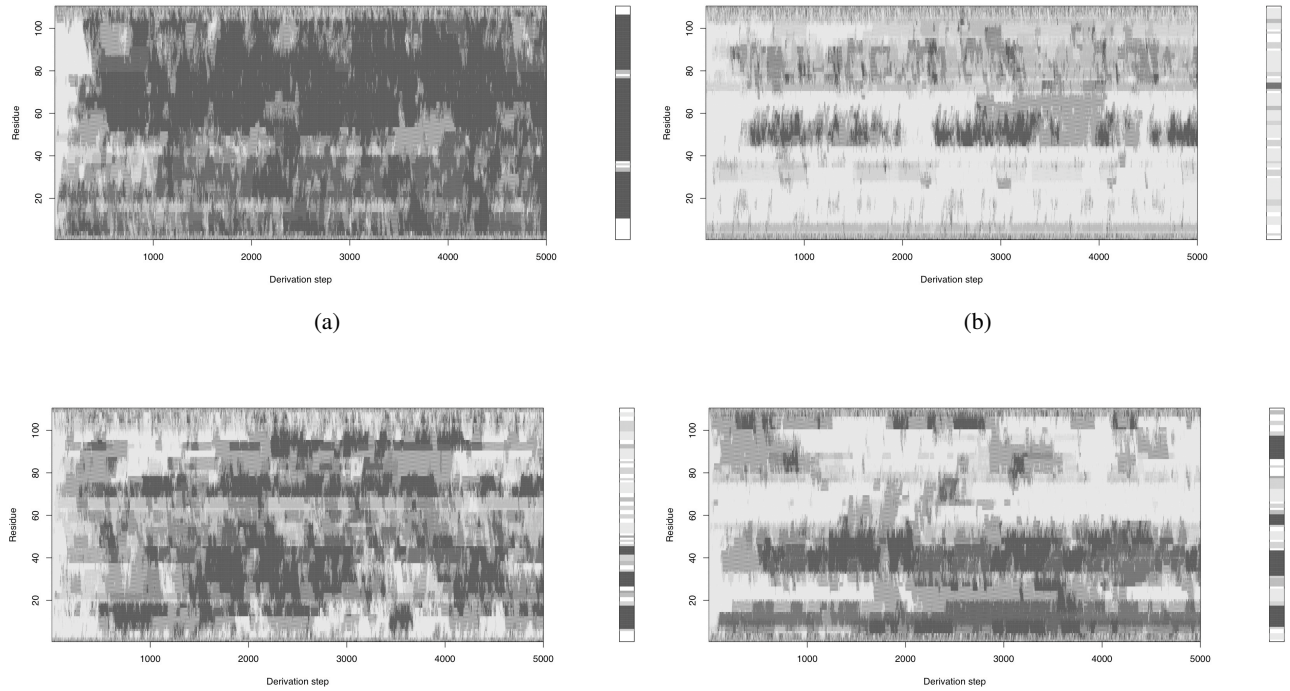


Figure 3: Results from four protein sequences, each from a different SCOP class, using the knowledge-based stochastic L-system rules for 5000 derivation steps. Each image shows the states of individual residues (y-axis) at each derivation step (x-axis). Lightest grey represents the extended state, black represents the α-helix. The 3/10 helix, π-helix, isolated beta bridge, turn and bend are shown in shades of grey from dark to light. Horizontal bands show the emergence of preferred local secondary structure. A bar to the right of each plot shows the native secondary structure for each protein (white represents unassigned states). Native global structures are shown in figure 2. (a) 1aj3 (all-alpha) (b) 1exg (all-beta) (c) 1bnr (α + β) (d) 2bjx (α/β).

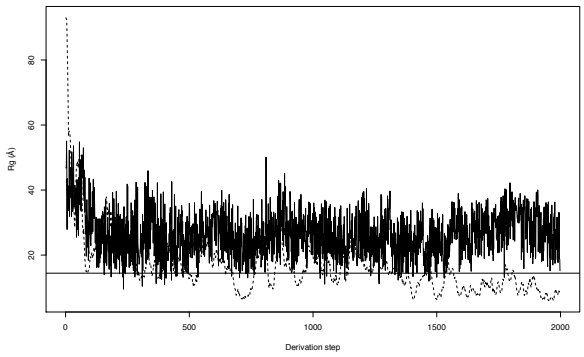


Figure 4: The radius of gyration, Rg - a measure of compactness, for each structure at each derivation step. The solid line shows the change in Rg in the knowledge-based model while the dashed line corresponds to the physics-based model for the the amino acid sequence of barnase, 1bnr. The horizontal line shows the Rg value of the native state.

abilities must be altered during folding to give each residue a final probability of being in only one state. Although not converging to one preferred structure each protein sequence seems to maintain a consistent cycling through similar states. Horizontal bands emerge for certain residues in the states images (figure 3) indicating that there is some secondary structure preference locally in the sequence. Each protein sequence also tends to adopt its particular pattern of states with different initial conformations (figure 6).

A difficulty with assessing global conformations in the knowledge-based model is the inaccuracies in mapping from individual residue secondary structure states to backbone torsion angles. This is particularly difficult when dealing with turns and bends where more than one region of torsion angle space appears in the data. The local conformations of residues that form a turn are dependent on their positions in that turn structure. This issue may be resolved by incorporating context dependence in the homomorphism rules.

The next stage in this work is to incorporate some physics into the knowledge-based model. A global driving force, for example to a compact global conformation, may be needed

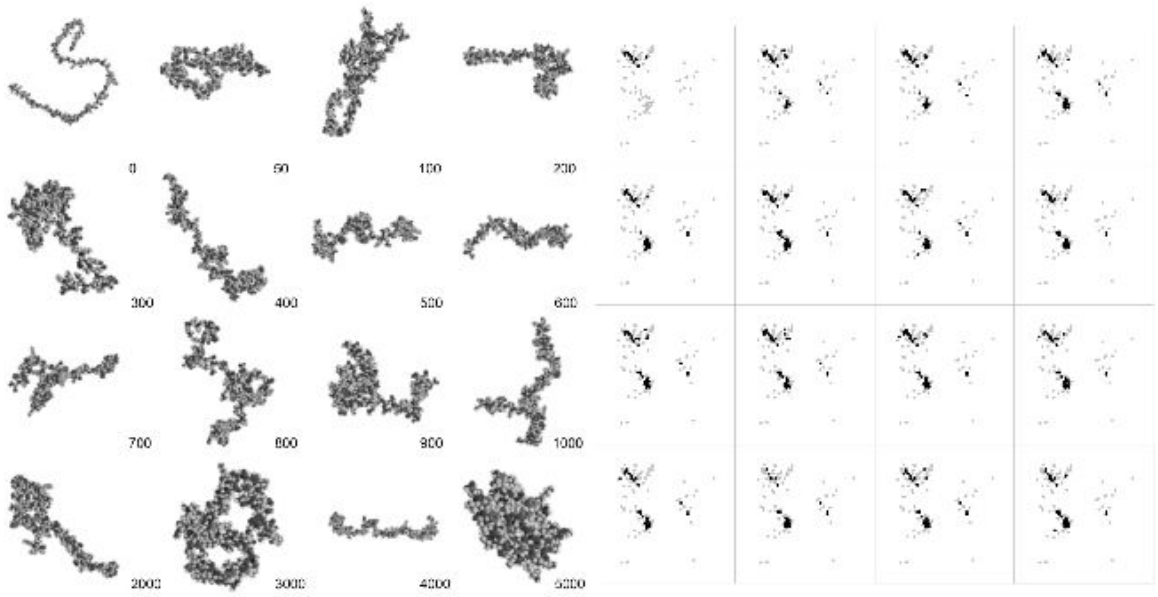


Figure 5: General features emerging from the L-system using the protein sequence of barnase, 1bnr. The initial state corresponds to an all extended conformation (image of state changes across all derivation steps shown in figure 3c). Images show the global changes in conformation, ϕ/ψ plots show the ϕ, ψ angles (black) for each amino acid at corresponding derivation steps with the native state angles shown in grey for reference.

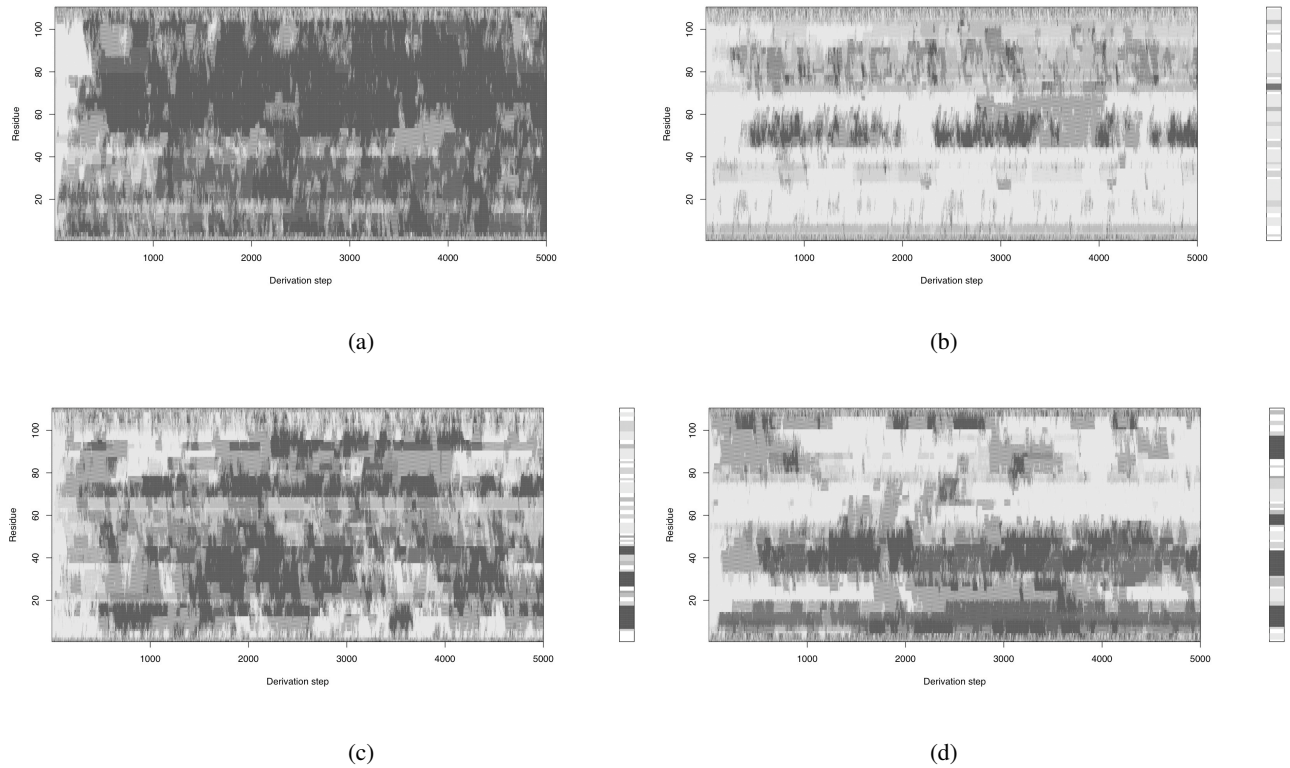


Figure 6: Results of protein 1exg in different initial conformations. Each image shows the states of individual residues (y-axis) at each derivation step (x-axis). Horizontal bands show the emergence of preferred local secondary structure. (a) initial state as all extended (b) initial state as all alpha (c) initial state all 3/10 helix (d) initial state in alternating alpha-beta.

to alter the probability table during folding. The combination of our simple physics-based L-systems model with the knowledge-based rules would also allow selection between states. This would allow local structural preference to work together with spatially local interactions and may lead to more *protein-like* structures that converge to a final folded state. Incorporating physics into the knowledge-based model may also help to prevent large global changes in conformations caused by unrestricted changes in local residue conformations.

Summary

We have presented an L-systems model that uses data-driven stochastic rewriting rules to fold protein sequences by altering the secondary structure state of individual amino acid residues. The state of each residue is rewritten in parallel across the whole protein. The state that an individual residue changes to depends on the amino acid type of that residue and the amino acid types and the current states of the neighbouring residues on either side. Seven secondary structure states are used based on those used in the DSSP database. The probabilities of adopting each of seven states were obtained from the frequencies of each state, given the states of residues either side, found in 10,954,172 3-residue fragments from 35,492 native protein structures in the DSSP database. Typical backbone ϕ, ψ torsion angles were also obtained for each amino acid type in each of the seven states from the data and used to reconstruct the 3D structure of a protein at each derivation step. This was used to assess the *protein-like* nature of global conformations.

Results are shown for four protein sequences from each major structural class. Local structure preference can be seen to emerge for some residues in a sequence. Overall differences in the proportion of local α-helix and extended conformations can also be seen between protein sequences using these rules. However, the resulting structures do not converge to a preferred global compact conformation. Further work will be to incorporate some physics-based bias into the probability table to allow a preferred global conformation to emerge.

Acknowledgements

This work is supported by the BBSRC. We thank Karim Elsawy for providing the fragment data.

References

Anfinsen, C. B. (1973). Principles that govern the folding of protein chains. *Science*, 181(96):223–230.

Bujnicki, J. M. (2006). Protein-structure prediction by recombination of fragments. *Chembiochem*, 7(1):19–27.

Danks, G. B., Stepney, S., and Caves, L. S. D. (2007). Folding protein-like structures with open L-systems. *ECAL 2007, LNCS*, 4648:1100–1109.

Dill, K. A., Bromberg, S., Yue, K. Z., Fiebig, K. M., Yee, D. P., Thomas, P. D., and Chan, H. S. (1995). Principles of protein-folding - a perspective from simple exact models. *Protein Science*, 4(4):561–602.

Duan, Y. and Kollman, P. A. (2001). Computational protein folding: From lattice to all-atom. *IBM Systems Journal*, 40(2):297–309.

Fitzkee, N. C., Fleming, P. J., Gong, H., Panasik, N., J., Street, T. O., and Rose, G. D. (2005). Are proteins made from a limited parts list? *TRENDS in Biochemical Sciences*, 30(2):73–80.

Ginalski, K. (2006). Comparative modeling for protein structure prediction. *Current Opinion in Structural Biology*, 16(2):172–177.

Hansmann, U. H. E. and Okamoto, Y. (1999). New monte carlo algorithms for protein folding. *Current Opinion in Structural Biology*, 9(2):177–183.

Kabsch, W. and Sander, C. (1983). Dictionary of protein secondary structure: Pattern recognition of hydrogen-bonded and geometrical features. *Biopolymers*, 22(12):2577–2637.

Lau, K. F. and Dill, K. A. (1989). A lattice statistical-mechanics model of the conformational and sequence-spaces of proteins. *Macromolecules*, 22(10):3986–3997.

Levinthal, C. (1969). How to fold graciously. *Mössbaun Spectroscopy in Biological Systems Proceedings, Univ. of Illinois Bulletin*, 67(41):22–24.

Lindenmayer, A. (1968). Mathematical models for cellular interactions in development. Parts I and II. *Journal of Theoretical Biology*, 18:280–315.

Mech, R. and Prusinkiewicz, P. (1996). Visual models of plants interacting with their environment. *SIGGRAPH 96, Computer Graphics*, pages 397–410.

Murzin, A. G., Brenner, S. E., Hubbard, T., and Chothia, C. (1995). SCOP - a structural classification of proteins database for the investigation of sequences and structures. *Journal of Molecular Biology*, 247(4):536–540.

Onuchic, J. N., LutheySchulten, Z., and Wolynes, P. G. (1997). Theory of protein folding: The energy landscape perspective. *Annual Review of Physical Chemistry*, 48:545–600.

Orengo, C. A., Michie, A. D., Jones, S., Jones, D. T., Swindells, M. B., and Thornton, J. M. (1997). CATH - a hierarchic classification of protein domain structures. *Structure*, 5(8):1093–1108.

Prusinkiewicz, P. and Lindenmayer, A. (1990). *The Algorithmic Beauty of Plants*. Springer.

Ramachandran, G. N., Ramakrishnan, C., and Sasisekharan, V. (1963). Stereochemistry of polypeptide chain configurations. *Journal of Molecular Biology*, 7(1):95–99.

Scheraga, H. A., Khalili, M., and Liwo, A. (2007). Protein-folding dynamics: Overview of molecular simulation techniques. *Annual Review of Physical Chemistry*, 58:57–83.

Zwanzig, R., Szabo, A., and Bagchi, B. (1992). Levinthal's paradox. *PNAS*, 89(1):20–22.

Unexpected Evolutionary Dynamics in a String Based Artificial Chemistry

James Decraene, George Mitchell, Barry McMullin

Artificial Life Laboratory - Research Institute for Networks and Communications Engineering
School of Electronic Engineering - Dublin City University, Ireland
Email: james.decraene@eeng.dcu.ie

Abstract

This work investigates closure in Cell Signaling Networks, which is one research area within the ESIGNET project[1]. We employ a string-based Artificial Chemistry based on Holland's broadcast language (*Molecular Classifier System, Broadcast Language*, or MCS.b). We present a series of experiments focusing on the emergence and evolution of self-maintaining molecular organizations. Such experiments naturally relate to similar studies conducted in artificial chemistries such as Tierra, Alchemy and Alpha-Universes. However, our results demonstrate some counter-intuitive outcomes, not indicated in previous literature. Each of these "unexpected" evolutionary dynamics (including an elongation catastrophe phenomenon) are examined and explained both informally and formally. We also demonstrate how the elongation catastrophe can be prevented using a multi-level selectional model of the MCS.b (which acts both at the molecular and cellular level). This work provides complementary insights into the understanding of evolutionary dynamics in minimal artificial chemistries.

Introduction

Cell Signaling Networks (CSNs) are complex biochemical networks of interacting molecules (proteins, ions, secondary messengers, etc.) occurring in living cells. Through complex molecular interactions (e.g., signal transduction), CSNs are able to coordinate critical cellular activities (e.g., cell differentiation, apoptosis) in response to internal and external stimuli.

As CSNs occur in cells, these networks have to replicate themselves prior to the cellular division. This allows the replicated CSNs to be "distributed" to the offspring cells. Errors may occur during this replication process, e.g., an offspring cell may inherit only a partial CSN. Thus resulting in potentially defective cells which would lead to a variety of undesired effects (e.g., premature cell death). As a result, the "fitness" of a cell is implicitly represented by the *survival* and *performance* of a cell in achieving self-maintenance *and* cell-level replication.

[1]ESIGNET: Evolving Cell Signaling Networks *in silico*, an EU FP6 project, contract no. 12789, http://www.esignet.net

Based on the above assumption, we hypothesize that CSNs may be regarded as subsets of closed (and thus self-maintaining) systems. The latter would have the additional ability to replicate themselves as a whole (cellular division). The signal processing ability of CSNs would emerge from the closure properties of these systems.

Examining such phenomena relates closely to other studies which have been conducted on Holland's Alpha-Universes (Holland, 1976), Tierra (Ray, 1991) and Alchemy (Fontana and Buss, 1994). Although these Artificial Chemistries (ACs) were developed for different purposes and were implemented differently, these systems exhibited common evolutionary phenomena such as the emergence of (collectively) autocatalytic reaction networks (Dittrich et al., 2001; McMullin, 2000). In this investigation, such classes of network are of interest as they would allow CSNs to self-maintain and replicate themselves. Moreover, as demonstrated in several ACs, it is commonly accepted that the emergence and maintenance of such collectively autocatalytic reaction networks is relatively trivial.

We introduce the *Molecular Classifier System, Broadcast Language System*, or MCS.b (J.Decraene et al., 2007). This addresses the reflexive nature of molecular species and automatically gives rise to an implicit molecular fitness function represented by the "replication" ability of the individual molecular species. We present a series of experiments focusing on the emergence of self-maintaining organizations and finally we examine the outcomes of these experiments together with possible modifications for further work.

Molecular Classifier Systems

Molecular Classifier Systems are a class of string-rewriting based AC inspired by the broadcast language (BL; see Holland, 1992). As opposed to more traditional string-rewriting systems, operations are stochastic and reflexive (no distinction made between operands and operators). The behavior of the condition (binding) properties and action (enzymatic functions) is defined by a language specified within the MCS. This "chemical" language defines and constrains the complexity of the chemical reactions that may be mod-

eled and simulated. In this AC, all reactants are catalytic in the sense that they are not consumed during reactions. These reactions result from successful molecular interactions which occur at random. When a reaction occurs, a product molecule is inserted in the reactor whereas another molecule, selected at random, may be removed from the reactor space (designating the system outflow).

A molecule may contain several condition/action rules which define the binding and enzymatic properties. A reaction between molecules occurs if at least one conditional part from any rules in a molecule A matches a target molecule B. A is regarded as an enzyme whereas B is regarded as a substrate molecule. When a reaction occurs, the action part from the satisfied rule in A is utilized to perform the enzymatic operations upon the bound substrate molecule B. This operation results in the production of another offspring (product). If several rules in A are satisfied by B, then one of these rules is picked at random and employed to carry out the enzymatic function.

We proposed a simplification of the BL (J.Decraene et al., 2007) which is used as the MCS chemical language resulting in the MCS.b system. MCS.b has some similarity with the Learning Classifier Systems, also pioneered by John Holland (Holland and Reitman, 1978); however there are also a number of differences. For example, the LCS strings are fixed length on an alphabet of $\lambda = \{\ 1,\ 0,\ \#\}$; whereas the BL strings are of variable length using a significantly larger alphabet of $\Lambda = \{1,\ 0,\ *,\ :,\ \Diamond,\ \nabla,\ \triangle,\ '\}$. BL strings are referred to as *broadcast devices*. A broadcast device is parsed into zero, one or more *broadcast units*, where each unit represents a single condition/action rule. The symbol $*$ separates broadcast units within a broadcast device. The symbol : separates a condition from an action within a single broadcast unit. $\{\Diamond,\ \nabla,\ \triangle\}$ are single/multiple character wildcards that may also copy matched (sub-)strings into output strings. A detailed description is omitted in this paper, see (J.Decraene, 2006) for full specification of our BL implementation.

Autocatalytic organizations

A series of experiments using the MCS.b is now outlined. These experiments first examine both the self-maintenance and the spontaneous emergence of autocatalytic molecules (i.e., molecules that can self-replicate). Both spontaneous emergence and self-maintenance were reported as easily obtained in Alchemy. Spontaneous emergence was not expected or reported for the original Tierra system; however, it did arise in the related Amoeba system, specifically devised for this purpose (Pargellis, 2001).

No selective advantages for universal replicases

An artifact of the BL's syntax is that it is moderately difficult to observe the spontaneous emergence of an individually autocatalytic molecule. Specifically, there are 4^8 $(65,536)$ distinct molecules of length 4 symbols (the minimal length to construct a functional/enzymatic molecule), of which only a single one ($R_0 = *\nabla : \nabla$) is autocatalytic. Although the probability of spontaneously obtaining such autocatalytic molecules is therefore quite low in MCS.b, the intuition was that, *once* such a molecule does appear, it should be able to rapidly fill the reaction space. This phenomenon was indeed observed in Alchemy and was expected to occur in MCS.b. We present here a series of experiments which explore and test this conjecture.

The behavior of the minimal self-replicase, R_0, is as follows. The matching condition is defined by a single symbol, ∇, which designates a multiple character wildcard. This indicates that R_0 may bind to any molecule. In addition when a reaction occurs between R_0 and a substrate molecule I_0, ∇ is assigned a value, being the matched substring of I_0. In this case, this will be the complete string I_0. A unique symbol ∇ also constitutes the action part of R_0. This specifies that the output string of R_0 is exactly the string bound by the ∇ in the condition part, i.e., a copy of I_0. Therefore the broadcast device R_0 is actually a "universal" replicase; which, by definition, means that it is also a *self*-replicase (in the special case that it binds to another instance of itself, i.e., $I_0 = R_0$). The "specificity" of R_0 is said to be *null*.

Fig. 1 presents a first experiment examining the behavior of R_0 averaged over 30 simulation runs. The broadcast "universe" (reaction space) is configured as follows:

- The system is seeded with 900 randomly generated molecules, each of length 10 symbols.
- In addition, 100 instances of R_0 are inserted.
- n_{max} designates the fixed maximum number of molecules that may be contained in the universe, $n_{max} = 1000$.
- Molecular interactions occur as follows: two molecules A and B are picked at random, A is considered as an enzyme and B as a substrate. If A can bind and react with B then a molecule C is produced. If the current size of the population, n, is less than n_{max} then C is simply added to the population (and n increases by 1); otherwise a molecule is picked at random and is replaced with C (and the population size remains unchanged at n).
- No mutation may occur in these experiments.
- A single "timestep" is arbitrarily defined as 50 molecular interactions.

A high concentration (0.1) of R_0 was chosen to minimise early extinction due simply to stochastic fluctuation.

From Fig. 1 it is clear that the species R_0 never grows to take over the population; on the contrary, it consistently diminishes, contrary to the original, informal, prediction. A

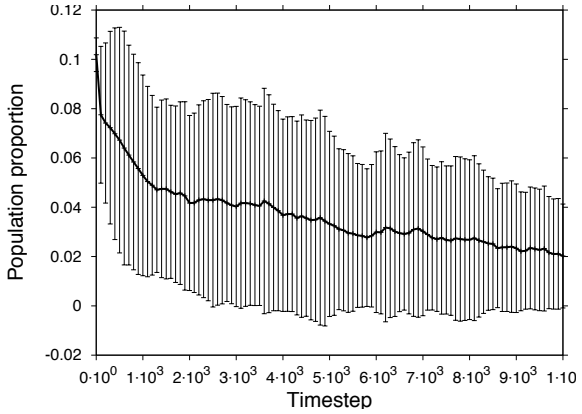


Figure 1: Relative population growth of replicators R_0 averaged over 30 simulation runs. Solid line is average concentration; error bars denote standard deviation.

formal explanation of this outcome is given by modelling the system with the (approximate, continuous) catalytic network equation (Stadler et al., 1993). The state of the system is described by the concentration vector $x = (x_1, \ldots, x_n)$ with $x_1 + \ldots + x_n = 1$ and $x_i > 0$, where x_i refers to the concentration of a molecular species (or collection of "chemically equivalent" species) s_i. The general dynamic behaviour is then given by:

$$\dot{x_k} = \sum_{i=1}^{n}\sum_{j=1}^{n} \alpha_{ij}^{k} x_i x_j - x_k \sum_{i,j,l=1}^{n} \alpha_{ij}^{l} x_i x_j \quad \text{with } k = 1, \ldots, n \tag{1}$$

α_{ij}^{k} are the rate constants for each reaction $s_i + s_j \rightarrow s_i + s_j + s_k$. In this experiment, these simplify to:

$$\alpha_{ij}^{k} = \begin{cases} 1 \text{ if } s_i + s_j \rightarrow s_i + s_j + s_k \\ 0 \text{ otherwise} \end{cases} \tag{2}$$

For simplicity, consider the simple case where only universal replicases (R_0) and non-enzymatic molecules (NE) (that may only act as substrates) are present. This is clearly the *most* favourable case for the growth of R_0. Denote the molecular concentrations of R_0 and NE by x_1 and x_2 respectively. Then $\alpha_{ij}^{1} = 1$ if $i = 1, j = 1$; otherwise $\alpha_{ij}^{1} = 0$. Similarly, $\alpha_{ij}^{2} = 1$ if $i = 1, j = 2$; otherwise $\alpha_{ij}^{2} = 0$. Inserting into Eq. 1, we obtain:

$$\dot{x_1} = x_1^2 - x_1(x_1^2 + x_1 x_2) \tag{3}$$

But given that $x_2 = 1 - x_1$:

$$\begin{aligned} \dot{x_1} &= x_1^2 - x_1^3 - x_1^2 + x_1^3 \\ \dot{x_1} &= 0 \end{aligned} \tag{4}$$

whereas the growth rate of molecules NE is:

$$\begin{aligned} \dot{x_2} &= x_1(1 - x_1) - (1 - x_1)[x_1^2 + x_1(1 - x_1)] \quad (5) \\ \dot{x_2} &= x_1 - x_1^2 - (1 - x_1)(x_1^2 + x_1 - x_1^2) \\ \dot{x_2} &= x_1 - x_1^2 - x_1 + x_1^2 \\ \dot{x_2} &= 0 \quad (6) \end{aligned}$$

Thus, both molecular species R_0 and NE share a common zero "expected" growth. Under the stochastic conditions of the reactor this would yield a random drift in relative concentrations—as opposed to a quasi-deterministic growth of the R_0 species. Qualitatively this is due to the fact that any (self-)replicase having low or zero specificity, such as R_0, will not only replicate itself but also replicate any other molecules; and therefore cannot selectively displace these molecules. But recall that this was the *best case* situation for growth of R_0, where none of the other molecules had any enzymatic activity. In the practical case of Fig. 1 the collection of such additional side reactions will give a nett negative growth rate for R_0, which therefore, quasi-deterministically, decays.

Specificity and domination of the replicases

To confirm the importance of specificity, we proceeded to a series of experiments in which we incrementally increased the specificity of the (self-)replicases. Table 1 shows the different replicases employed in these experiments. R_1 designates a molecule that would only react with molecules whose strings end with the symbol "1". As the latter occurs at the rightmost position of R_1, it may react with itself, producing another instance of R_1. Similarly, R_2 only binds to molecular strings containing the suffix 01. This "signature" forms a constraint on the replicases, allowing them to react only with a progressively more restricted set of substrate molecules. This impacts directly on these molecules' binding specificity.

Replicase	Informational string
R_0	$*\nabla : \nabla$
R_1	$*\nabla 1 : \nabla 1$
R_2	$*\nabla 01 : \nabla 01$
R_3	$*\nabla 101 : \nabla 101$
R_4	$*\nabla 0101 : \nabla 0101$

Table 1: (self-)replicases with increasing specificity

The results depicted in Fig. 2 confirm the importance of specificity upon the system dynamics. The ability of a (self-)replicase to dominate and sustain itself, against a random initial population of molecules, increases progressively with its binding specificity. As in the previous section, we can explain and demonstrate this behavior through the use of a simple ODE model.

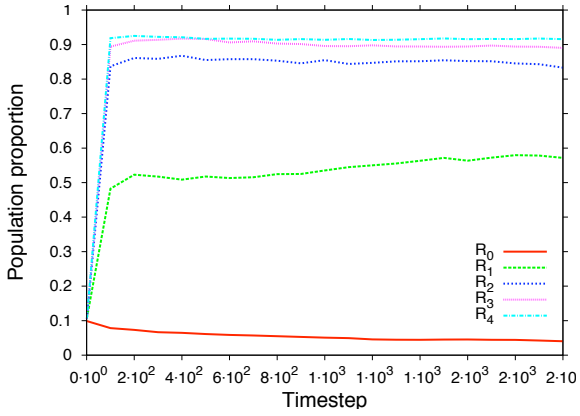


Figure 2: Population growth of replicators R_0, R_1, R_2, R_3 and R_4. Each line represents the average concentration of corresponding replicase over 30 simulation runs.

In this case, we consider a reactor containing only the following molecular species:

- Replicases R_1 which only replicate molecules terminating with the symbol "1" (which includes R_1 molecules themselves).
- A variety of non-enzymatic molecules NE which are randomly generated. $NE_1 \subseteq NE$ is the subset of molecules whose strings terminate with the designated symbol. These molecules contained in NE_1 can be replicated by molecules R_1.

The concentration vector is given by $x = (x_1, x_2, \ldots, x_n)$ with $x_1 + x_2 + \ldots + x_n = 1$ where x_1 is the concentration of R_1 and x_2 is the sum of concentrations of molecules in NE_1. The growth rate of the different molecular species in this reactor are as follows:

$$\begin{aligned} \dot{x_1} &= x_1^2 - x_1(x_1^2 + x_1x_2) && (7) \\ \dot{x_1} &= x_1^2 - x_1^3 - x_1^2x_2 \\ \dot{x_1} &= x_1^2(1 - x_1 - x_2) && (8) \end{aligned}$$

The growth rate of molecules NE_1 is:

$$\begin{aligned} \dot{x_2} &= x_1x_2 - x_2(x_1^2 + x_1x_2) && (9) \\ \dot{x_2} &= x_1x_2 - x_1^2x_2 - x_1x_2^2 \\ \dot{x_2} &= x_1x_2(1 - x_1 - x_2) && (10) \end{aligned}$$

Since $x_1 + x_2 + \ldots + x_n = 1$, we have $x_1 + x_2 < 1$ and therefore $\dot{x_1} > 0$ and $\dot{x_2} > 0$. Whereas the growth rate of any other molecules (that may be not replicated by R_1) in the reactor space is given by:

$$\begin{aligned} \dot{x} &= 0 - x_i(x_1^2 + x_1x_2) && (11) \\ & \text{with } 2 < i \leq n \end{aligned}$$

In Eq. 12, we note that any given molecules $s = (s_3, \ldots, s_n)$ possess a negative growth rate which indicate that these molecules would be displaced by molecules R_1 and NE_1.

In this model, only NE_1 molecules are able to parasite the replicases R_1. By increasing the specificity of replicases, we decrease the range of molecule that may parasite the replicases. This explains the behavior observed in Fig. 2, in which replicases with higher specificity are more likely to take over the reactor space.

Therefore in this system, for replicase molecules to successfully sustain themselves and/or to dominate the molecular population, a significant binding specificity is required. We conjecture that this underlying phenomenon may have been implicated in the dynamics of a variety of previously reported artificial chemistries; but, to our knowledge, it has not previously been explicitly isolated in the manner presented here.

Spontaneous emergence of replicases

In the previous set of experiments, mutation was turned off in order to facilitate our investigation on replicases, which were hand-designed and inserted into the initial population. This led to a limited diversity in the population. To examine the spontaneous emergence of autocatalytic molecules, we performed a second series of experiments in which no replicases are specified and molecular mutation could occur. The latter is implemented as follows:

- When a new molecule is produced, a mutation with probability $p_{sym} = 0.001$ may be applied to each of its symbols. Therefore, the longer the molecule, the higher the probability of mutation occurring.
- Three types of mutation are distinguished and are applied with equal probabilities:
  - *Symbol flipping:* The current symbol is replaced with a symbol picked uniformly at random from Λ.
  - *Symbol insertion:* A symbol is picked uniformly at random from Λ and inserted after the current symbol.
  - *Symbol deletion:* The current symbol is removed.
- To maintain diversity in the event of low ongoing reaction activity, a global mutation technique occurring every 100 timesteps is also available. A subset ($r_{mut} = 0.01$) of the population is selected at random and one of the three types of mutation mutation (chosen as above) is then applied to a single symbol picked uniformly at random in each molecule of this subset.

As mutation now occurs, diversity is maintained during long term evolution. The spontaneous appearance of replicators was expected. Results indicated that (self-)replicases do emerge, however they never manage to self-sustain.

This is explained as follows:

- As already noted, the BL syntax does not strongly facilitate the spontaneous emergence of replicators. This syntactical constraint may discourage the spontaneous emergence of self-replicators. The BL syntax may also have an impact on the robustness of these self-replicators against mutation effects.

- Secondly if self-replicators do emerge, they would be required to possess a specificity higher than null to sustain themselves.

- Finally, replicators are likely to possess a low molecular concentration when emerging. This low concentration diminishes the capacity of these molecular species to persist against side reactions and mutation events.

These three factors, when combined, significantly lower the probability of having a replicator spontaneously emerge and self-sustain in the MCS.b.

We examined the nature of the (self-)replicases that may emerge during evolution. An additional set of experiments was specified as follows:

- Each simulation run was initialised with 100 randomly generated, 10-symbol long, molecules.

- $n_{max} = 1000$ (i.e., the population initially grew without any displacement; but once the total number of molecules reached 1000 it was limited to this value, by displacing one random molecule for each new molecule generated, as previously described).

- 30 simulation runs were performed, each for 100000 timesteps.

To identify spontaneously emerging self-replicases, every molecule was tested at each timestep for self-replication functionality. The spontaneously emerging self-replicases identified in these experiments are listed in Table 2. This shows that 15 distinct self-replicases appeared. However, note that it is a property of the BL syntax that some symbols are ignored when functionally interpreted (they are, in a certain sense, "junk" symbols). Thus, although 15 distinct self-replicases were identified, it turns out that the core broadcast units (the "active sites", after discarding "junk" symbols) are, in fact, identical for 14 of these; and are all equivalent to the original universal self-replicase, $R_0 = *\nabla : \nabla$, discussed earlier. Only the broadcast device $*\nabla 0\nabla\nabla : \nabla 0$ possesses a core broadcast unit of a different form, namely $*\nabla 0 : \nabla 0$. This is an alternate form of R_1, having just the minimal specificity of one symbol.

In the 30 experimental runs, the highest concentration achieved by any of these spontaneously occurring self-replicases was 0.001—i.e., just a single isolated molecule.

Self-replicases	
$00'\triangle * \nabla : \triangle\nabla\nabla * 0$	$1\nabla 0 * \nabla : \nabla$
$00'\triangle * \nabla : \triangle\nabla\Diamond\nabla * 0$	$1\triangle\nabla 0 * \nabla : \nabla$
$: 1 * \nabla : \nabla\Diamond : 1 * \nabla : \nabla\Diamond$	$: 0\nabla\nabla * \nabla : \triangle\triangle\nabla$
$: *\nabla : \nabla\nabla * 01$	$*\nabla * \nabla : \nabla\triangle\nabla\triangle\triangle$
$1\Diamond\nabla : *\nabla\nabla : \nabla :$	$*\nabla : *\nabla : \nabla\triangle\nabla\triangle\triangle$
$*\nabla : \nabla$	$*\nabla\nabla : \nabla$
$*\nabla 0\nabla\nabla : \nabla 0$	$\Diamond\nabla * \nabla * \nabla : \Diamond\nabla\nabla$
$\triangle 1 * \nabla : \nabla\Diamond$	

Table 2: Spontaneously emergent self-replicases in MCS.b

This is consistent with the comments earlier in this section, and the results of the previous section. It is progressively more difficult for self-replicases of higher specificity to spontaneously arise by chance (due to their greater length, and relatively rare frequency as defined by the BL syntax); but self-replicases of very low specificity (which do spontaneously occur) cannot grow to significant concentrations.

The spontaneous emergence of a "sustainable" self-replicase (i.e., of sufficient specificity to establish itself) remains theoretically possible in MCS.b. However, both the experimental results and the informal analysis presented here suggest that the expected emergence time would be extremely (perhaps infeasibly) long. While we have not formally quantified this, it appears that MCS.b therefore shares this property with the Tierra system.

Rise and fall of the fittest

In the Tierra system, a hand-designed molecule called the "ancestor" is manually introduced into the space. This initially grows to saturate the available core memory. The population subsequently evolves into a variety of collectively autocatalytic reaction networks (where Tierra "creatures" or programs are here considered analogous to "molecules"). Accordingly, our next step is to mirror this methodology, and introduce a hand-designed self-replicase of relatively high specificity into the MCS.b system.

However, the results indicate that MCS.b does *not* exhibit an evolutionary dynamic at all comparable to Tierra in this case. Fig. 3 presents an example of such an experiment. The "ancestor" self-replicators do, at first, quickly fill the reaction space ($n_{max} = 1000$), just as expected. However, this population immediately collapses again. The average molecular length then increases dramatically, while the overall reaction rate (indicating the average rate of binding between random molecules in the population) also collapses. In this particular run, molecules were arbitrarily limited to a maximum length of $BD_{lmax} = 500$. Other experiments, without such a limit, indicated that the growth in molecular length appeared to continue indefinitely, subject only to available physical (computer) resources.

As with the experiments discussed earlier, these results were not expected. In fact, certain mutants of the original au-

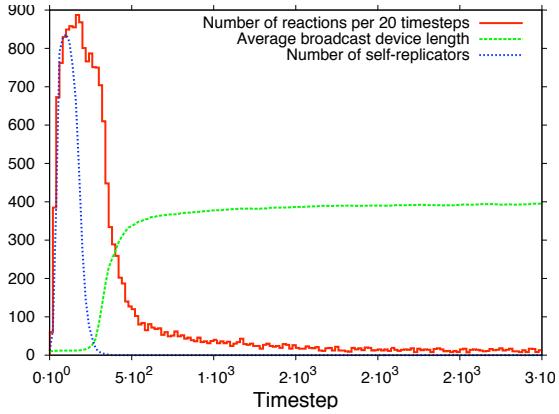


Figure 3: Effects of molecules length growth upon overall system reactions rates. In this experiment, an ancestor ($R_4 = \nabla 0101 : \nabla 0101$) is inserted (with initial concentration $[R_4] = 0.1$) in addition to randomly generated molecules. Moreover mutation per molecule and per symbol is turned on.

tocatalytic molecule developed a distinct advantage over the ancestor. That is, these mutants could be replicated by the ancestor molecules but only to the cost of these ancestors, i.e., an asymmetric relationship. Moreover, some of these mutants also lose their ability to self-replicate, explaining the rapid decrease in the global number of self-replicases. By exploiting their molecular signature and the ancestors, these non-autocatalytic molecules succeed in displacing the dominant ancestors.

To illustrate this phenomenon, we present a simple example of such a case in which we define two molecules: $R_4 = *\nabla 0101 : *\nabla 0101$ and $R'_4 = *\nabla 0101 : *\nabla 00101$. The latter is a readily accessible mutant of R_4. Once it appears, the mutant R'_4 allows for a runaway degenerative scenario to occur. The possible reactions are as follows:

$$
\begin{aligned}
R_4 + R_4 + X &\longrightarrow 3R_4 \\
R_4 + R'_4 + X &\longrightarrow R_4 + 2R'_4 \\
R'_4 + R_4 + X &\longrightarrow R_4 + 2R'_4 \\
R'_4 + R'_4 + X &\longrightarrow 2R'_4 + R''_4
\end{aligned}
$$

X is a molecule picked at random and removed from the population. The product R''_4 is of the form $*\nabla 0101 : *\nabla 000101$ and similarly has a selective advantage over both R_4 *and* R'_4. The reaction $R''_4 + R''_4 + X$ would result in the production of a molecule R'''_4 of the form $*\nabla 0101 : *\nabla 00000101$ and clearly shows the potential for unlimited elongation in molecule length. Of course, as molecule length increases, the per-molecule mutation rate also increases, leading to progressively more frequent disruptive changes to molecular structure. The observed consequences are twofold:

- Molecules may become inactive (i.e., lose all enzymatic activity). This is a direct consequence of the BL syntax. A mutation leading to the removal or insertion of structural symbols such as $*$ or : will commonly "break" the active site. This degenerative effect may be regarded as a consequence of syntactic "brittleness" of BL.

- The binding specificity may be increased. This arises when mutations lead to the insertion of informational symbols such as $0s$ and $1s$s. As a result, although some molecules may still possess an active site capable of some enzymatic function, their high specificity decrease the variety of target molecules that it can bind to; ultimately meaning there may be few, if any, functional targets for it left in the population.

Both of these phenomena result in a continual decrease in the overall reaction rate until reactions effectively cease completely (i.e., system death). Fig. 4 summarises this cascade of events. Note that this system level degeneration (the "elongation catastrophe") occurs precisely because of the stepwise emergence of molecules which are progressively "fitter" at the molecular level.

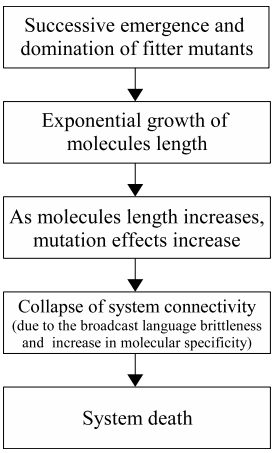


Figure 4: Elongation catastrophe in MCS.b

Fixing the elongation catastrophe: 1

In this section, we first describe different qualitative modifications conducted on the MCS.b, which were aimed at preventing the elongation catastrophe from occurring. These various technical modifications directed at limiting the string length of product molecules. Following this, the different outcomes are briefly presented.

1. In the system presented earlier, reactions leading to the production of molecules that were longer than BD_{lmax} were simply not permitted. An initial modification was to permit such reactions to proceed, but to truncate the product molecules at length BD_{lmax}. The system could

then remain active (with ongoing reactions) even though the molecules have reached a critical size.

2. The multiple symbols wildcard $\triangledown$ was altered so that it would not be able to pass an unlimited number of symbols from the input molecule (substrate) to the output molecule (product). An integer parameter $1 \leq c \leq c_{max}$ represents the number of symbols that can be matched and passed by $\triangledown$, i.e., the capacity of the wildcard. This capacity may be subjected to form of "parametric" mutation, where its value would change randomly in $[1, c_{max}]$ over time.

3. Similarly to (1), a finite total number, A, of "free" symbol objects (atoms) available in the broadcast universe was defined. This reservoir of (untyped) atoms is reduced when new molecules are produced, and increased when molecules are destroyed. If insufficient atoms are available to complete a reaction, the reaction fails. This should favor smaller molecules over longer ones suffering from elongation catastrophe.

4. Proposal (3) was extended, further constraints were defined to limit the number of particular symbols available in the universe. Different arbitrary symbols distributions were employed (e.g. structural and informational symbols such as $*$,:,1,0 could be made more frequent than multiple symbols wildcards such as $\triangledown$.)

5. Another extension to (3) was to vary the probability of a reaction to occur according to the product's length and A. Smaller molecules could then be given a selective advantage over the longer ones.

In summary, the above system changes generally produced one of the following outcomes:

- Did not prevent the elongation catastrophe.
- The system evolved towards a population of inactive and relatively small ($[1 - 4]$ symbols long) molecules. The system activity was also quasi null.
- The system converged towards a population where enzymatic molecules were still present but could not react with any other molecules present in the reaction space. The specificity continuously increased until no further reactions occur.

Thus, although a range of modifications were implemented, the different outcomes do not differ substantially from the degenerative cases presented above (section *Rise and fall of the fittest*).

Fixing the elongation catastrophe: 2

In this section we present an alternative approach to the MCS.b elongation catastrophe, based on *multi-level* selection. This has previously be demonstrated to be an effective means to provide resistance against parasites for catalytic networks Hogeweg and Takeuchi (2003). In such systems parasitized cells decay and may be displaced by neighboring healthy cells.

In the single-level selectional MCS.b model, competing molecules were contained in a single reactor, which we refer to as the molecular level of selection. In the multi-level selectional model, we introduce multiple reactors, each containing a population of molecules. These reactors ("cells") may be subjected to cellular division, which results in the replacement of the parent cell and creation of two offspring cells. However, the number of cells in the broadcast universe is fixed. As a result such a cellular division also triggers the removal of another cell selected at random. In a similar manner to molecules, cells are competing with each other which is regarded as the second level of selection.

In contrast to the single level model, successful reactions do not lead to the removal of a random molecule in the reaction space. Thus the number of molecules contained in a cell may increase until it reaches a finite limit l. When a cell reaches this size, a spontaneous division occurs. Half of the molecules are selected at random. These are removed from the "parent" cell and inserted into a newly created "daughter" cell. This is then inserted in the population of cells. Finally, a cell is picked at random (other than the parent and daughter cell) and removed from the population.

For time efficiency our multi-level model was implemented on a distributed, symmetrical, computer cluster where each cell was run on a single CPU. In this concurrent model, the fittest cells would not only be the cells that exhibit a high molecular growth rate, but cells that also contain molecules that are fast to compute (in real time). In other words, if we consider two cells which present an equal overall molecular growth rate, but contains molecules with different computational complexities, the cell which possesses a smaller overall molecular computational complexity will have the selective advantage.

We conducted a series of experiments as follows:

- 32 cells are employed.
- $l = 1000$ is the cell capacity.
- Mutation is turned on.
- Each cell is seeded with 250 replicases $R_4 = *\triangledown 0101 : \triangledown 0101$ and 250 randomly generated molecules of length 10.
- 5 simulation runs were conducted for at least 50 million molecular interactions per "cell object" (i.e., the run terminates when every concurrent cell object, one per CPU, has run for at least 50 million interactions each).

Results indicated that none of the evolved cells resulting from the simulation suffered from elongation catastrophe.

During an evolutionary run, we may observe the elongation catastrophe phenomenon to occur as expected. However we know that if the parasitic mutants appear in a cell, the cell would degenerate and not produce sufficient molecules to trigger cellular division and ultimately the displacement of another cell. As a result those cells would not have any selective advantages over the other "healthy" cells. On the contrary, healthy cells may still possess a high molecular reactivity and would consequently displace the infected cells.

Moreover, our results indicate that infected cells do not get displaced only when their connectivity is quasi null. In fact, these cells can get displaced at an early stage, when they would still present a high molecular activity although still being considered as being infected. As mentioned earlier, another fitness aspect to be considered is the computational complexity of molecules contained in a cell. Infected cells would rapidly produce molecules which have an increasing length, and this elongation has the effect of increasing their computational cost. As a consequence, although such cells may contain molecules with a richly connected reaction network, and therefore with a high continuing molecular replication rate, the overall cell growth rate is now penalized for having a higher molecular computational cost; as opposed to the healthy cells which generally still exhibit relatively short molecules and thus have lower computationally cost. This ultimately leads to a rapid displacement of infected cells whenever they would appear.

This multi-level selectional model successfully prevented the elongation catastrophe phenomenon from occurring. The nature of the evolved populations resulting from the simulation runs were at least somewhat comparable to those expected from systems such as Alchemy. Specifically, we have observed the rapid domination of molecular organizations which involve a range of replicases, capable of self-sustaining over time.

Conclusion

We conducted a series of experiments using the MCS.b system. These focused on the emergence and evolution of self-maintaining molecular organizations. Our results indicated counter-intuitive outcomes when compared with a variety of other AC systems in the literature. Each of these unexpected evolutionary dynamics was described and explained in detail. We also demonstrated how the elongation catastrophe can be prevented using a multi-level selectional model, which allowed for the evolution of organizations that were capable of self-sustaining over time. We propose to extend this multi-level selectional model by introducing new cellular division criteria, which would constrain and drive the evolution of the molecular networks. This may ultimately give rise to the emergence of proto-CSNs, being subsets of closed molecular systems, capable of some distinct CSN control-like features.

Acknowledgments

This work was funded by the EU FP 6 ESIGNET (Evolving Cell Signaling Networks in Silico)contract no. 12789. The computing facilities of ESIGNET and the Irish Center for High-End Computing (project code: dccom011c) are acknowledged. We are grateful to Ciarán Kelly for valuable comments.

References

Dittrich, P., Ziegler, J., and Banzhaf, W. (2001). Artificial Chemistries-A Review. *Artificial Life*, 7(3):225–275.

Fontana, W. and Buss, L. (1994). What Would be Conserved if "the Tape were Played Twice"? *Proc. of the National Academy of Sciences*, 91(2):757–761.

Hogeweg, P. and Takeuchi, N. (2003). Multilevel selection in models of prebiotic evolution: compartments and spatial self-organization. *Origins of Life and Evolution of the Biosphere*, 33(4-5):375–403.

Holland, J. (1976). Studies of the spontaneous emergence of self-replicating systems using cellular automata and formal grammars. *Automata, Languages, Development*, pages 385–404.

Holland, J. (1992). *Adaptation in natural and artificial systems*. MIT Press Cambridge, MA, USA.

Holland, J. H. and Reitman, J. S. (1978). Cognitive systems based on adaptive algorithms. In Waterman, D. A. and Roth, H. F., editors, *Pattern-Directed Inference Systems*. Academic Press, New York.

J.Decraene (2006). The Holland Broadcast Language. Technical Report ALL-06-01, RINCE, Dublin City University. `http://elm.eeng.dcu.ie/~alife/jd/ALL-06-01/`.

J.Decraene, Mitchell, G., and McMullin, B. (2007). Evolving Artificial Cell Signaling Networks: Perspective and Methods. *Advances in Biologically Inspired Information Systems: Models, Methods, and Tools*, 69.

McMullin, B. (2000). Remarks on Autocatalysis and Autopoiesis. *Annals of the New York Academy of Sciences*, 901(1):163–174.

Pargellis, A. N. (2001). Digital life bahavior in the amoeba world. *Artificial Life*, 7(1):63–75.

Ray, T. (1991). An approach to the synthesis of life. *Artificial Life II*, 10:371–408.

Stadler, P. F., Fontana, W., and Miller, J. H. (1993). Random catalytic reaction networks. *Phys. D*, 63(3-4):378–392.

Predictive information and emergent cooperativity in a chain of mobile robots

Ralf Der[1], Frank Güttler[2], Nihat Ay[1,3]

[1] Max-Planck Institute for Mathematics in the Sciences Leipzig, Germany
[2] University Leipzig Institute of Informatics, Germany
[3] Santa Fe Institute, Santa Fe, USA

Abstract

Measures of complexity are of immediate interest for the field of autonomous robots both as a means to classify the behavior and as an objective function for the autonomous development of robot behavior. In the present paper we consider predictive information in sensor space as a measure for the behavioral complexity of a chain of two-wheel robots which are passively coupled and controlled by a closed-loop reactive controller for each of the individual robots. The predictive information, the mutual information between the past and the future of a time series, is approximated by restricting the time horizons to a single time step. This is exact for Markovian systems but seems to work well also for our robotic system which is strongly non-Markovian.When in a maze with many obstacles, the approximated predictive information of the sensor values of an individual robot is found to have a clear maximum for a controller which realizes the spontaneous cooperation of the robots in the chain so that large areas of the maze can be visited.

Introduction

Despite much progress in biologically inspired robotics, biological systems are still singled out by a high degree of self-actualisation. This phenomenon is approached by the scientific community on different levels. Concepts like autopoiesis (Maturana and Varela, 1980) try to provide a general theoretical framework for the phenomena of self-creation and self-maintenance of living beings. On the other hand, concrete modes of action are formulated by mechanisms like homeostasis as a general theory of self-regulation (Ashby, 1954). It is widely believed that the integration of self-phenomena into artificial beings would not only lead to a better understanding of living beings but also to robots with internal motivation, curiosity, the self-exploration of bodily and environmental affordances, and quite generally to creative behaviors.

There are many different approaches towards the self-actualisation of behavior in autonomous robots. Relevant for this paper is the attitude that behavior is less a sequence of actions in order to reach a prespecified goal but instead a means for (i) structuring the input information (creating statistical correlations) the robot gathers with its sensors (Lungarella and Sporns, 2005); (ii) the maximization of the information flow in the sensorimotor loop (empowerment) (Klyubin et al., 2007); (iii) the maximization of the sensorimotor coordination (Lungarella and Sporns, 2006), and others. The main question is how this can be realized. There are interesting approaches of realizing systems on the basis of concrete modes of action like homeostasis, see for instance (di Paolo, 2003), but a more systematic way is by convenient measures for the information contained in or the complexity of the sensor stream. Of methodological interest are approaches of formulating general measures for the realisation of self-organisation (Shalizi et al., 2004).

This paper tries to further develop this direction in a concrete embodied robotic system. In order to further systematize the field we introduce the notion of self-referential robotic systems – adaptive, embodied systems where the objective of adaptation is a function of the robot's sensor values alone. In particular, there is no domain specific goal or externally specified aim formulated into this function. We favor predictive information measuring the complexity in sensor space as such an objective function. The predictive information of a process quantifies the total information of past experience that can be used for predicting future events. Technically, it is defined as the mutual information between the future and the past (Bialek et al., 2001). It has been argued that predictive information, also termed excess entropy (Crutchfield and Young, 1989) and effective measure complexity (Grassberger, 1986), is the most natural complexity measure for time series. The behaviors emerging from maximizing the PI are qualified by the fact that predictive information is high if – by its behavior – the robot manages to produce a stream of sensor values with high information content under the constraint that the consequences of the actions of the robot remain still predictable. This is why we favor predictive information as an objective function.

Under this paradigm, behaviors are entirely contingent, depending on the physical embodiment of the robot and the starting and environmental conditions. From the point of view of applications, the question of central interest is what kind of behaviors may be expected to arise with a given em-

bodiment and as a next step whether these behaviors are of any interest as behavioral primitives for the construction of higher-level goal-oriented strategies. It is in the nature of the question that there is not a unique answer but we are convinced that a certain systematics can be found at the level of phenomena. This paper considers the case of a chain of passively coupled two-wheel robots, each robot being controlled independently by a simple neural network under the closed-loop control paradigm. There is no central control so that a coherent motion of the chain is an emerging phenomenon based on a synchronisation of the wheels of the individual robots, the task being aggravated by the fact that the robots are moving in a maze with only narrow passages between obstacles. Nevertheless, we show that the predictive information of a single sensor value (the wheel velocity) of an individual robot is in close relationship to the ability of the chain to spontaneously self-organize into a coherent mode. In this mode, the chain may successfully navigate in the maze. This result extends the earlier finding (Ay et al., 2008) that the maximum MI in the sensor channels defines a working regime where the controller reacts in a specific way to the sensor values.

Our approach relates to other approaches of using statistical measures for robotics, a good introduction is (Lungarella et al., 2005) where a set of univariate and multivariate statistical measures are used in order to quantify the information structure in sensory and motor channels, see also (Klyubin et al., 2007) and (Klyubin et al., 2005). In particular we consider the predictive information as a prospective tool for concepts like internal motivation. Potential applications of this approach are expected in developmental robotics which has found some interest recently (Lungarella et al., 2003). There is a close relationship to the attempts of guiding autonomous learning by internal reinforcement signals (Stout et al., 2005) and to task independent learning (Oudeyer et al., 2005), (Schmidhuber, 2005), (Still, 2007). Quite generally, using a complexity measure as the objective function for the development of a robot corresponds to giving the robot an internal, task independent motivation for the development of its behavior.

The robot

In the present paper we are considering a chain of passively coupled two-wheel robots, Fig. 1, simulated in the *lpzrobots* simulation tool (Martius and Der, 2007) based on the physics engine ODE (Smith, 2005), which simulates in a realistic way effects due to the inertia of the robot, slip and friction the effects of the wheels with the ground and the effects of both the couplings and collisions. Each individual robot has a controller consisting of two neurons with the vector $x \in \mathbf{R}^2$ of the measured wheel rotation velocities as input and the vector $y \in R^2$ of nominal motor activities as output, i.e.

$$y_i = g\left(C_{i1}x_1 + C_{i2}x_2\right) \tag{1}$$



Figure 1: In the arena the chain of passively coupled two-wheel robots is simulated in the *lpzrobots* simulation tool. Each robot is "blind" and feels the environment only by the reactions of its wheel counters on collissions with the obstacles.

where $g(z) = \tanh z$, the controller matrix C defining the behavior of the system. In the present paper we want to determine empirically the predictive information over the controller parameters C_{ij} which parameterize the behavior of the robot.

The sensorimotor loop

If the wheels are moving freely we may assume that the nominal velocity y and the measured true velocity x are equal. In a realistic situation there will be perturbations so that we write the sensorimotor dynamics

$$x_{t+1} = y_t + \xi_{t+1} \tag{2}$$

where $x_t = (x_{t1}, x_{t2})^T \in R^2$ and ξ contains all the effects due to friction, slip, inertia and so on which make the response of the robot to its controls uncertain. In particular, if the robot hits an obstacle, the wheels may get totally or partially blocked so that in this case ξ may be large, possibly fluctuating with a large amplitude if the wheels are not totally blocked. Moreover ξ will also reveal whether the robot hits a movable or a static object. Additional strong effects result from the couplings between the robots which exert strong forces if the robots are not in complete synchrony.

In order to discuss the nature of the spontaneous cooperation phenomena observed we consider the trivial (but relevant, see below) case of a diagonal matrix C with $C_{11} = C_{22} = c$ so that the sensorimotor loop of each wheel is described by the one-dimensional system ($x_t \in R^1$)

$$x_{t+1} = g\left(cx_t\right) + \xi_{t+1}$$

the properties of which are obtained by analysing the fixed points obtained from

$$x = g\left(cx\right)$$

As discussed in earlier papers (Ay et al., 2008), the system has one stable FP for $0 < c < 1$ which becomes unstable at the bifurcation point $c = 1$ so that for $c > 1$ we have a bistable system with FPs $x = \pm q$ with q increasing for increasing c. The noise causes fluctuations around the FPs with occasional switching of the FPs, the probability of switching decreasing exponentially with increasing $c > 1$. There is however a subtlety in the fact that, under the noise, the bifurcation is effectively taking place only at the so called effective bifurcation point which is at $c = 1 + \delta$ with $0 < \delta \ll 1$ and δ increasing with the noise. This region is of particular interest since it is there that the wheel velocities are already quite large but can easily be switched by noise events caused for instance by collisions with obstacles or by the influence of the forces exerted by the other robots in the chain. In fact, due to the effect described, the wheel velocity will feel a tendency to switch sign if a torque in the opposite direction is exerted on it by the other robots in the chain. By switching the velocity, the wheel is now acting in the direction of the force exerted on it and this is the self-amplification effect necessary for the occurrence of self-organization.

The videos at (Martius and Der, 2007) demonstrate quite clearly the strength of this self-organized synchronization effect which not only makes the robot chain move into one direction but also keeps it still explorative in the sense that after some time it also inverts its direction of motion. Moreover, when colliding with a wall the chain of robots often will change velocity in an integrated manner. Finally and most importantly for the topic of the present paper, it will also effectively explore the spatial extensions of a maze the chain is put into.

Information theoretic measures

The central aim of this paper is the relation between the internal world of the robot, based on a complexity measure of its sensor values, and its relation to the external world. As motivated above, a convenient complexity measure is predictive information in sensor space, i.e. we consider the time series $S = \{X_t | t = 0, 1, 2, \ldots\}$ of the sensor values of the behaving robot.

Predictive information

The predictive information is the mutual information between the future and the past, relative to some instant of time t, of the time series S

$$I\left(X_{past}; X_{future}\right) = \left\langle \log_2 \frac{p\left(X_{past}, X_{future}\right)}{p\left(X_{past}\right) p\left(X_{future}\right)} \right\rangle$$

where the averaging is over the joint probability $p\left(X_{past}, X_{future}\right)$, time horizons of both past and future extending to infinity. This expression simplifies considerably if X is a Gauss-Markov process, see (Ay et al., 2008). In this case the time horizon can be restricted to just a single step so that the PI is given by the mutual information (MI) between two successive time steps, i.e.

$$I\left(X_{past}; X_{future}\right) = \left\langle \log_2 \frac{p\left(X_{t-1}, X_t\right)}{p\left(X_{t-1}\right) p\left(X_t\right)} \right\rangle \quad (3)$$

which simplifies the sampling process considerably. Moreover, in the experiments we observed that it is sufficient to study the MI of just a single sensor, one of the wheel counters of an individual robot, and still get the full information on the behavior of the robot chain.

Of course our time series of sensor values is far from being a Gauss-Markov process. However, as shown in (Ay et al., 2008) in specific cases the full PI can very well be approximated by that of a process with white Gaussian noise of conveniently chosen strength. The reason for this agreement probably is in the fact that both in linear and in weak noise nonlinear dynamical systems the PI does not depend on the noise at all. The PI however was found to depend very sensitively on the parameters of the controller which define the behavior of the robot.

This result is very important for the practical use of the PI. In fact, it tells us that in many cases the actually infinite time horizons may be restricted to just a few steps without losing much of the information on the behavior of the system as a function of the controller parameter.

The self-referential robotic system

The PI is given in terms of the sensor values the robot produce in the course of time alone. There is no domain specific knowledge invoked into this function. We obtain a self referential robotic system when using the PI as the objective function for the adaptation of the parameters of the controller. In particular we may consider the gradient ascent on the MI as given by eq. 3

$$\Delta m_t = \varepsilon \frac{\partial I\left(X_t; X_{t-1}\right)}{\partial m_t}$$

where m is any parameter of the controller of the robot. The properties of the self-referential robotic system depends also on the choice of the learning rate ε which actually has to be chosen small enough so that the time scales are well separated.

The learning rule has been substantiated earlier (Ay et al., 2008) for the case of a simple sensorimotor loop and was shown to reduce to a simple synaptic dynamics consisting of a general driving term plus an anti-Hebbian learning term. The example shows that the sampling problem with the PI can be partially avoided and the gradient obtained explicitly

if convenient approximations are made. We do, however, not aim to derive concrete learning rules in this paper but instead try to further elucidate the role of the PI with restricted time horizons.

The experiment

In order to keep the sampling effort manageable we use, based on symmetry arguments, two different parametrizations of the matrix C chosen such that there are only two parameters to be varied. The experiments have been carried through on a LINUX cluster of the Max Planck Institute for the Mathematics in the Sciences with about 100 nodes and have been run for $500,000$ time steps each. Results are averages over three runs for each pair of parameter values, see below.

Symmetric cross channel couplings

The first parametrization of the matrix C is given by

$$C = \begin{pmatrix} c & b \\ b & c \end{pmatrix} \tag{4}$$

We may extend the FP analysis of a single robot given above to the present case by assuming that the wheel velocities are $x_1 = x_2 = v$ (straight on motion) or $x_1 = -x_2 = v$ (on-site rotation) so that the FP is now obtained from the solution of

$$v = g\,(rv)$$

where $r = c + b$ or $r = c - b$ plays now the role of the feed-back strength in the loop for the straight or rotational motion, respectively. Obviously, with $b > 0$ we find that in the bifurcated region ($r > 1$) FPs are more stable for the straight on motion whereas with $b < 0$ the rotational motion is favored.

The central questions of our investigation is the behavior of the MI as a function of the behavior parameters of the robot and its relation to the behavior in physical space. Our robot chain has complicated physical properties, the videos might give an impression of the range of behavioral possibilities. When in the maze, the information transmission between the individual robots, which takes place by the physical forces transmitted via the passive coupling elements in a rather intricate way, is further corrupted by the collisions of the robots with the obstacles and the bumpers of adjacent robots in the chain, see the videos. Nevertheless, we observe for certain parameter combinations of the controller that the robot chain covers a wide area of the maze which is a clear indication of successful cooperation between the individual robots.

Figure 2 shows the MI as a landscape over the parameters c and b of the controller. We find a clear ridge structure the ridge running along the curves given by

$$c + |b| \approx 1.1$$

which means that the MI has a relative maximum close to the effective bifurcation point (now realized in the coupled system) be it either in the rotational ($b < 0$) or straight on ($b > 0$) mode. The rotation mode seems a little surprising at this point since in the chain the individual robot can not rotate. Most probably this is explained by the fact that we evaluated the MI for the first robot in the chain, which would execute an oscillatory motion by switching between the two rotational modes repeatedly. In further experiments we will evaluate the MI for the inner robots as well. The landscape moreover displays a clear local maximum which is at $b = 0$ and $c \approx 1.1$ meaning that the two channels are decoupled so that the best cooperation in the chain is if each wheel is controlled individually such that its single channel MI is maximal. This is also a little surprising since one would have expected that cooperation in the chain is best if the straight on motion is supported.

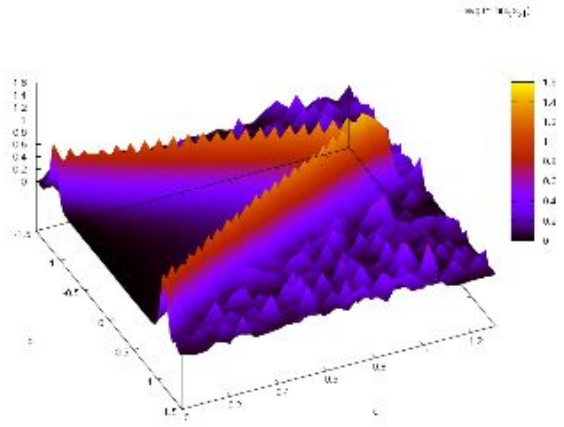


Figure 2: Th average mutual information of the sensor value for the case of symmetric cross channel couplings where $c = C_{11} = C_{22}$ and $b = C_{12} = C_{21}$.

The diversity of sites visited by the robot in the course of a fixed time is measured by the entropy of the probability distribution over the sites. As seen from Fig. 3, the landscape is very similar to the that of the MI. We have the absolute maximum at the same position so that the main message of the MI (individual control of the wheels) is corroborated. However there are also differences. On the one hand we see that the ridge corresponding to a preferred rotational motion ($c < 0$) is not so high as the one for the straight on motion. Moreover we find that there is a second clear maximum at around $c = 0$ and $b = 1.1$ corresponding to the case that the direct coupling is zero so that the control of the wheel is completely based on the angular velocity of the opposite wheel. The MI seems to have also a small local maximum in this region but this needs corroboration by a better statistics.

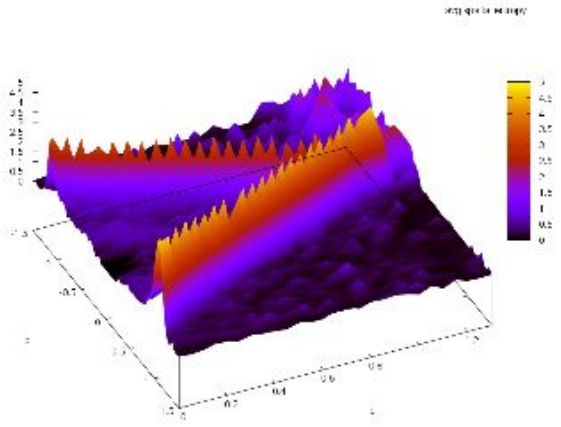


Figure 3: The entropy of the probability distribution over the sites that could be visited by the chain of robots in the maze. The entropy is over the parameters c and b of the couplings in and across channels. The entropy is maximal if all sites are visited by the chain with equal probability and is zero if the robot remains in its starting position.

Nevertheless there is a strong correlation between the MI in sensor space and the behavior of the robot as measured in physical space. This has an even stronger implications than in the single robot case considered in (Ay et al., 2008). Noting that the MI is taken by considering just one of the sensor values (wheel velocities) of an individual robot (the first one in the chain) we may conclude that the adaptation according to the maximum MI principle makes the robot capable of effectively cooperating in a collective of robots without any central control.

Antisymmetric cross channel couplings

Our second parametrization is taken as

$$C = \begin{pmatrix} c & -b \\ b & c \end{pmatrix} = \alpha \begin{pmatrix} \cos\phi & -\sin\phi \\ \sin\phi & \cos\phi \end{pmatrix}$$

which is known to support a Neimark-Sacker bifurcation if α exceeds 1 into an oscillatory regime (Pasemann et al., 2003), with frequency roughly given by $f \sim \phi/2\pi$. The landscape of the MI and the spatial entropy now are even more similar (although the ridge of the spatial entropy landscape is more pronounced) so that we depict only that of the MI. We see again the maximum of the single channel control ($\alpha = 1.1$, $\phi = 0$) as observed above. However, surprisingly there is clear second maximum at $\alpha \approx 2.35$ which corresponds to a very strong direct coupling combined with a very high frequency of about 3 Hz of the oscillations of the wheel velocities. This maximum is clearly seen for the MI and is even more pronounced for the entropy of the spatial distribution. In order to understand the phenomenon that the chain well manages to navigate in the maze with the strategy of rapidly (but still sensitively since under the closed loop control paradigm) switching wheel velocities we have to note that in our simulations we use values for the friction and slip parameters corresponding to a snow underground. This setting was chosen with the intention that emerging cooperation is possible best for a chain of sensitive "drivers". The strange high frequency regime is counterintuitive to this argument. Possibly this strategy is useful since it may excite a kind of navigation by controlled skidding but this will need further investigations.

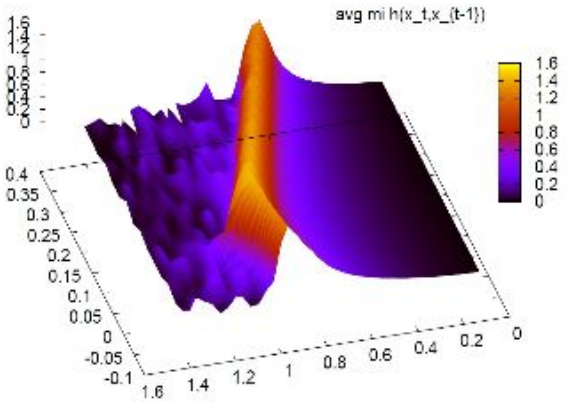


Figure 4: The MI for the case of the antisymmetric cross channel coupling. The landscape is only for positive ϕ because of the symmetry against sign inversion of ϕ. There are two local maxima of about the same height at $(\alpha = 1.1, \phi = 0)$ and $(\alpha = 2.35, \phi = .35)$.

The decisive point however is that obviously under the present parametrization the MI singles out specific nontrivial behavior modes which unexpectedly represent an effective control strategy. This is one further hint for the usefulness of the predictive information as a general tool for the self-organisation of behavior.

Concluding remarks

This paper has investigated the usefulness of predictive information for the self-organisation of behaviour in a chain of passively coupled robots. Predictive information has been approximated by the mutual information of sensor values over one time step which is much better accessible in real systems. Despite of this drastic simplification, we have shown that the maximum of the MI specifies a working regime of the single robot where it can effectively cooperate in the chain under difficult environmental conditions. Thus, the predictive information of even a single sensor channel is seen to be a far reaching indicator of the global behavior in physical space. In other words it is a link between the

internal world of the robot (sensor space) and the behavior in the external world which is maximum if the behavior of the robot is "rich" but with a high degree of self-established sensorimotor coordination.

This concept will be continued in further work where we will in particular investigate in how far the extension of the time horizon, in particular into the past, will give measures which are more discriminative. For instance, we want to understand if such a more extended measure is able of discriminating between the two control modes of preferentially straight or rotational modes which are different in their spatial behavior but not so much in the MI. The present results clearly support the point of view that the link between the information measure in sensor channels and the behavior of the robot is of a more fundamental nature, as claimed for instance in (Lungarella and Sporns, 2006). This suggests, as a possible application, the use of the PI as an auxiliary fitness function in artificial evolution which helps driving agents into working regimes with high prospectives for emerging functionalities. This will be one of our future projects.

Another focus is on the relation of the PI to another complexity measure, the so called time loop error, and the principle of homeokinesis (Der and Liebscher, 2002), (Der et al., 1999), (Der, 2001), which has been the basis for concrete learning rules leading to the self-organization of explorative behaviors in complex robots with many degrees of freedom in dynamic, unstructured environments, see (Der et al., 2006), (Der and Martius, 2006), (Der et al., 2005) and the videos on `http://robot.informatik.uni-leipzig.de/`. We hope in the near future to produce similar results on the basis of information theoretic measures. Preliminary results indicate that the gradients of the time loop error and the mutual information can be related to each other by a change in the metric of the parameter space.

Acknowledgements

Nihat Ay thanks the Santa Fe Institute for supporting him as an external professor. The facilities provided by the LINUX cluster of the Max Planck Institute for the Mathematics in the Sciences are gratefully acknowledged.

References

Ashby, W. R. (1954). *Design for a Brain*. Chapman and Hill, London.

Ay, N., Bertschinger, N., Der, R., Güttler, F., and Olbrich, E. (2008). Predictive information and explorative behavior of autonomous robots. *European Physical Journal: Complex Systems*. to appear.

Bialek, W., Nemenman, I., and Tishby, N. (2001). Predictability, complexity and learning. *Neural Computation*, 13:2409.

Crutchfield, J. P. and Young, K. (1989). Inferring statistical complexity. *Phys. Rev. Lett.*, 63:105–108.

Der, R. (2001). Self-organized acquisition of situated behavior. *Theory in Biosciences*, 120:179–187.

Der, R., Hesse, F., and Martius, G. (2005). Rocking stamper and jumping snake from a dynamical system approach to artificial life. *J. Adaptive Behavior*, 14:105 – 116.

Der, R. and Liebscher, R. (2002). True autonomy from self-organized adaptivity. In *Proc. Workshop Biologically Inspired Robotics. The Legacy of Grey Walter 14-16 August 2002, Bristol Labs*, Bristol.

Der, R. and Martius, G. (2006). From motor babbling to purposive actions: Emerging self-exploration in a dynamical systems approach to early robot development. In Nolfi, S., editor, *From Animals to Animats*, volume 4095 of *Lecture Notes in Computer Science*, pages 406–421. Springer.

Der, R., Martius, G., and Hesse, F. (2006). Let it roll – emerging sensorimotor coordination in a spherical robot. In Rocha, L. M., editor, *Artificial Life X*, pages 192–198. MIT Press.

Der, R., Steinmetz, U., and Pasemann, F. (1999). Homeokinesis - a new principle to back up evolution with learning. In *Computational Intelligence for Modelling, Control, and Automation*, volume 55 of *Concurrent Systems Engineering Series*, pages 43–47, Amsterdam. IOS Press.

di Paolo, E. (2003). Organismically-inspired robotics: Homeostatic adaptation and natural teleology beyond the closed sensorimotor loop. In Murase, K. and Asakura, T., editors, *Dynamical Systems Approach to Embodiment and Sociality*, pages 19 – 42, Adelaide. Advanced Knowledge International.

Grassberger, P. (1986). Toward a quantitative theory of self-generated complexity. *Int. J. Theor. Phys.*, 25(9):907–938.

Klyubin, A. S., Polani, D., and Nehaniv, C. L. (2005). Empowerment: A universal agent-centric measure of control. In *Proc. CEC*. IEEE.

Klyubin, A. S., Polani, D., and Nehaniv, C. L. (2007). Representations of space and time in the maximization of information flow in the perception-action loop. *Neural Computation*, 19:2387–2432.

Lungarella, M., Metta, G., Pfeifer, R., and Sandini, G. (2003). Developmental robotics: a survey. *Connect. Sci.*, 15(4):151–190.

Lungarella, M., Pegors, T., Bulwinkle, D., and Sporns, O. (2005). Methods for quantifying the informational structure of sensory and motor data. *Neuroinformatics*, 3(3):243–262.

Lungarella, M. and Sporns, O. (2005). Information self-structuring: Key principle for learning and development. In *Proceedings 2005 IEEE Intern. Conf. Development and Learning*, pages 25–30.

Lungarella, M. and Sporns, O. (2006). Mapping information flow in sensorimotor networks. *Comput Biol*, 2(10):e144.

Martius, G. and Der, R. (2007). lpzrobots - simulation tool for autonomous robots. `http://robot.informatik.uni-leipzig.de/`.

Maturana, H. R. and Varela, F. J. (1980). *Autopoiesis and Cognition. The Realization of the Living*. Reidel, Dordrecht.

Oudeyer, P.-Y., Kaplan, F., Hafner, V. V., and Whyte, A. (2005). The playground experiment: Task-independent development of a curious robot. In Bank, D. and Meeden, L., editors, *Proceedings of the AAAI Spring Symposium on Developmental Robotics, 2005, Pages 42-47, Stanford, California, 2005.*

Pasemann, F., Hild, M., and Zahedi, K. (2003). SO(2)-networks as neural oscillators. In Mira, J. and Alvarez, J., editors, *Computational Methods in Neural Modeling*, pages 144–151, Berlin, Heidelberg, New York. Springer.

Schmidhuber, J. (2005). Completely self-referential optimal reinforcement learners. In *ICANN (2)*, pages 223–233.

Shalizi, C. R., Shalizi, K. L., and Haslinger, R. (2004). Quantifying self-organization with optimal predictors. *Phys Rev Lett*, 93(11):118701.

Smith, R. (2005). Open dynamics engine. `http://ode.org/`.

Still, S. (2007). Statistical mechanics approach to interactive learning. *arXiv:0709.1948v1 [physics.data-an]*. submitted.

Stout, A., Konidaris, G., and Barto, A. (2005). Iintrinsically motivated reinforcement learning: A promising framework for developmental robotics. In *The AAAI Spring Symposium on Developmental Robotics*.

Artificial-Life Ecosystems: What are they and what could they become?

Alan Dorin[1], Kevin B. Korb[1] and Volker Grimm[2]

[1]Faculty of Information Technology, Monash University, Clayton, Australia 3800
[2]UFZ, Center for Environmental Research - UFZ, Department of Ecological Modelling, Permoserstr. 15 04318 Leipzig, Germany
alan.dorin@infotech.monash.edu.au

Abstract

This paper summarises the history of the terms *ecology* and *ecosystem*, before examining their application in the early and recent literature of A-Life agent-based software simulation. It investigates trends in A-Life that have led to a predominance of simulations incorporating artificial evolution acting on generic agents, but lacking a level of detail that would allow the emergence of phenomena relating to the transfer and transformation of energy and matter between the virtual abiotic environment and biota. Implications of these characteristics for the relevance of A-Life's virtual ecosystem models to Ecology are discussed. We argue a position that the inclusion of low-level representations of energetics, matter and evolution, in concert with pattern-oriented modelling techniques from Ecology for model validation, will improve the relevance of A-Life models to Ecology. We also suggest two methods that may allows us to meet this goal: artificial evolution can be employed as a mechanism for automating pattern-oriented ecological modelling from the level of individual species up to that of the ecosystem, or it may be employed to explore general principles of ecosystem behaviour over evolutionary time periods.

Introduction

As even a cursory survey of the early and current literature reveals, within the fields of Artificial Life and Ecological Modelling, agent (individual)-based virtual ecosystem model construction has been widely practiced ([1-4] are some early examples from A-Life, also see [5], for surveys of ecological examples [6], and more recently [7, 8]). Of course there is overlap between Ecological Modelling and A-Life publications in this regard, but a careful elucidation of the differences between the historical and current trends in the fields' approaches to virtual ecosystem construction allows us to recommend a mechanism for overcoming some of their limitations. We suggest this may be achieved by melding the approaches of both fields into models that explicitly represent energetics, matter (chemical stoichiometry) and evolution within a single simulation framework. The need for including these three frameworks was noted in the literature some time ago [9, 10]. Additionally, any purportedly descriptive simulations must be validated against ecological data.[1] One method to achieve this is through *pattern-oriented modelling* [11], a technique summarised below.

We will discuss two under-explored ways in which hybrid A-Life/Ecology ecosystem simulations of this kind may be built. Firstly, models of energy and matter transfer adopted from A-Life's artificial chemistry simulations may be incorporated within generic ecosystem simulations. In this context, artificial evolution may be employed to select agent parameters producing general patterns that may be validated against ecological field data, without regard for the behaviours of particular species or habitats. This would allow for studies of the generic properties of ecosystems.

Secondly, artificial evolution may be used to select parameters for pattern-oriented modelling from the level of specific species up to the level of specific ecosystems. Once a set of patterns has been matched, the evolution algorithm can be disabled and the ecosystem simulation may be used to answer questions concerning that specific ecosystem over sub-evolutionary time periods (or over longer periods, disregarding the effects of evolution).

By adopting such approaches it is possible to extend the range of questions that may be answered by ecosystem simulations for both Ecology, by locating parameters that match field data, and A-Life, by answering questions of ecological relevance whilst permitting exploration of the general properties of ecosystem behaviour outside the familiar domain of evolution.

Before investigating the application of ideas from Ecology and A-Life to the construction of virtual ecosystems, we shall give a brief overview of significant and relevant stages in the development of these fields.

Ecology and the Ecosystem

Ernst Haeckel coined the term *ecology* in *Generelle Morphologie der Organismen* (1866) to give form to the study of Natural History in the context of Darwin's ideas that organisms must struggle for survival. Ecology was to be the study of animals, their relationships amongst themselves, with plants and with the inorganic environment that affected their

[1] Some A-Life researchers will feel that there is no need for A-Life models to reflect reality in the way this paper proposes. It is true that many A-Life models are interesting regardless of their ability to represent reality. However, this paper examines how A-Life and Ecology may be of mutual benefit to one another. Hence we discuss ways of improving the correspondence between virtual and real ecosystems.

survival and reproduction (see [12], p207). Sixty years later South African ecologist Phillips, championing the view of another ecologist, Clements [13], insisted that a collection of plants and animals that had come into a harmonic relationship with one another and their habitat through succession to climax could be seen quite literally as a *Complex Organism* [14]. Phillips viewed the process of succession as a kind of ontogeny for his *Biotic Communities*, basing his ideas for the wholeness of this community on the holistic philosophy of Smuts. His aim was, in part, to unify Botany and Zoology under a new banner.

The ecologist Tansley, unhappy with Philips' argument, chimed into the debate and countered the use of *Complex Organism* by coining the word *ecosystem* in his retort, *The Use and Abuse of Vegetation Concepts and Terms* (1935). Several alternatives to *ecosystem* have been offered (e.g. *biogeocenosis*, *microcosm*, *epimorph*, *elementary landscape*, *microlandscape*, *biosystem*, *holocoen*, *biochora*, *ecotope*, *geocenosis*, *facies*, *epifacies*, *diatope* and *bioecos* [15]), each with a slightly different slant. However in the UK, Europe, Australia, the USA and many other research communities, Tansley's term and its designated focus have stuck [12]. This is true not only in science but also in politics, philosophy and even in marketing and popular culture.

Tansley's aim for the term was to give expression to a physical system that could legitimately take its place alongside those studied by Physics. Its components were animals, plants *and* abiotic material. He called attention to the significance of the exchange of materials and energy between organisms and the abiotic environment.

It is worth noting that Tansley was not concerned with Systems Theory, a field that came to the fore only after WWII. However his term's natural fit to this mould may be a part of the reason why the idea gathered popularity in the post-war years. The preference for *ecosystem* by U.S. ecologist Odum in the editions of his textbook *Fundamentals of Ecology* played a significant role in the term's post-war success also. He refined his definition for the term across the three editions of his text. In the third (1971) he wrote, "Any unit that includes all of the organisms… in a given area interacting with the physical environment so that a flow of energy leads to clearly defined trophic structure, biotic diversity, and material cycles (i.e. exchange of materials between living and non-living parts) within the system is an ecological system or ecosystem" [16], and established for a generation of ecologists the importance of the relationships between the Earth's biotic and abiotic components and the processes by which they exchange materials and energy.

Artificial Life and the (Virtual) Ecosystem

When building models we necessarily abstract away detail and represent only what we believe is responsible for determining the behaviour we wish to study or predict. Both the subjects of study and the decisions made regarding the level of abstraction can enlighten us about the different perspectives ecosystem modellers have adopted.

Langton's call for research to explore "life-as-it-could-be" was in keeping with John von Neumann's original interest in abstract self-reproducing computational systems. The idea that life might be some property of form, independent of matter, however much it is debated philosophically, has set the stage for explorations into virtual ecosystems within the field of A-Life that are more often generic than representative of particular organisms, species or their abiotic habitats. Below we shall discuss several of the early A-Life virtual ecosystems to highlight the ways in which this has been evident. The proffered research interests of these systems' creators parallel those from *Systems Ecology*, "The goal of Systems Ecology is an ecosystem phenomenology that does not necessarily require detailed information about individual species" [17]. Ray's *Tierra* [4], Yaeger's *Polyworld* [18], Holland's *Echo* [2] and Packard's *Bugs* [3] ecosystems (to list a few from the early days) fall within this category of what might be labeled *Generic Virtual Ecosystems*. Within a mini-review of individual based models in Ecology conducted in 1999, Grimm called for similar studies of the generic properties of ecosystems by ecologists [6].

Given the early interest in artificially evolving ecosystem models by A-Life researchers (shown below), it is ironic that Systems Ecologists at around the time of A-Life's inception wrote, "In spite of some attempts to address evolution in the systems literature, evolution is not well represented or adequately incorporated" [9]. The authors continue,

> …incorporating evolutionary theories into systems ecology is an underexplored but potentially fruitful avenue. Reiners (1986) has proposed that unifying ecosystem ecology requires at least three separate but complementary theoretical frameworks: energetics, matter (stoichiometry), and some aspect of population interactions or ecosystem "connectedness".

In keeping with Reiners' view, Loehle and Pechmann argue that evolution theory can provide the third of these frameworks. As these three authors highlight, whilst it is (of course) possible to study ecosystems without simultaneous reference to all three frameworks, numerous cases stand as examples in which the interaction of ideas from all three enhances our level of understanding. We treat each of these frameworks below in the context of virtual ecosystems in A-Life and Ecology. These two fields have many simulations that address the above concerns. As far back as 1999, Grimm requested that "individual-based modelling must refer to the framework of classical theoretical ecology" [6]. We maintain that it would be beneficial to construct complete simulations that encompass all three of the (sub-)frameworks suggested by Reiners, Loehle and Pechmann. Finally, this paper explains how pattern-oriented modelling may validate these well-rounded virtual ecosystems against their real counterparts — an issue that must be addressed to satisfy the demands of Ecology, and one that A-Lifer's should take seriously if they wish even their generic models to be pertinent.

Energetics

Energetics in A-Life's Virtual Ecosystems

At least since Odum, energetics has played a significant, even defining role in Ecology. It is considered by some to be the best-developed aspect of ecosystem ecology [10]. Energy flows within an ecosystem give rise to various well-studied phenomena including trophic levels, food chains and webs,

productivities, and efficiencies. In addition, processes for energy flow are one determining factor for organism evolutionary adaptation (e.g., metabolism, organism morphology and locomotion). Ecology's focus on energetics has not been duplicated in A-Life's studies of virtual ecosystems, an issue flagged as early as 1994 by Lindgren and Nordahl in a paper that describes a similarly limited model [19]. (Also see [20] for a model explicitly modelling food webs.)

Energy enters A-Life's virtual ecosystems in different ways. In some cases it makes a "magical" appearance at a specified rate and is acquired by agents without taking the form of virtual matter at any stage (e.g., [21] in which a virtual Sun shines on the space).

In their model employing *Echo*, Forrest and Jones do not include anything that an Ecologist would recognise as an energy model [2]. As a substitute, they model different types of matter that agents must collect to persist and reproduce. This will be discussed in the following section.

In Ray's *Tierra* energy is equated with CPU cycles in which to execute instructions [4]. CPU cycles are consumed by an agent to reshape its local environment by executing instructions which may, in some cases, involve writing new instructions into daughter memory cells. The focus is not on how these CPU cycles are converted into particular types of instructions (all CPU cycles are equal, and all instructions require them in equal measure). However, improved copying efficiency can be achieved by minimising energy expenditure (CPU usage). Hence there is selection pressure acting on Tierra agents to be short and reliable.

The model described by Cooper and Ofria employs *Avida* [22] and is similar to *Tierra* in some respects, however organisms "metabolise" (see the section below on *Matter* for a discussion of the use of this term) different resources to gain benefit from them by performing computations. There is inter-agent competition for these resources, which in their model, are in limited supply. Hence, to some extent this model represents the acquisition of energy by organisms. The simulation *Cosmos* by Taylor and Hallam [23] was based on *Tierra*. Its authors explicitly note *Tierra's* problematic "energy for free" and partially rectify this in their own system by requiring *Cosmos* programs to capture energy and store it, in order to use it to perform useful work.

PolyWorld incorporates a different energy model than those listed above [18]. An initial dose of energy is allocated to newborns from their parents at birth. To persist and act, agents require an amount of energy that depends on their physiology. Agent size dictates energy storage capacity. Agents maintain this store by eating plant-like food that grows in the environment at a regulated rate, or by consuming one another. Agent bodies have a food energy value separate from their current health value. Although they may die in a fight or for lack of health, this ensures that their body provides energy to a predator or scavenger.

Even in *PolyWorld* the mechanisms by which energy is extracted from materials do not emerge from the system. They are simplified to a single, high-level, hard-coded behaviour — agents just "eat". Therefore, although the agents must optimise their use of energy, the model cannot tell us about the way in which energy is transformed from one form to another across trophic levels, from sunlight to chemical energy stored in plant or animal biomass, to heat or even to kinetic energy.

The energy models employed within EVOLVE IV [1] and the simulation framework introduced by us (Dorin and Korb) [24] aimed to facilitate a greater range of emergent properties relating to energetics than any of the models above. These authors explicitly wished to explore the impact of energetics on ecosystem behaviour and consequently settled on the provision of an artificial chemistry that lies at the heart of their simulations.

We do *not* mean to imply that simple models of energetics render a simulation useless (far from it). This simplicity however limits their utility when addressing the specific problems posed by ecologists concerning particular species in particular habitats and the manner in which ecosystems give rise to the transformation of energy from one type to another — an issue that has concerned ecologists for decades. (See [6] for a discussion of the ways and degree to which these concerns have been expressed in the individual-based models of Ecology.)

Simplified models of energetics leave questions concerning trophic levels, productivities etc. essentially untouched. Simulations that incorporate low-level mechanisms for the storage and retrieval of energy by manufacture and dissociation of various chemical bonds allow for the emergence of interactions relating to agent short-term resource collection and utilization. They also provide a means to study the evolution of these survival strategies. In addition, they allow study of modes and pathways of energy transformation at the level of the ecosystem.

Matter

Matter in A-Life's Virtual Ecosystems

Since Tansley's paper, ecosystems have been identified as entities whose significant components are biotic *and* abiotic. Odum later made it clear that the cycles of materials through ecosystems were significant in shaping them as a whole, even in defining them. This interest in biogeochemical cycles has not shaped the history of Ecosystem Ecology as closely as has energetics, but it provides a valuable means to understand ecosystems [10]. For instance: knowledge of the similarities and differences between organisms' chemical composition can inform us about their evolution and their relationships to specific habitats; an organism's ability to extract elements from complex materials to construct biomass is an important aspect of its physiology and morphology; organisms excrete waste and accumulate matter in their biomass, impacting on the relative abundance of materials in the environment and its suitability as habitat for other species. In all of these ways and many others, the properties of matter play *the* dominant role.

Some of A-Life's simulations incorporate impassable barriers [18] or explore collective construction using abiotic building material [25]. These materials play no role in the construction of agent bodies and therefore do not constitute an ecologically relevant model of matter. For a model to assist exploration in the areas listed above, it must include mechanisms for combining simple materials into more complex aggregates, and for their dissociation — an artificial chemistry. Models

lacking this facility preclude the emergence of organism strategies for biosynthesis and decay, a field of study that has frequently been addressed within Ecology. Biogeochemical cycles just haven't been the focus of A-Life's virtual ecosystems.

When A-Life writers say their agents have a model "metabolism" all many of them mean is that agents persist by decrementing an energy counter at each time step (e.g., *PolyWorld*). This is a poor model of metabolism! There is usually no transformation of matter from "food" (itself a high-level abstraction) to different types of biomass or waste. Indeed, waste is hardly ever mentioned. Exceptions to this trend have appeared, including the artificial chemistry-based model ecosystems EVOLVE IV used to examine the impact of waste accumulation on mobile agents [26], and the system we proposed (Dorin and Korb) [24], mentioned above.

Our model includes decomposition as one of many chemical transformations that emerge naturally from the artificial chemistry. Patterns of material cycling through the biota and abiotic environment likewise emerge from this simulation.

Forrest and Jones' simulation [2] allows for simple material cycling through agent bodies. Materials are collected by the agents and stored for a time before being released back into the environment when the agent dies. However these materials are not used in any way to construct complex aggregates and there is no way for decomposer agents to emerge.

Without low-level representations of the type found in A-Life's artificial chemistries (such as those surveyed in [27]), virtual ecosystems cannot tell us about the emergence of chemical cycles between the abiotic environment and the biota. As we have noted, since Odum the emergence of these cycles and their impact on ecosystem behaviour has been a significant, defining issue in Ecology. In addition, as discussed in the previous section, the mechanism by which organisms store and extract energy is through the transformation of matter. Hence, the two birds of matter and energy can be killed with the single stone of artificial chemistry. Without this stone, A-Life's virtual ecosystems realise only a (small) fraction of their potential.

Evolution

Evolution in A-Life's Virtual Ecosystems

If there are two widespread traits of A-Life's virtual ecosystems in addition to a lack of interest in energy and matter transformation, perhaps they are these: (i) a focus on studying the general properties of the evolutionary process itself, (ii) the use of the artificial evolutionary process to select parameters that allow agents to meet the requirements of existence and replication in a dynamic virtual environment.

Ray's *Tierra* was constructed, "with hand-crafted organisms already capable of replication and open-ended evolution... to generate increasing diversity and complexity" [4]. The evolutionary process was needed to drive change and maintain order in *Tierra* as much as it was the focus of the researcher's attention. Ray's system was intended to model replicators and their interactions within the digital medium, not as representatives of particular biological organisms or their habitats. Ten years later, Ray's focus continued to be the evolutionary process, his aim being to, "use organic life as a source of ideas on how to create a richer evolutionary process in the digital medium" [28].

Yaeger states as one of his aims for *PolyWorld*, "create artificial life that is as close as possible to real life, by combining as many critical components of real life as possible in an artificial system" [18]. Like Ray, he includes in his essential traits of living systems their ability to replicate and evolve. Although his study is a level of abstraction above *Tierra* (his virtual organisms have behavioural primitives including the ability to eat, mate, move, see etc.), and is more literal in its representation of agents and space, still the system is generic — it aims to tell the researcher about the evolution of behaviour appropriate to *PolyWorld* agents in the virtual world, rather than about the behaviour of specific species in real habitats.

Packard's *Bugs* [3] are similar in some respects to agents from *PolyWorld*. The independent bug agents roam a space and search for food that allows them to sustain themselves and reproduce.

Forrest and Jones write of the *Echo* platform, "Echo is intended to capture important generic properties of ecological systems, and not necessarily to model any particular ecology in detail" [2]. They go on to explain that if their software could be correctly validated, it could inform us about the impact of evolution on ecosystem behaviour. For them this is perhaps the most important contribution such models may make.

There are countless much more recent examples of artificial evolution implemented in virtual ecosystems to study the evolution of altruism, parental investment, group selection, aging, epidemiology etc. (e.g., [21, 29, 30]). This path is well trodden by A-Life researchers.

Why incorporate artificial evolution?

In addition to the obvious and widely stated interest in studying the evolutionary process itself, the difficulty in writing and manually parameterising software to establish ongoing, dynamic but stable relationships between software agents in a virtual space motivates the simulation of evolution. The Genetic Algorithm (GA) has seen wide application for the solution of optimization problems and this motivates its use in virtual ecosystems also. Handcrafted systems may operate dynamically for a while, however adjustment of parameter values is needed to keep agents from extinction and virtual ecosystems from collapse.

Returning to a familiar example, Yaeger implemented in *PolyWorld* a mode to kick-start the evolutionary process and overcome the problem of parameter specification for creating viable agents. At first he imposes an external fitness function to optimise agents until they evolve parameters to successfully sustain themselves and replicate independently.

Similarly, Ray seeds *Tierra* with a hand-designed replicator. Evolution then takes over, altering the copying algorithm and enhancing its efficiency. Again, evolution optimises the agents in the current, dynamic environment.

The implications of employing digital evolution

Artificial evolution is currently limited in its ability to accurately mimic the detailed behaviour of real evolution. The

application of artificial evolution as an instigator of agent change in A-Life simulations, without matching agent parameters to field data, further ensures that the simulations of ecosystems remain generic. In contrast, especially during the period to 1999 [6], Ecology's agent-based models have tended to be non-evolutionary and focused on modelling specific aspects of specific species (for instance the behaviour and resulting distribution of lynx [31]). Typical A-Life models are of limited relevance to this form of Ecology.[2] In the context of conventional ecological modelling, evolutionary algorithms are not sufficiently true-to-life to replicate the evolutionary path that gave rise to specific creatures. Additionally, in many simulations the agent and environmental models capture only a few traits of the real systems they represent. All of these simplifications to models of highly sensitive, non-linear systems ensure that whilst digital evolution may optimise agent parameters for survival in virtual environments, the values of these parameters will quite likely *not* reflect values that may be obtained from field data. In fact, the parameters often do not reflect any aspect of any real system [2]. Obviously, this does not render existing models useless — it merely renders them inapplicable where ecological problems need to be addressed concerning the evolution of specific species in specific habitats.

Pattern-Oriented Modelling

Artificial Evolution for Pattern-Oriented Modelling

Despite the tenuous connections between the specifics of artificial and real evolution, we suggest that the former may be included in ecological models for the benefit of Ecology and A-Life alike, by assisting in the process of *Pattern-Oriented Modelling* (POM) [11, 32]. POM is a general strategy for developing, validating, and parameterising ecological models: "In POM, we explicitly follow the basic research program of science: the explanation of observed patterns. Patterns are defining characteristics of a system and often, therefore, indicators of essential underlying processes and structures. Patterns contain information on the internal organization of a system, but in a *coded* form. The purpose of POM is to *decode* this information" ([11], p. 987). Patterns can thus also be viewed as *regularities* or *signals*; in economics, often the similar notion of *stylized facts* is used [33].

Multiple patterns observed in real systems are used in POM in three ways: they indicate which state variables a model should have so that in principle the same patterns can emerge in the model; they are used to select the most appropriate alternative sub-model representing a certain process; and they can be used to determine entire sets of unknown parameters (i.e. for *inverse modelling*).

Basically, POM reminds system modellers of the very basic principle of science: if we focus on single patterns to build a model we risk producing one that provides a good fit to data for the wrong reasons (i.e., mechanisms) instead of a model that is structurally realistic and captures the essentials of the system's organisation. Thus, the more pattern matches between a model and field data and the more variety these exhibit, the more certain the correspondence between model and reality.

Ideally, simulation patterns match at multiple levels of organization to the system being modelled. For example, patterns might be matched at the level of an individual organism's behaviour, the dynamics of a population, and at the level of an entire ecosystem. Commonly cited A-Life models (for instance models of animal flocking such as Reynolds' *boids* [34]) do not seek this *multi-level* pattern match since their aim is to locate *any* parameters or agent behaviours that produce a desired global behaviour. Yet if A-Life models are to be relevant to Ecology, Ecologists rightfully insist that multi-level pattern matches are required for validation.

Pattern-Oriented Modelling of Evolutionary Processes

As described earlier, artificial evolution may optimise agent parameters in virtual ecosystems enabling agents to survive and replicate. If these models also include low-level representations of energetics and matter, the simulations may inform us about *general* principles of ecosystem behaviour of relevance to Ecology, including, for example, how *physical ecosystem engineering*[3] impacts on habitats, the number of niches, the number of trophic levels in an ecosystem, species diversity, ecosystem resilience and stability etc. over evolutionary time periods. Data regarding the impact of physical ecosystem engineers on ecosystems is available [35-37] and could form the basis for pattern-oriented modelling, even if the specifics of species fell beneath the level of abstraction of the model. To the authors' knowledge no generic simulations exploring these properties of ecosystems over evolutionary time periods have yet been devised. In fact, we believe that as yet there are no simulations investigating the emergence of ecosystem engineers at all. Do they evolve readily and under what circumstances? Is there a basic organizational property of ecosystems that requires them?

This domain is ideally suited to a simulation of the form we describe, since it requires detailed models of the transformation of the abiotic and biotic environment and its impact on the evolution of biota.

[2] In Systems Ecology (as in A-Life) the need to model specific species and habitats does not apply (see above). Grimm has also argued that the general principles of ecosystem behaviour *are* (or at least *should be*) of interest to Ecologists [6].

[3] To varying extents, all organisms are physical ecosystem engineers. Physical ecosystem engineers physically alter the biotic or abiotic environment and thereby control or modulate the availability of resources to (or forces acting on) other organisms. These physical changes destroy, maintain or create habitat for other organisms [36]. Their presence is often a key factor in ecosystem behaviour. A tree is an example of a significant physical ecosystem engineer: it provides habitat for mosses, insects and birds; its roots trap soil and leaf matter, altering the impact of wind and water erosion; its branches harbour larvae or tadpoles within pools etc. Coral produces reefs, wombats dig holes, lyrebirds and blackbirds sift leaf litter. These species (and humans!) are physical ecosystem engineers that have a large impact on organisms around them.

Pattern-Oriented Modelling with Evolutionary Parameter Optimisation

POM has already been applied within ecology to validate models of specific species against their real counterparts. For instance, the technique matched the behaviour of a specific species (lynx) to an agent-based model [31]. Whereas in the POM of evolutionary processes we proposed to ignore the specific behaviours at the level of the individual organism (these fall beneath the level of detail incorporated into a generic simulation), in this case the simulation is intended to be accurate in its specific detail from the level of the individual agent up to the level of the ecosystem. In this second method, the aim is not just to find general parameters that create a virtual ecosystem that behaves like a real one, but also to locate agent parameters matching observed organism patterns so that the model correctly reflects the behaviour of real species.

At the point where parameters for agents give rise to realistic patterns of behaviour, and this has been shown to give rise to patterns matching the behaviour of the ecosystem, artificial evolution can be *disengaged*. It has served its purpose as a means for automatically parameterising the model. The simulation can then be used to answer ecological questions by researchers who wish (for whatever reason!) to disregard evolutionary effects.

If this approach is to succeed, the pattern-matching process must be automated so that evolution (which acts *within* the virtual ecosystem in this instance) can play its role as an optimization algorithm. Field data at various ecological levels must be available for this process to work. We identified (above) studies of physical ecosystem engineers as a source of data for such an implementation.

A difficulty may arise when attempting to match the behaviour of several simulated species in the virtual ecosystem simultaneously, and without allowing the dynamical whole of which they are a part to collapse. A mode in which an externally defined fitness function operates (akin to that discussed above for *PolyWorld*) may assist when virtual ecosystem collapse appears imminent, by ensuring population levels, diversity or some other measure of virtual ecosystem state does not fall outside a specified range.

There is evidence in the ecological literature that a GA is capable of correctly parameterising a model based on field data. Strand et al have successfully used this technique to generate parameters for a model of fish behaviour [38], suggesting not only the feasibility of the approach, but also its acceptability to Ecologists.

Discussion and Conclusions

We do not expect there to be significant debate concerning our position on the abundance or value of evolution-focussed A-Life virtual ecosystems. However we have also argued the position that while some examples of individual-based simulations incorporating energetics and matter have been created, this area is under-explored in A-Life's virtual ecosystems, even whilst it may be a component of the field's artificial chemistry models. Our overview of several early models from the A-Life literature highlights the initial trends in the field but as we have seen, even the updates to these models and many recent virtual ecosystems are blinkered when it comes to energy and matter. This state of affairs must be rectified if our models are to be of relevance to Ecology, as it has been traditionally understood, and as it is now being practiced.

By incorporating low-level models of the three frameworks of energetics, matter and evolution into virtual ecosystems we open significant pathways for ecologically relevant research in A-Life, particularly in domains where the interactions of the abiotic and biotic play a role. To our minds, and those of Reiners [10], Loehle and Pechmann [9], the interactions of these frameworks is a core concern.

We have indicated two strategies for employing these frameworks within virtual ecosystems. The first of these requires the application of the evolutionary process in much the same way as it has been traditionally applied within A-Life: as a means to dynamically adjust agent parameter values to support their viability and reproduction within the virtual environment. These simulations will allow us to determine generic properties of ecosystems only if the models are validated against field data employing a technique such as pattern-oriented modelling. This must ensure multi-level correspondence between simulation and reality, even if the level of abstraction of the model does not reach down to the detailed simulation of specific real species.

The second approach we suggest employs artificial evolution to match simulation patterns against data gathered from the level of specific species up to data concerning specific ecosystems. Once the parameters of the system have been optimised so as to reproduce the patterns observed in field data, the evolution algorithm is turned off. The model may then be employed to answer questions relating to the specific ecosystem and species that it represents. Unfortunately it may not then be used to study the evolution of these specific species in specific environments. This is a shortcoming of the artificial evolution algorithm (it does not model real evolution in detail) that would be worth overcoming.

It remains to be seen whether artificial evolution will be suited to pattern matching across several levels of an entire dynamical system consisting of hundreds of components. However, the widespread success of the algorithm in the solution of complex optimization problems bodes well for the approach. Additionally, the technique has already been employed successfully in Ecology to match a single species' behaviour to that of software agents and we have indicated some preliminary strategies that may assist where multiple species are interacting in a simulation. The value of correctly parameterised, detailed models of ecosystems ensures the importance of attacking this problem in the future.

References

1. Brewster, J.J. and M. Conrad. *Computer Experiments on the Development of Niche Specialization in an Artificial Ecosystem*. In *Congress on Evolutionary Computation*. 1999. Washington DC: IEEE: p. 444-451.
2. Forrest, S. and T. Jones. *Modelling Adaptive Systems with Echo*. In *Complex Systems: Mechanisms of Adaptation*. 1994: IOS Press: p. 3-21.

3. Packard, N.H. *Intrinsic Adpatation in a Simple Model for Evolution*. In *Artificial Life, SFI Studies in the Sciences of Complexity*. 1988: Addison-Wesley: p. 141-155.
4. Ray, T.S. *An approach to the synthesis of life.* In *Artificial Life II*. 1990. Santa Fe, New Mexico: Addison Wesley: p. 371-408.
5. Conrad, M. and H.H. Pattee, *Evolution Experiments with an Artificial Ecosystem.* Journal of Theoretical Biology, 1970. **28**: p. 393-409.
6. Grimm, V., *Ten years of individual-based modelling in ecology: what have we learned and what could we learn in the future?* Ecological Modelling, 1999. **115**: p. 129-148.
7. DeAngelis, D.L. and W.M. Mooij, *Individual-based modelling of ecological and evolutionary processes.* Annu. Rev. Ecol. Evol. Syst., 2005. **36**: p. 147-168.
8. Grimm, V. and S.F. Railsback, *Individual-based Modeling and Ecology*. Princeton Series in Theoretical and Computational Biology. 2005: Princeton University Press. 428.
9. Loehle, C. and J.H.K. Pechmann, *Evolution: The Missing Ingredient in Systems Ecology.* American Naturalist, 1988. **132**(6): p. 884-899.
10. Reiners, W.A., *Complementary Models for Ecosystems.* The American Naturalist, 1986. **127**(1): p. 59-73.
11. Grimm, V., et al., *Pattern-Oriented Modelling of Agent-Based Complex Systems: Lessons from Ecology.* Science, 2005. **310**: p. 987-991.
12. Golley, F.B., *A History of the Ecosystem Concept in Ecology: more than the sum of the parts*. 1993, New Haven and London: Yale University Press. 254.
13. Clements, F.E., *Preface*, in *Plant Succession: An Analysis of the Development of Vegetation*. 1916, Carnegie Institution of Washington: Washington D.C. p. 1-7.
14. Phillips, J., *The Biotic Community.* The Journal of Ecology, 1931. **19**(1): p. 1-24.
15. Sukachev, V. and N. Dylis, *Fundamentals of Forest Biogeocoenology*. 1964, Edinburgh and London: Oliver and Boyd.
16. Bergandi, D., *Reductionist Holism: An oxymoron or a philosophical chimera of Eugene Odum's Systems Ecology?* Ludus Vitalis: Journal of Philosophy and Life Sciences 3, 1995: p. 145-178.
17. Ulanowicz, R.E., *Growth and development: ecosystems phenomenology*. 1986, New York: Springer-Verlag. 203.
18. Yaeger, L., *Computational Genetics, Physiology, Metabolism, Neural Systems, Learning, Vision and Behavior or Polyworld: Life in a New Context*, in *Artificial Life III*. 1992, Addison-Wesley: City. p. 263-298.
19. Lindgren, K. and M.G. Nordahl, *Cooperation and Community Structure in Artificial Ecosystems.* Artificial Life, 1994. **1**(1/2): p. 15-37.
20. Lindgren, K. and M.G. Nordahl. *Artificial Food Webs*. In *Artificial Life III*. 1994: Addison-Wesley: p. 73-104.
21. Dorin, A. *A Co-Evolutionary Epidemiological Model for Artificial Life and Death*. In *8th European Conference on Artificial Life*. 2005. Canterbury, UK: Springer: p. 775-784.
22. Cooper, T.F. and C. Ofria. *Evolution of Stable Ecosystems in Populations of Digital Organisms*. In *Artificial Life VIII*. 2003. Sydney, Australia: MIT Press: p. 227-232.
23. Taylor, T. and J. Hallam. *Studying Evolution with Self-Replicating Computer Programs*. In *Fourth European Conference on Artificial Life*. 1997: MIT Press: p. 550-559.
24. Dorin, A. and K. Korb. *Building Artificial Ecosystems from Artificial Chemistry*. In *9th European Conference on Artificial Life*. 2007. Lisbon: Springer-Verlag: p. 103-112.
25. Theraulaz, G. and E. Bonabeau, *Modelling the Collective Building of Complex Architectures in Social Insects with Lattice Swarms.* Journal of Theoretical Biology, 1995. **177**(4): p. 381-400.
26. Brewster, J.J., R.G. Reynolds, and M.A. Brockmeyer, *Not In My Backyard: A simulation of the effects of agent mobility on environmental poisoning*, in *Proceedings of the 2002 Congress on Evolutionary Computing*. 2002: City. p. 849-854.
27. Dittrich, P., J. Ziegler, and W. Banzhaf, *Artificial Chemistries - A Review.* Artificial Life, 2001. **7**(3): p. 225-276.
28. Ray, T.S. and J.F. Hart. *Evolution of Differentiation in Multithreaded Digital Organisms*. In *Artificial Life VII, Proceedings of the Seventh International Conference on Artificial Life*. 2000: MIT Press: p. 132-140.
29. Mascaro, S., K.B. Korb, and A.E. Nicholson. *ALife Investigation of Parental Investment in Reproductive Strategies*. In *Eighth International Conference on Artificial Life (ALife VIII)*. 2003: MIT Press: p. 358-361.
30. Woodberry, O., K.B. Korb, and A.E. Nicholson, *The Evolution of Aging*, in *Proceedings of the Australian Conference on Artificial Life (ACAL 2005)*. 2005: City. p. 319-333.
31. Kramer-Schadt, S., et al., *Patterns for parameters in simulation models.* Ecological Modelling, 2007. **204**: p. 553-556.
32. Grimm, V. and U. Berger, *Seeing the forest for the trees, and vice versa: pattern-oriented ecological modelling*, in *Handbook of scaling methods in aquatic ecology: measurement, analysis, simulation*, L. Seuront and P.G. Strutton, Editors. 2003, CRC Press: Boca Raton. p. 411-428.
33. Kaldor, N., *Capital Accumulation and Economic Growth*, in *The Theory of Capital, Reprint*, F. Lutz and D. Hague, Editors. 1961/1968, Macmillan: London. p. 177-222.
34. Reynolds, C.W., *Flocks, Herds and Schools: A Distributed Behavioural Model.* Proceedings of SIGGRAPH '87, 1987. **21**(4): p. 25-34.
35. Badano, E.I. and L.A. Cavieres, *Ecosystem engineering across ecosystems: do engineer species sharing common*

features have generalized or idiosyncratic effects on species diversity? Journal of Biogeography, 2006. **33**: p. 304-313.
36. Gutiérrez, J.L. and C.G. Jones, *Physical Ecosystem Engineers as Agents of Biogeochemical Heterogeneity.* BioScience, 2006. **56**(3): p. 227-237.
37. Jones, C.G., J.H. Lawton, and M. Shachak, *Positive and negative effects of organisms as physical ecosystem engineers.* Ecology, 1997. **78**(7): p. 1946-1957.
38. Strand, E., G. Huse, and J. Giske, *Artificial Evolution of Life History and Behavior.* The American Naturalist, 2002. **159**(6): p. 624-644.

Programmable Architectures That Are Complex and Self-Organized: From Morphogenesis to Engineering

René Doursat[1]

[1]Institut des Systèmes Complexes, CREA, CNRS & Ecole Polytechnique, Paris, France
doursat@shs.polytechnique.fr

Abstract

Outside biological and social systems, natural pattern formation is essentially "simple" and random, whereas complicated structures are the product of human design. So far, the only self-organized (undesigned) *and* complex morphologies that we know are biological organisms and some agent societies. Can we export their principles of decentralization, self-repair and evolution to our machines, networks and other artificial constructions? In particular, can an "embryomorphic" engineering approach inspired by evo-devo solve the paradoxical challenge of planning autonomous systems? In this work, I wish to better understand and reproduce complex morphogenesis by investigating and combining its three fundamental ingredients: *self-assembly* and *pattern formation* under *genetic regulation*. The model I propose can be equivalently construed as (a) *moving* cellular automata, in which cell rearrangement is influenced by the pattern they form, or (b) *heterogeneous* collective motion, in which swarm agents differentiate into patterns according to their location. It offers a theoretical framework for exploring the causal and programmable link from genotype to phenotype.

Introduction

Faced with a rapid growth in size and complexity of computer systems, whether hardware, software or networks, engineers are gradually led to rethink ICT in terms of *complex systems*. In particular, as the field of Artificial Life demonstrates, it is compelling and fruitful to seek inspiration from biological and social examples such as organism development, neural networks, insect colonies, or human communities. Understanding natural emergence should help design a new generation of artificial complex systems by importing into our machines highly desirable properties that are still largely absent from traditional engineering: *decentralization*, *autonomy* (self-organization, homeostasis) and *adaptation* (learning, evolution). Simply formulated, the new challenge is: How can we make a multitude of agents get together and do something useful without placing them by hand? Emergent engineering will be less about direct design than developmental and evolutionary *meta-design*. Changing from micro-managers to law-makers, future engineers would "step back" from their creation and only set generic conditions for systems to self-assemble and evolve, instead of building them directly.

Darwinian evolution consists of random variation followed by non-random selection. Concerning evolutionary engineering, the present work stresses the importance of establishing fundamental laws of *developmental variations* before these can be selected on the evolutionary time scale [20]. Understanding variation by comparing the development of different species is the concern of "evo-devo", a fast growing field of biology [4, 12]. The genotype-phenotype link cannot remain an abstraction if we want to unravel the generative laws of development and evolution—and ultimately transfer them to artificial self-organized systems. Moreover, fine-grain, hyperdistributed architectures (i.e., many light-weight agents, as opposed to a few heavy-weight agents) such as multicellular organisms might be in a unique position to provide the "solution-rich" space needed for successful selection.

Within this framework, the goal of this article is to understand and model the *self-organization of complex morphologies*. To this aim, it proposes to combine three ingredients: morphogenetic *self-assembly* (SA) and *pattern formation* (PF) under the control of *non-random, structured genetic regulation* (GR) stored inside each agent of a swarm.

Toward Self-Organized Complex Architectures

Non-biological/social self-organization exhibits "simple" patterns. Self-organized systems of physical-chemical matter generally form random, repetitive spatial patterns: ripples in sand dunes, convection cells in hot liquids, spots and stripes in reaction-diffusion solutions à la Turing [1], etc. Despite a huge and fascinating diversity of pattern formation behaviors across many scales and substrates, emergent structures at the macroscopic level are fairly regular, essentially consisting of repeated motifs. They display a statistical uniformity and relative "poorness of information" similar to textures. Moreover, most of these pattern formation phenomena rely on instabilities and amplification of fluctuations to generate order. Because of this inherent stochasticity, the number and position of emerging entities (spots, stripes, etc.) are generally unpredictable. The only self-organized systems able to create truly complex structures are biological organisms and agent societies (e.g., termite mounds, cities, markets, Internet).

Non-biological/social complex structures are deliberately designed. Outside biology and agent societies, most complicated structures made of segments and parts arranged in specific ways are the product of direct human control: computers, cars, buildings, etc. Contrary to physical pattern formation systems, human constructions are fundamentally reproducible and programmable. They are made of a diversity of modules that are statistically heterogeneous and information-rich.

However, the cost of such complexity is *heteronomy*: these structures rely entirely on centralized design and deterministic planning at the macroscopic level, imposing order from the outside. Again, the only complex forms that are also truly undesigned, i.e., naturally emergent, are biological and social.

Re-creating structures that are complex *and* self-organized. Compared to physical pattern formation, the unique feature of biological and social morphogenesis is that it relies on *agents* (cells, insects, computers, humans) that carry *sophisticated instruction sets* (DNA, stigmergy, program, cognition). This functional information endows the agents with a repertoire of non-trivial behaviors, vastly superior to units of inert matter. Most importantly, it opens the door to agent *diversity* through differentiation and evolution, which in turn allows rich combinations and recombinations of agents into modules and hierarchical constructions. Therefore, focusing for now on multicellular organisms, can we strive toward a new kind of morphogenesis-inspired or "embryomorphic" engineering? It is the purpose of this work to show how genetic-like regulation at the agent level can be used to control an artificial process of complex self-organization.

Integrating Self-Assembly and Pattern Formation Under Agent-Level Genetic Regulation

In this modeling work, I propose that, from an abstract viewpoint, self-organized complex morphologies such as biological development can be best understood as a combination of *self-assembly* (SA) and *pattern formation* (PF). To take an artistic metaphor, this would be similar to mixing "self-sculpting" and "self-painting" in one composition [6]. On the one hand, embryogenesis can be seen as a "self-made puzzle", i.e., a spontaneous sculpting process in which the puzzle pieces (the cells) reshape and reassemble themselves dynamically. On the other hand, it can also be seen as a "deformable screen", i.e., a spontaneous painting process where color strokes (gene expression levels) modify each other on top of an irregular and shifting geometry.

Self-assembly (SA). Research in natural or artificial *self-assembling* systems, mostly following "molecular soup" models, has traditionally focused on pre-existing components endowed with fixed shapes. Biological development, by contrast, dynamically creates new cells that acquire selective adhesion properties through differentiation induced by their neighborhood. I propose here a model of self-organized swarm in which the agents undergo dynamical *positioning* from neighbor forces, dynamical *creation* by division, and dynamical *reshaping* by non-uniform modification of their interactions. These elementary SA behaviors are induced and controlled by agent differentiation (see PF in next subsection).

Pattern formation (PF). Pattern formation phenomena are generally construed as orderly states of activity on top of a continuous 2-D or 3-D substrate. Yet, again, the spontaneous patterning of an organism into regions of gene expression arises within a multicellular medium in perpetual expansion and reshaping. In the present model, agents undergo dynamical *differentiation* into various types and subtypes. The swarm becomes inherently *heterogeneous*, breaking up into local groups. PF activity is based on the exchange of two categories of signals within and among these groups: *positional information* (spread of gradients or signalling "counters") and *identity information* (gene-expression levels). These elementary PF behaviors are induced and controlled by agent positions (see SA in previous subsection).

Genetic regulation (GR). Finally, traditional SA and PF are often thought of in terms of *stochastic* events, i.e., collisions and fluctuations. By constrast, biological cells are not randomly mixed but pre-positioned where divisions occur (before migrating). Genetic identity regions are not randomly distributed but highly regulated in number and position. They dynamically *unfold in time*, on the basis of simple calculations and decisions carried out by each agent at every time step. Agents contain a complete *genotype G*, of which they execute only a small portion at any time, depending on their current differentiation type and input from their neighborhood.

From Biology to Engineering

This study is inherently interdisciplinary, as it closely follows biological principles at an abstract level, but does not attempt to model detailed data from real genomes or organisms. Thus, it lies at crossroads between different families of works, from developmental and systems biology to artificial life, in particular spatial computing, evolutionary programming and swarm robotics. It is an original attempt to integrate the three mechanisms of SA, PF and GR discussed above. Only few previous theoretical models of biological development or bio-inspired artificial life systems have combined them in various ways. The evo-devo works of [11, 17], or [19, 16] with lesser morphogenetic abilities, are among these notable achievements. Other interesting studies have explored the combination of two out of three: SA and PF, no GR—self-assembly based on cell adhesion and signalling pattern formation, but using only predefined cell types without internal genetic variables (e.g., [15]); PF and GR, no SA—non-trivial pattern formation from instruction-driven intercellular signalling, but on a fixed lattice without self-assembling motion (e.g., [7]); SA and GR, no PF—heterogeneous swarms of genetically programmed, self-assembling particles, but in empty space without mutual differentiation signals (e.g., [18]).

Model

This section presents a computational model, with illustrative numerical simulations, of *programmable* and *reproducible* artificial morphogenesis. The differential properties of cells (adhesion, division) are determined by the regions of gene expression to which they belong, while at the same time these regions further expand and segment into subregions due to the self-assembly of differentiating cells. The model can be construed from two different vantage points: either (a) pattern formation on *moving* cellular automata, in which the cells spatially rearrange under the influence of their activity pattern, or (b) collective motion in a *heterogeneous* swarm, in which the agents gradually differentiate and modify their interactions according to their positions and the regions they form.

First, the motion of a homogeneous swarm (pure SA) and the patterning by gradient propagation on a fixed swarm (pure PF) are introduced separately. Then, these two components are combined to form reproducible growing patterns

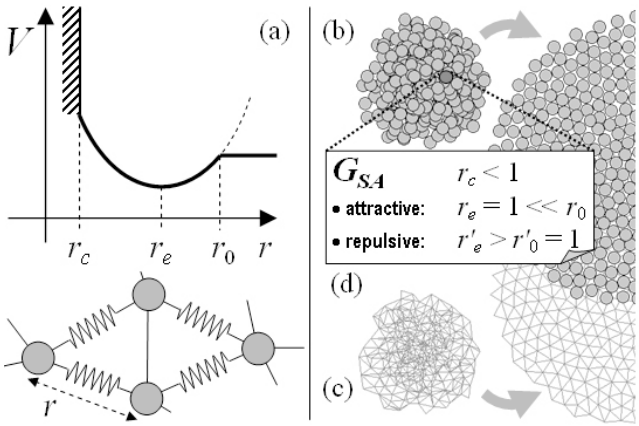


Figure 1: Deployment of a homogeneous swarm (SA; see text). (a) Agent-level interaction potential V similar to elastic springs. (b) Relaxation of a 400-agent swarm from an initially compressed state (only half of the end state shown). (c) Same swarm without nodes, showing interaction mesh obtained by Delaunay triangulation and pruning of edges longer than r_0. (d) Genetic SA parameters inside every agent (here, attractive mode only).

(SA + PF). The genetic program controlling these arrangements inside every agent is also explained. Finally, this combination is repeated as modules ($SA^{(k)} + PF^{(k)}$) inside a larger, heterogeneous system to create complex morphologies by recursive refinement of details. All swarm formations presented in the figures result from actual simulations (in Java).

Deployment of a Homogeneous Swarm (SA)

Exploring the principles of multicellular development as an inspiration for self-organized artificial systems, the model incorporates two major aspects of cellular biomechanics: cell *adhesion*, in the form of elastic rearrangement, and cell *division* (addressed in a later subsection). Schematically, a self-assembling swarm is composed of agents or "puzzle pieces" described by their *geometrical variables*, *motion dynamics* and *interaction network*. In 2-D, each agent A has a position $\mathbf{r}_A = (x_A, y_A)$, velocity $d\mathbf{r}_A/dt$ and shape at a certain orientation. In this model of swarm dynamics, agent shapes actually represent mutual adhesion affinities implemented by local *interaction potentials* $V(\mathbf{r}_A, \mathbf{r}_B)$ around the agents. Thus, swarm motion is caused by agent-centered forces derived from V. Here, simple discs of diameter r_c are used, creating isotropic potentials $V(\| \mathbf{r}_A - \mathbf{r}_B \|) = V(r_{AB})$ in their vicinity. Similar to other collective motion models [21], $V(r)$ consists of three parts (Fig. 1a): (i) infinite repulsion for $r < r_c$ representing non-deformable particles, (ii) elastic (quadratic) attraction around an equilibrium distance r_e representing the resting length of small springs, and (iii) flat potential for $r > r_0$ representing the absence of force beyond a certain "visibility" horizon. Agents interact through a dynamic network topology that depends on their positions. Edges $A \rightarrow B$ are created and removed according to a given connectivity scheme, e.g., circular scope ($r_{AB} < r_0$), k-nearest neighbors, Delaunay triangulation, or a combination thereof. For low values of r_0 and $k = 6$, these schemes are roughly equivalent. Starting from a compressed swarm, agents quickly relax to a resting state, in which they tend to form quasi-regular triangular meshes (Fig. 1c). Experiments in the rest of this paper are based on the Delaunay triangulation with additional pruning for $r > r_0$.

Thus, at this stage, each agent in the swarm possesses (a) fixed "genetic" SA parameters, denoted by G_{SA} (Fig. 1d), and (b) dynamic SA state variables—its position and connections with other agents. The genetic parameters consist of V's parameters r_c, r_e and r_0. Typically, $r_c < r_e = 1 << r_0$ for attractive potentials, but V can also become neutral or repelling if $r_0 \le r_e$. Repulsion will be later used between different types of agents (see last subsection about modular development).

Propagation of Positional Information (PF-I)

Pieces of a jigsaw puzzle are defined not only by their position and shape but also by the "image" they carry. In the self-organized swarm, this translates into state variables inside each agent that determine their PF activity. The present model distinguishes between two kinds of PF-specific state variables: *gradient* variables (PF-I) and *pattern* variables (PF-II), addressed in the next subsection. Gradient values propagate from neighbor to neighbor and establish *positional information* across the swarm [23]. Pattern values are calculated from the gradient values and create different *agent types*, which in turn affect the SA behavior (see SA + PF integration below).

Thus, agents not only interact mechanically according to the SA forces, but also exchange activity signals on the same graph edges according to PF rules. Halting the SA dynamics for now, let us consider a fixed swarm such as the one produced by Fig. 1b. Assume that one agent denoted by W contains a counter variable $n_W = 0$ and passes messages to its neighbors, instructing them to set their own n_W to 1. These neighbors in turn instruct their neighbors to set n_W to 2, and so on. To avoid back-propagation effects, the actual value of n_W remains the minimum of all received values. The result is a roughly circular wave pattern centered on W (Fig. 2a), which represents a discrete approximation of a heat-like diffusive gradient in continuous space. Discrete counter increments are also the method of choice for spreading positional informa-

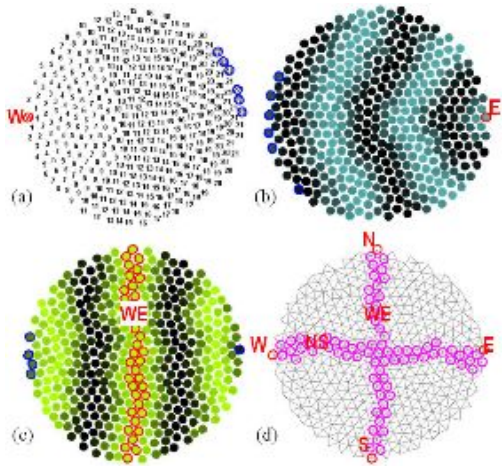


Figure 2: Propagation of positional information (PF-I; see text). (a) Circular gradient of counters originating from source agent W in red (gradient ends in blue). (b) Opposite gradient coming from antipode agent E viewed by a cyclic color map. (c) Planar gradient triggered by agents WE, whose W and E counters are equal ±1. (d) Full coordinate compass on mesh, with midlines.

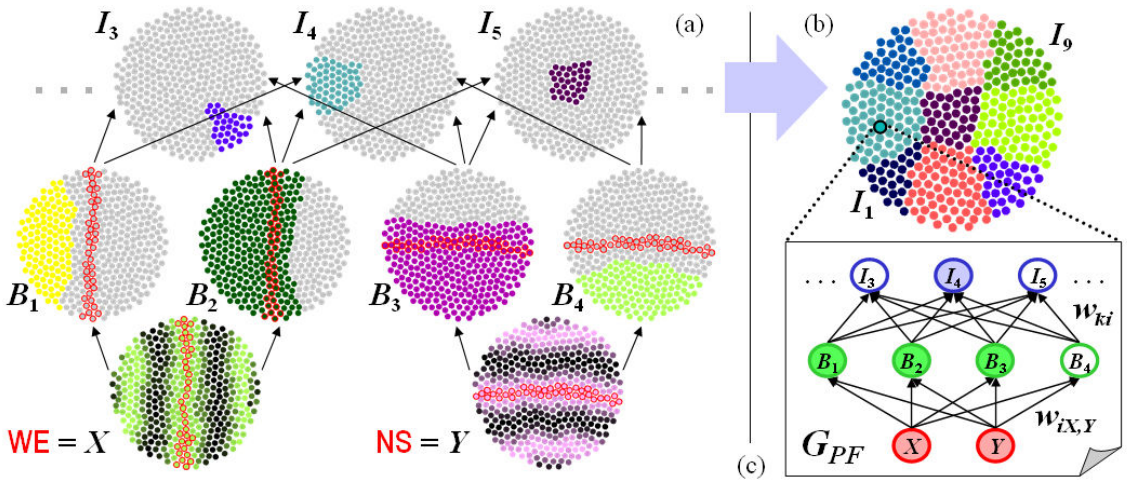


Figure 3: Programmed patterning (PF-II; see text). (a) The same swarm in different colormaps to visualize the agents' internal patterning variables X, Y, B_i and I_k (virtual equivalent of *in situ* hybridization in biology). (b) Consolidated view of all identity regions I_k for $k = 1...9$. (c) Gene regulatory network used by each agent to calculate its expression levels, here: $B_1 = \sigma(1/3 - X)$, $B_3 = \sigma(2/3 - Y)$, $I_4 = B_1B_3(1 - B_4)$, etc.

tion in amorphous and spatial computing systems [7, 16, 2]. In the present model, the role of source W can be transferred to another agent, thereby shifting the entire gradient landscape in successive corrective waves, as agents continually communicate with each other to adjust their counters.

In parallel to W, assume that another gradient propagates from a source agent E located at a certain distance from W, e.g., at two antipodes of the swarm (Fig. 2a,b). All agents have now two counters, n_W and n_E. In the example of Fig. 2, $n_W = 0$ and $n_E = 22$ in W and conversely in E. Together, these two gradients define a midline across the swarm, denoted by WE. It is the subset of agents that are equidistant from W and E, i.e., $n_W \approx n_E$, for example $|n_W - n_E| \leq 1$ (see, e.g., [16]). Agents belonging to the WE midline become in turn the sources of a new gradient, creating a planar wave of n_{WE} counters that propagates symmetrically toward W and E (Fig. 2c; $n_{WE} = 0$ in WE, and 11 in W or E). Finally, assume that two other gradient sources, N and S, are located at two other antipodes of the swarm on the WE midline. This creates a second midline NS perpendicular to WE and a second planar wave of n_{NS} counters (Fig. 2d). Each agent now has 6 counters: n_W, n_E, n_{WE}, n_N, n_S and n_{NS}. Together, they establish a 2-D *pattern coordinate system* (X, Y) in the swarm—distinct from the physical coordinates (x, y) of the SA process—for example by setting: $X = \text{sign}(n_W - n_E)n_{WE}$, and $Y = \text{sign}(n_S - n_N)n_{NS}$. To obtain normalized coordinates, each agent can also divide X and Y by local estimates of the global width w and height h of the swarm: $X' = X/w$, where $w = \max_A(n_{WE})/2 \approx n_{WE} + n_W$ for $X < 0$ and $w \approx n_{WE} + n_E$ for $X > 0$—and similarly for the vertical axis, replacing X, W, E and w by Y, S, N and h.

Naturally, the polar and equatorial locations of the four sources N, S, W and E are not imposed by hand, but are themselves the result of a self-organizing process via a feedback loop between gradients and sources. This is explained below in the subsection about SA + PF integration.

Programmed Patterning (PF-II)

On top of the coordinate system created by the gradient variables, each agent calculates another set of variables that are responsible for the swarm's patterning or "image". This process represents the emergence of *heterogeneity*, i.e., the segmentation of the swarm into *different types* of agents. In principle, any arbitrary pattern I (at the level of resolution offered by the swarm) could be programmed into the agents as a direct function of the gradient coordinates $I(X, Y)$. However, for reasons explained below (see modular patterning), it is preferable to proceed stepwise and let the swarm build itself in a *modular* fashion. The present model uses elementary patterns such as stripes and checkerboards. Naturally, unlike Turing patterns, each region is controlled here by a different gene set.

A biological embryo is a swarm of cells, where each cell contains a *gene regulatory network* (GRN) coding for its signalling and mechanic behavior. Through intercellular coupling between neighboring GRNs, the embryo becomes patterned into identity regions of differentiated gene expression, creating a "hidden geography" revealed by *in situ* hybridization. Essentially, logical combinations of regulatory switches ('or', 'and') translate geometric combinations of precursor patterns into new patterns (by union and intersection). Developmental genes are roughly organized in tiers or "generations". Earlier genes map the way for later genes, and gene expression propagates in a cascade. This principle has been beautifully demonstrated in the *Drosophila* embryo (see [4]). The intersection of various striping patterns along its three main axes gives rise to smaller regions such as the organ primordia and "imaginal discs," which are groups of cells marking the location and identity of the fly's future appendages (legs, wings, antennae). Going back in time, the whole process begins with concentration gradients of maternal proteins that diffuse across the initial cluster of cells and create the functional equivalent of a coordinate system, in a way similar to the PF-I process described in the previous subsection.

The early striping process of *Drosophila* is controlled by a regulatory hierarchy containing five main tiers of regulatory genes [4]. The present model relies on a three-tier caricature of the same idea, the *positional-boundary-identity* gene network [8, 9], which represents the genetic parameters of the PF process and is denoted by G_{PF} (Fig. 3c). In each cell-agent of our 2-D virtual embryo-swarms, the bottom layer of G_{PF} con-

tains the two positional variables X and Y seen previously; the middle layer, n "boundary" nodes $\{B_i\}_{i=1...n}$; and the top layer, m identity nodes $\{I_k\}_{k=1...m}$. Variables X, Y, B_i and I_k denote the gene expression levels or "activity" of the nodes. The boundary nodes compute linear discriminant functions of the positional nodes: $B_i = \sigma(w_{ix}X + w_{iy}Y - \theta_i)$, where $\{w_{ix}, w_{iy}\}_{i=1...n}$ are the regulatory weights from X and Y to B_i, parameter θ_i is B_i's threshold and sigmoid function $\sigma(u) = 1/(1 + e^{-\lambda u})$. The effect of a boundary node is to segment the embryo's plane into half-planes of strong and weak expression levels, 1 and 0 (Fig. 3a, middle row). Finally, the identity gene levels are given by logical combinations of the near-binary boundary gene values, for example, by calculating the products $I_k = \prod_i |w'_{ki}|(w'_{ki}B_i + (1-w'_{ki})/2)$, where $w'_{ki} \in \{-1, 0, +1\}$ represent ternary weights from B_i to I_k. This means that the i-th factor inside I_k can take three possible values: $(1 - B_i)$, 0 or B_i.

With this type of gene regulatory network, the "identity regions", i.e., the regions of high I expression, take the form of polygons at the intersection between several boundary lines (Fig. 3a, top row). When viewed together, they create a checkered pattern (Fig. 3b). These different colored regions represent different agent types and will be the starting point of new local SA and PF processes (see below). At this stage, similar to SA, each agent in the swarm also possesses (a) fixed "genetic" PF parameters in G_{PF} and (b) dynamic PF state variables—the gradient values n and the activity of G_{PF}'s nodes.

Simultaneous Growth and Patterning (SA + PF)

After describing the self-assembly of a non-patterned swarm and the patterning of a fixed swarm, SA and PF are now combined to create growing patterns (Fig. 4). Agents continually adjust their positions according to the elastic SA constraints, *while* continually exchanging gradient values and PF signals over the same dynamic links. This dual dynamics is guided by both genotypes G_{SA} and G_{PF} (Fig. 4d). Another mechanism, *cell division*, is also introduced at this point. Any agent A may divide with probability p at every time step and produce a new agent B, which is initially positioned a small distance from A with a random angle (Fig. 4c). Then the position of B and its neighbors rearrange under potential V as usual. Agent B inherits all of A's attributes, including genotype G_{SA+PF} and internal PF variables. It immediately starts contributing to the traffic of PF gradients that maintain the pattern's consistency at all times in the swarm.

From the SA point of view, a dividing swarm starting from few agents reliably grows through successive round shapes (Fig. 4a,c). In Fig. 1, the number of agents was constant and the expansion of the swarm was only due to elastic relaxation. In Fig. 4, agents are perpetually added while the swarm remains approximately in mechanical equilibrium at all times. From the PF point of view, the pattern is also maintained at all times by the continual propagation and readjustment of the gradients but also by the continual self-positioning of the four source agents N, S, W and E. To achieve a well-deployed compass as the one of Fig. 4b, source *migration* rules are added. Each agent contains four binary variables or "source flags" s_W, s_E, s_N and s_S, which are 0 almost everywhere and 1 in one of the four sources. According to the first migration rule, the W source must then always transfer value 1 of the s_W flag to a neighbor that has a greater n_E count than itself, and vice versa for E (same between N and S). This makes labels W and E move away from each other, hopping from agent to agent. The second migration rule stipulates that the W and E agents must also seek to minimize $|\, n_S - n_N \,|$, i.e., hop toward the NS midline (and symmetrically for N and S toward WE).

Modular, Recursive Patterning (PF[k])

Embryological patterns do not develop in one shot but in numerous incremental stages [6]. An adult organism is produced through gradual morphological refinement, following a cascade of genetic regulation from precursor developmental genes to secondary genes, tertiary genes, and so on. Importing this critical feature into the present model, the above gene network G_{PF} is extended to include a pyramidal *hierarchy* of network modules (Fig. 5g) able to generate patterns in a recursive fashion. First, the base network G_{PF} establishes main identity regions as before (Fig. 5a). Then a few subnetworks $G_{PF}^{(k)}$ further partition these regions into smaller identity compartments at a finer scale (Fig. 5e,f). The execution of $G_{PF}^{(k)}$ is triggered by the activity of node I_k in G_{PF}. This means that all agents with a high value of I_k start trading new local gradient counters $n_W^{(k)} \ldots n_S^{(k)}$, $n_{WE}^{(k)}$ and $n_{NS}^{(k)}$ (Fig. 5c,d).

Moreover, the sources of the four cardinal gradients are positioned at the *borders* of the I_k regions by "induction" from neighbors (Fig. 5b). This means that high-I_k agents set their source flags $s_W^{(k)} \ldots s_S^{(k)}$ to 1 if they are connected to agents from other regions $I_{k'}$. The exact flags that are switched on depend on the relative location of the regions, for example, $s_E^{(4)} = 1$ for all I_4 agents in contact with region I_5, while $s_S^{(4)} = 1$ for I_4 agents in contact with I_1, and so on (Fig. 5b,c). In cases where a particular gradient is missing because there is no adjacent border, e.g., the W sources in I_4, its sources are created from the *ends* of the opposite gradient (blue circles in Fig. 5d). Locally, an agent can recognize that it is the end of a gradient if it is a local maximum of that gradient counter n with respect to its neighborhood. Thus, in addition to source

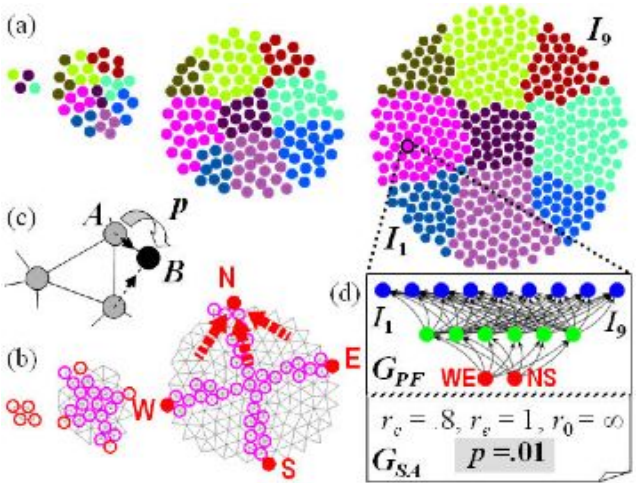


Figure 4: Simultaneous growth and patterning (SA+PF; see text). (a) Swarm growing from 4 to 400 agents by division. (b) Swarm mesh, showing gradient sources and midlines continually maintained by source migration, e.g., N moves away from S and toward WE. (c) Detail: an agent B created by A's division submits to SA forces and PF traffic. (d) Combined genetic programs inside each agent.

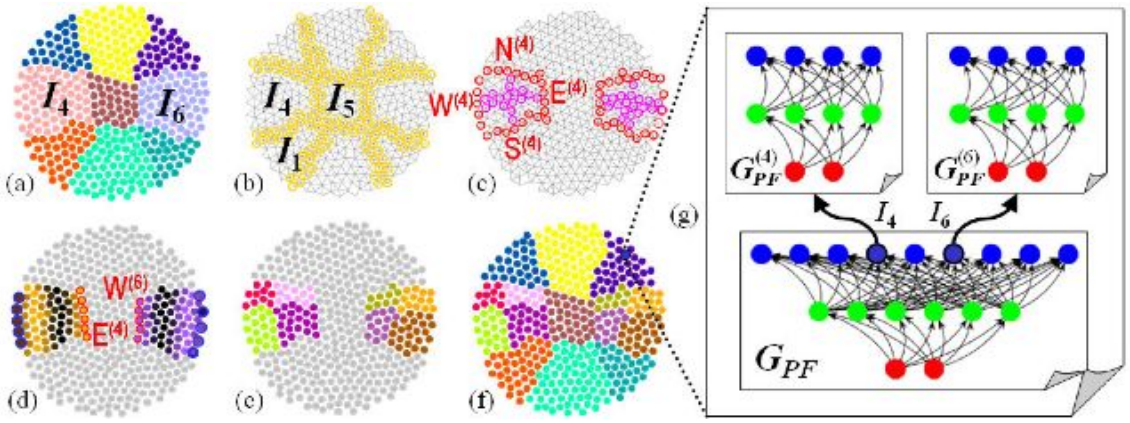


Figure 5: Modular, recursive patterning (PF[k]; see text). (a) 9-region swarm, as in Fig. 4a. (b) Border agents highlighted in yellow circles. (c) Border agents become new gradient sources at a lower scale inside certain identity regions. (d) Missing border sources arise from the ends (blue circles) of other gradients. (e,f) Subpatterning of the swarm in I_4 and I_6. (g) Corresponding hierarchical gene regulation network.

flags s, agents also contain "end flags" e_W … e_S that are switched on if the proper local-maximum conditions are filled. For example, $e_E^{(4)} = 1$ (hence $s_W^{(4)} = 1$) where $n_E^{(4)}$ is maximum, and conversely in region 6.

Modular, recursive patterning is similar to the imaginal discs of *Drosophila*; once a region has been marked to be the future site of a leg, wing or antenna (high I_k activity), a local coordinate system of morphogen gradients arises inside this region to form that organ [4]. From the artificial-life engineering viewpoint, recursive patterning is also preferable to one-shot patterning. In theory, Fig. 5f could also be produced by a direct $I(X, Y)$ mapping, but as the swarm continues to increase it would require maintaining global gradients over longer distances and would be unstable. Building a complicated image $I(X, Y)$ directly would also require maintaining a large number of pattern variables in each agent to implement every detail, and thus would be difficult to evolve. Modularity, by contrast, is an essential condition of *evolvability* [22]. In Fig. 5g, mutating G_{PF} would modify the whole body plan of Fig. 5a, whereas mutating $G_{PF}^{(4)}$ or $G_{PF}^{(6)}$ would only modify the "organs" of Fig. 5e. Moreover, without modules it would not be possible to have *differential* SA, necessary for the growth of *morphogenetic* structures and "limbs" other than blob swarms (see next subsection). Finally, modules can be *reused*, e.g., I_4 and I_6 could point to a common G_{PF} block. In summary, modularity is a desirable feature in genotypes just as in any software architecture or evolvable system. It seems that biological evolution discovered this principle naturally [3].

Modular, Anisotropic Growth (SA[k])

What is so far missing from the model is a true topological *deformation* dynamics, or "morphodynamics", that can confer non-trivial shapes to the organic system beyond simple blobs. To this aim, agents must be able to diversify their SA characteristics, *depending on their PF type and spatial position*, thus closing the feedback loop between SA and PF. In particular, they have to exhibit *inhomogeneous*, *anisotropic* cell division (varying p) and *differential* adhesion (varying V). For example, the growth of limb-like structures can be achieved by a coarse imitation of meristematic plant offshoots. In this process, only the tip or "apical meristem" of the organ is actively dividing at any time (Fig. 6). It is implemented here by letting agents have a non-zero probability of division p if and only if they are ends of a gradient (blue circles in Fig. 6c,d). These dividing cells can also control the angle of the "plane of cleavage". For example, a daughter cell B spawned by cell A can be placed opposite to the center of mass of A's neighbors (Fig. 6b). Almost equivalently, that position can be computed by factoring in the gradient values of the A's neighbors, i.e., calculating a discrete estimate of the local gradient slope in A.

Biological cells also stick to each other by means of adhesion proteins that cover their membrane. A great diversity of

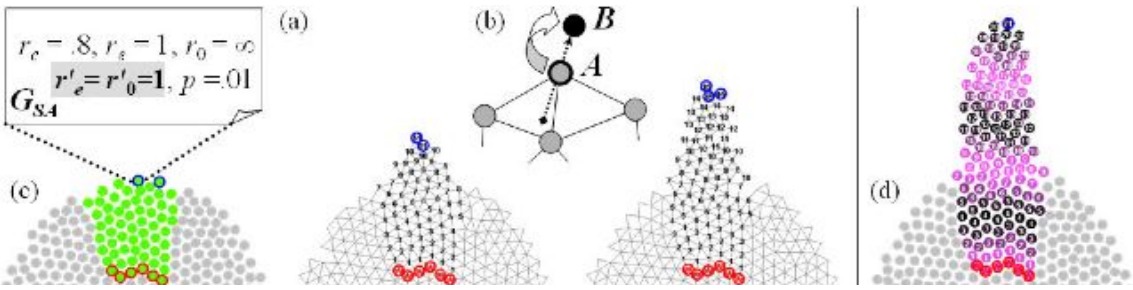


Figure 6: Modular, anisotropic growth (SA[k]; see text). (a) Genetic SA parameters are augmented with repelling V values r'_e and r'_0 used between the growing region (green) and the rest of the swarm (gray). (b) Daughter agents are positioned away from the neighbors' center of mass. (c) Offshoot growth proceeds from an "apical meristem" made of gradient ends (blue circles). (d) The gradient underlying this growth.

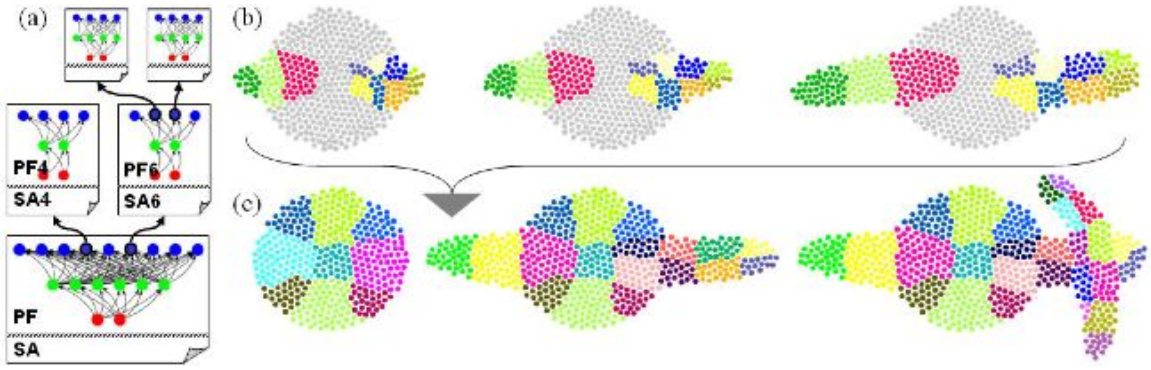


Figure 7: Modular growth and patterning (SA[k] + PF[k]; see text). (a) Example of a three-tier modular genotype giving rise to the artificial organism on the right. (b) Three iterations detailing the simultaneous limb-like growth process (Fig. 6) *and* patterning of these limbs during execution of tier 2 (modules 4 and 6). (c) Main stages of the complex morphogenesis, showing full patterns after execution of tiers 1, 2 and 3.

these proteins gives cells the ability to selectively recognize one another, thereby modulating the intercellular adhesion force or "stickiness". Some cells slide along one another without attaching, while others form tight, dense clumps. In the simple elastic force model, differential adhesion can be mimicked by varying V's parameters r_c, r_e, and r_0 depending on the agent types (Fig. 6a). For example, if agent A belongs to the limb region (green area) then $V(r_{AC})$ is attractive ($r_c < r_e = 1 << r_0$) for all neighboring agents C in that region, while it is repelling ($r'_0 < r'_e$) for all agents C outside that region (gray area). This can be decided locally by comparing the types of A and C, i.e., whether their respective highest-valued I_k nodes are the same or not. Just like inhomogeneous division, differential adhesion is an essential condition of complex shape formation [11, 15].

Modular Growth and Patterning (SA[k] + PF[k])

Putting everything together, full morphologies can develop and self-organize from a few agents (Fig. 7). These morphologies are *complex*, *programmable* and *reproducible*. They are architecturally complex because they can be made of any number of various modules and parts that are not necessarily repeated in periodic or trivial ways. They are programmable phenotypes emerging from the same genotype carried by every agent of the swarm (Fig. 7a). They are also reproducible, as their morphological structures are not left to chance but dictated by the genotype. The exact agent positions at the microscopic level are still random, but not the mesoscopic and macroscopic regions that they form.

The modularity of the phenotype is also a direct reflection of the modularity of the genotype: the hierarchical SA + PF dynamics recursively unfolds inside the different regions and subregions that it creates. Each $SA^{(k)} + PF^{(k)}$ block can be reused, either by convergent I_k links (not shown here) or by exact *duplication*. It can also *diverge* from other blocks, i.e., receive different internal genetic SA and PF parameters that give each region a different morphodynamic behavior and activity landscape. Duplication followed by divergence is the basis of *serial homology* (e.g., vertebrae, teeth, digits), a major natural evolutionary mechanism. The integration between SA and PF is controlled through the identity nodes I_k: just as these nodes turn on gene expression activity in subordinate $G_{PF}^{(k)}$ modules to create new local segmentation patterns, they also simultaneously turn on behavioral changes in subordinate $G_{SA}^{(k)}$ modules to create new morphodynamical behaviors.

There remains to determine the *scheduling* policy of genotype execution inside each agent. When does an agent decide to follow the latest $SA^{(k)} + PF^{(k)}$ branch opened by a new identity gene I_k? Since there is no centralized control in the swarm, module-switching decisions must be asynchronous. However, starting a new module k as soon as I_k's activity is high would not be a good strategy, especially while the agent's current region is still developing. For example, in the early stages of Fig. 4a, cells often change type (color) and should not start creating new subpatterns before they reach maturity. Thus there must be some regional synchronization mechanisms to help agents make scheduling decisions. The present model, however, only adopts a primitive clocked scheme based on the number of iterations. For now, all agents simply switch to the next $SA^{(k)} + PF^{(k)}$ stage if their internal timer exceeds a time point t_k set in advance and added to their genetic baggage.

Discussion

The goal of this work was to contribute to a better theoretical understanding of complex morphogenesis, especially biological, in order to reproduce it artificially and pave the way for development-based evolutionary innovation. It presented a model of pattern formation in self-assembling swarms that contained a large number of agents and displayed complex but reproducible phenotypic emergence from a modular genotypic program. As e*mbryomorphic engineering*, it essentially advocated a "fine-grain" approach to systems design based on relatively simple programmed agents. Naturally, beyond the proof-of-concept simulations presented here, and other preliminary work [8, 9], a more systematic exploration is needed. Next steps should involve the mass-production of virtual organisms to support (a) statistical analysis of shape and (b) evolutionary search based on module variation and *function*.

Future Work

From form to function. While the task of "meta-designing" laws of artificial development inspired from biology is challenging, it only constitutes the first part of an embryomorphic engineering effort. Another important question is *functional*

meta-design: once a self-developing infrastructure is mature, what computing capabilities can it support? What do its cell-agents and organ-regions actually represent in practice? In biological organisms, although cell physiology often partakes in development (e.g., electrical signals of neurons guiding synaptogenesis), there seems to be a broad distinction between developmental genes and the rest of the genome. In computing systems, these two modes could also be decoupled into two different sets of agent variables. After reaching developmental maturation, and while still fulfilling maintenance and self-repair tasks, morphogenetic SA and PF activity (i.e., division, position information and patterning signals) would give way to another type of activity subserving functional computation. Obviously, the type of computation would entirely depend on the nature of the agents: processors, software, robot parts, mini-robots, etc. In fact, in many computing domains, there is already a demand for precise self-formation capabilities. A multitude of micro-components containing the same code could self-organize without traditional VLSI precision or reliability [7, 16]. Mobile sensor and actuators could dynamically connect in self-managing networks [2]. Small-footprint software objects could diversify and self-deploy to achieve a desired level of application functionality (e.g., "immune" security). Articulated robotic parts, reconfigurable devices [14, 13, 10], or mobile robot formations [5] could also be guided by complex *and* controllable morphologies.

From ontogeny to phylogeny. After growth and function, one must also define how the system *evolves*, i.e., how it *varies* (randomly) and how it is *selected* (non-randomly). Different selection strategies are possible, either focusing on pre-specified forms, or pre-specified functions, or allowing unspecified outcomes. When *selecting for form*, a hard reverse engineering problem must be addressed: given a desired phenotype, what is the genotype that can produce it? While deterministic reverse compilation is possible in some cases [16], parameter search is difficult in general. Fitness criteria that reward only the target shapes create jagged landscapes of unreachable peaks. A smoother approach is to define a "shape distance" as an increasing function of favorable mutations. It is conjectured here that this kind of gradual search might actually *benefit*, not suffer, from the high genotype dimensionality of an embryomorphic model, compared to the direct mappings of genetic algorithms. Hierarchical gene regulatory networks might be better at providing the fine-grain mutations required by the gentle-slope search. Complex systems inherently have greater variational power, as they allow combinatorial tinkering on highly redundant parts.

However, beside gaining self-repair properties, why constrain a self-assembling system to produce a pre-defined shape? More benefits might come from such systems by *selecting for function* while leaving freedom of form. Gradual optimization could rely on a distance of *performance* to pre-defined goals, instead of shapes, allowing the most successful candidates to reproduce faster and mutate. Functional selection under free form is used in evolutionary robotic systems [14, 13], but mostly based on macroscopic genotype-phenotype encodings. Here, too, a larger number of agents, such as in multicellular embryogenesis, could prove more favorable to a successful search. Finally, in a third scenario, specifications could be relaxed to the point of *being open to surprise* and harvesting unexpected but useful organisms from a free-range menagerie. Reconciling the antagonistic poles of "planning" and "autonomy" ultimately hinges on two complementary aspects: (a) fine-grain variation-by-mutation mechanisms yielding a large number of search paths and (b) loose selection criteria yielding a large number of fitness maxima. With more search paths covering more fit regions, evolution is more likely to find good matches.

References

1. Ball, P. (1999). *The Self-Made Tapestry.* Oxford University Press.
2. Beal, J. and Bachrach, J. (2006). Infrastructure for engineered emergence on sensor/actuator networks. *IEEE Intell. Sys.*, 21(2): 10–19.
3. Callebaut, W. and Rasskin-Gutman, D. editors. (2005). *Modularity: Understanding the Development and Evolution of Natural Complex Systems*. The MIT Press, Cambridge, MA.
4. Carroll, S. B., Grenier, J. K. and Weatherbee, S. D. (2001). *From DNA to Diversity*. Blackwell Scientific, Malden, MA.
5. Christensen, A., O'Grady, R. and Dorigo, M. (2007). Morphology control in a self-assembling multi-robot system. *IEEE Robotics & Automation Magazine*, 14(4): 18–25.
6. Coen, E. (2000). *The Art of Genes*. Oxford University Press, UK.
7. Coore, D. (1999). *Botanical Computing: A Developmental Approach to Generating Interconnect Topologies on an Amorphous Computer*, Ph.D. thesis, Dept. of Elec. Eng. & Computer Science, MIT.
8. Doursat, R. (2006). The growing canvas of biological development: Multiscale pattern generation on an expanding lattice of gene regulatory networks. *InterJournal: Complex Systems*, 1809.
9. Doursat, R. (2008). Organically grown architectures: Creating decentralized, autonomous systems by embryomorphic engineering. In Würtz, R. P., ed., *Organic Computing*, pages 167-200. Springer.
10. Goldstein, S. C., Campbell, J. D. and Mowry, T. C. (2005). Programmable matter. *IEEE Computer*, 38(6): 99–101.
11. Hogeweg, P. (2000). Evolving mechanisms of morphogenesis: On the interplay between differential adhesion and cell differentiation. *Journal of Theoretical Biology*, 203: 317–333.
12. Kirschner, M. W. and Gerhart, J. C. (2005). *The Plausibility of Life: Resolving Darwin's Dilemma.* Yale University Press, New Haven.
13. Komosiński, M. and Rotaru-Varga, A. (2001). Comparison of different genotype encodings for simulated three-dimensional agents. *Artificial Life*, 7(4): 395–418.
14. Lipson, H. and Pollack, J. B. (2000). Automatic design and manufacture of robotic lifeforms. *Nature* 406: 974–978.
15. Marée, A. F. M. and Hogeweg, P. (2001). How amoeboids self-organize into a fruiting body: Multicellular coordination in *Dictyostelium discoideum*. *PNAS*, 98(7): 3879–3883.
16. Nagpal, R. (2002). Programmable self-assembly using biologically-inspired multi-agent control. *First International Conference on Autonomous Agents*, Bologna, July 15-19.
17. Salazar-Ciudad, I. and Jernvall, J. (2002). A gene network model accounting for development and evolution of mammalian teeth. *PNAS*, 99(12): 8116–8120.
18. Sayama, H. (2007). Decentralized control and interactive design methods for large-scale heterogeneous self-organizing swarms. *Advances in Artificial Life: Proceedings of the 9th ECAL*.
19. Shapiro, B. E., Levchenko, A., Meyerowitz, E. M., Wold, B. J. and Mjolsness, E. D. (2003). Cellerator: Extending a computer algebra system to include biochemical arrows for signal transduction simulations. *Bioinformatics*, 19(5): 677–678.
20. Stanley, K. O. and Miikkulainen, R. (2003). A taxonomy for artificial embryogeny. *Artificial Life*, 9(2): 93–130.
21. Vicsek, T., Czirók, A., Ben-Jacob, E., Cohen, I. and Shochet, O. (1995). Novel type of phase transition in a system of self-driven particles. *Physical Review Letters*, 75: 1226–1229.
22. Watson, R. A. and Pollack, J. B. (2005). Modular interdependency in complex dynamical systems. *Artificial Life*, 11(4): 445–458.
23. Wolpert, L. (1969). Positional information and the spatial pattern of cellular differentiation development. *J. Theoret. Biology* 25: 1–47.

Entropy production in an energy balance Daisyworld model

J. G. Dyke

University of Sussex, Brighton, East Sussex, UK, BN1 9QH
j.g.dyke@sussex.ac.uk

Abstract

Daisyworld is a simple mathematical model of a planetary system that exhibits self-regulation due to the nature of feedback between life and its environment. A two-box Daisyworld is developed that shares a number of features with energy balance climate models. Such climate models have been used to explore the hypothesis that non-equilibrium, dissipative systems such as planetary atmospheres are in a state of maximum entropy production with respect to the latitudinal flux of heat. When values for heat diffusion in the two-box Daisyworld are selected in order to maximize this rate of entropy production, the viability range of the daisies is maximized. Consequently planetary temperature is regulated over the widest possible range of solar forcing.

Introduction

Although not intended as such, Daisyworld can be regarded as an example of an artificial life model. It develops a conceptual framework that explores life and environment as they could be on a planet similar to the Earth. Daisyworld was originally proposed as a mathematical proof of concept for James Lovelock's Gaia Hypothesis: that the Earth and its biota are a self-regulating system that will reduce the effects of otherwise deleterious perturbations (Lovelock, 1995). Since its creation some twenty five years ago (Lovelock, 1983) Daisyworld has been extended and modified in order to address a number of different research questions. See (Wood et al., 2008) for a recent review. The consensus is that it has matured to the extent that it can be considered as a model in its own right rather than as an ancillary component of Gaia theory. A central message from Daisyworld is that when life affects the environment as well as being affected by the environment, self-regulation may emerge.

Modelling of the Earth's climate can take place at various levels of complexity, from zero-dimensional models such as the original Daisyworld to three-dimensional general circulatory models that are very computationally expensive and defy exact analysis. A relatively recent development of a two-box Daisyworld (Harvey, 2004) is conceptually similar to one-dimensional energy balance climate models that have been used to explore the *Maximum Entropy Production Principle* (MEPP) (Lorenz et al., 2001). It is proposed that certain energetically open and driven systems, such as planetary atmospheres, are in states that maximise the rate of entropy production. It is straightforward to introduce thermodynamic constraints into the two-box Daisyworld such that maximum entropy production can be achieved. If the Earth's atmosphere is in a non-equilibrium state that maximses entropy production then it seems reasonable to ask what would the effects on Daisyworld regulation and stability be if it were in a MEPP state?

Daisyworld

Daisyworld is an imaginary grey planet orbiting a star similar to the Sun. It is home to two daisy types: black and white. Albedo is a measure of the reflectivity of an object. In Daisyworld the black daisies have a low albedo (0.25), the grey bare earth intermediate albedo (0.5) and the white daisies a high albedo (0.75). The white daisies, having the highest albedo in the model, reflect more of the short wave energy from the star and so have a lower temperature than either the grey planet or the black daisies. The same applies to the black daisies but in reverse. The black and white daisies share a viability range of temperature. They are only able to grow when the local ambient temperature is within 5-40 degrees Celsius. Within this range growth rates of the daisies vary, with optimum growth being achieved when the temperature is 22.5C. Simulations begin when the star is dim and the temperature of the planet is below 5C. As the star increases in brightness the planetary temperature reaches 5C and black daisies begin to grow. The increase in the number of black daisies is further increased via a feedback loop: more black daisies leads to a lower planetary albedo and so more energy is absorbed that warms the ground that leads to an increase of the growth rate. This feedback loop is regulated by the parabolic growth rate of the daisies. As the temperature increases past 22.5C, the daisy growth rate decreases. At steady state the ambient temperature, growth rate and death rate are at equilibrium. As the energy output of the star continues to increase, coverage in black daisies decreases and white daisies begin to grow. This initiates a feedback loop that is the inverse of the effect of the black

daisies and so decreases their temperature. Again, this feedback loop is regulated by the parabolic growth rate of the daisies. Increasing the amount of energy from the star results in a progressive increase in white daisies (and decrease in black daisies) until the maximum coverage of white daisies is reached. Any further increase in energy takes the ambient temperature past the point where growth rates balance death rates and so the coverage of white daisies decreases. This leads to a rapid collapse of white daisies similar in nature to the population explosion of the black daisies. The differential coverage of white and black daisies results in a system that effectively regulates ambient planetary temperature to within the viability range. Whereas the temperature of a bare lifeless planet would increase in an approximately linear fashion with increases in luminosity, when black and white daisies are present, ambient temperature remains within the viability range over a wide range of solar forcing.

The Maximum Entropy Production Principle

The Earth's atmosphere can be viewed as a heat engine: motions are driven by the flow of heat from the hot equator to the cold polar regions. The amount of work that can be done by this flow of heat depends on the temperatures of the reservoirs: the greater the drop in temperature for a given amount of heat flow, the greater the thermodynamic efficiency of the system and so a greater amount of work output and entropy produced. Such heat processes do not produce entropy at an arbitrary rate. Two extreme principles have been formulated to describe their characteristic behaviour. For systems near thermodynamic equilibrium with fixed boundary conditions, (Prigogine, 1962) formulated the principle of minimum entropy production (MinEP) stating that the steady state of the process is associated with a MinEP state. However, many processes do not have fixed boundary conditions and are far from equilibrium. For those processes it has been proposed that they maintain steady states in which the production of entropy is maximized if there are sufficient degrees of freedom associated with the processes - the Maximum Entropy Production Principle (MEPP). It may not be necessary to understand the detailed internal dynamics of such systems in order to make accurate models and predictions. An impressive demonstration of this was with (Paltridge, 1975) who successfully reproduced the Earth's latitudinal temperature profile in a simple energy balance climate model by assuming that the rate of diffusion was that which maximized the rate of entropy production via latitudinal heat transport. However, in the absence of any proposed mechanism or other system that maximized entropy production in this manner, Paltridge's results failed to gain significant traction within the scientific community. This situation has changed somewhat as MEP has been observed in the atmospheres of Mars and Titan (Lorenz et al., 2001) and there have been attempts (Dewar, 2003) and (Martyushev and Seleznev, 2006) to place the MEPP on firm information theoretic foundations. See (Ozawa et al., 2003) for a review.

Two-Box Daisyworld

The two-box Daisyworld was motivated by a desire to produce the simplest implementation of the Daisyworld control system. Whilst the original Daisyworld was itself a simple model of complex real-world phenomena, analysis was not trivial. The two-box model simplifies Daisyworld further by removing space competition dynamics by having each daisy occupy separate 'beds' or boxes and dispensing with a separate birth and death rate and finding steady state daisy coverage with a single linear function. Each grey box is seeded with either black or white daisies. When the seeds are dormant, the boxes have the same temperature. As the seeds germinate and daisies begin to cover each box, a temperature gradient is established. The two-box Daisyworld is represented schematically in Figure 1.

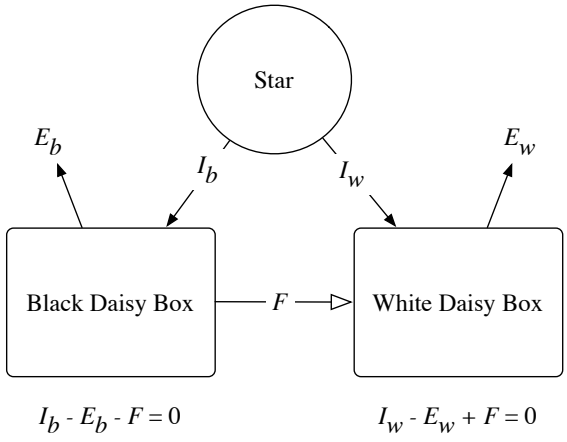


Figure 1: Schematic of Two-Box Daisyworld

The two-box Daisyworld which is conceptually similar to a two-box energy balance climate model (North et al., 1981). Incoming energy in the form of short-wave energy I_i warms each box and is radiated back into space as long-wave emissions E_i with an amount of heat flux F proportional to the temperature gradient between the boxes. Only black daisies are seeded in the black daisy box and only white daisies in the white daisy box. White daisies, being lighter than the black daisies will reflect more energy from the star. Black daisies, being darker, absorb more energy. Hence the black daisies are warmer and the white daisies are cooler than the grey bare earth. The two boxes are coupled via a heat conducting medium which when there is non-zero coverage of daisies allows heat to flow from the black to white daisy box.

Those with a familiarity of climate modelling will immediately see the similarity to a zonal energy balance model. The temperature of each daisy box is found with

$$T_b^4\sigma = I(1 - A_b) - F, \quad (1)$$

$$T_w^4\sigma = I(1 - A_w) + F, \quad (2)$$

where A_i is the albedo of the boxes, σ is the Stefan-Boltzmann constant having a value of 5.67×10^{-8} J $s^{-1}m^{-2}K^{-4}$, I is insolation, the amount of energy received on the surface of the planet from the star in units of $W^{-1}m^{-2}$. As in previous studies on Daisyworld, the amount of insolation will be parameterized by a luminosity variable, L. L is a non-dimensional index of stellar brightness. $I = L \times 1000W^{-1}m^{-2}$ where $L \in [0, 2]$. We can think of luminosity as a 'dimmer switch' that modulates the brightness of the star and thus the amount of energy received on the surface of the planet. The temperature of the planet is the mean of the two box temperatures, $T_p = 0.5(T_b - T_w)$. The heat flux F between the two boxes is found with

$$F = D(T_b - T_w). \quad (3)$$

The flux of heat is proportional to the temperature difference between the two boxes and a diffusion parameter D normally measured in units of $W^{-1}m^{-2}K^{-1}$. As we will be investigating the comparative effects of entropy production, we will set the surface area between the two boxes to unity, and heat flux and entropy will be measured in non-dimensional or arbitrary units. The diffusion term D will be scaled from 0, which produces no heat flux, to 1 which produces maximal heat flux with the boxes being isothermal. The albedo of each daisy box is found in a similar fashion to the original Daisyworld

$$A_b = AG(1 - \alpha_b) + AB\alpha_b, \quad (4)$$

$$A_w = AG(1 - \alpha_w) + AW\alpha_w. \quad (5)$$

AG, AB, AW are parameters that determine the fixed albedo of bare ground, black daisies and white daisies, and these are set to 0.5, 0.25 and 0.75 respectively. α_i is the proportional daisy coverage from 0 to 1 which is found with equation 6. There is an optimal temperature T_{opt} that produces 100% coverage, with coverage decreasing linearly to zero as temperature decreases or increases away from this value.

$$\alpha_i = Max[1 - 2(|T_{opt} - T_i|)/R, 0], \quad (6)$$

where $T_{opt} = 22.5$ and R is the range of temperature over which non-zero daisy coverage is achieved. It is fixed at 35 thus daisies grow at 5C, have maximal coverage at 22.5C and are back to zero coverage at 40C. It is assumed, as with other studies on Daisyworld, that the rate of change of daisy coverage is sufficiently faster than that of luminosity so as to allow the above equations to be numerically integrated to steady state whilst luminosity is fixed. The implementation details can be found in previous studies (Harvey, 2004), (Dyke and Harvey, 2005) and (Dyke and Harvey, 2006). Essentially, for any fixed luminosity and fixed rate of heat diffusion, the daisy coverage is initialised to 1 (maximum coverage), albedo, temperature, heat flux and then temperature plus heat flux and new coverage values are computed. The current daisy coverage is then adjusted a small amount towards this new coverage. When the percentage change of daisy coverage is 0.001 per iteration of this loop, 200,000 iterations produce changes in coverage, albedo, heat flux and temperature that are no greater than 10^{-22}. Figure 2 shows numerically computed steady state values for daisy coverage and planetary temperature when the diffusion parameter is fixed at 0 and 0.5.

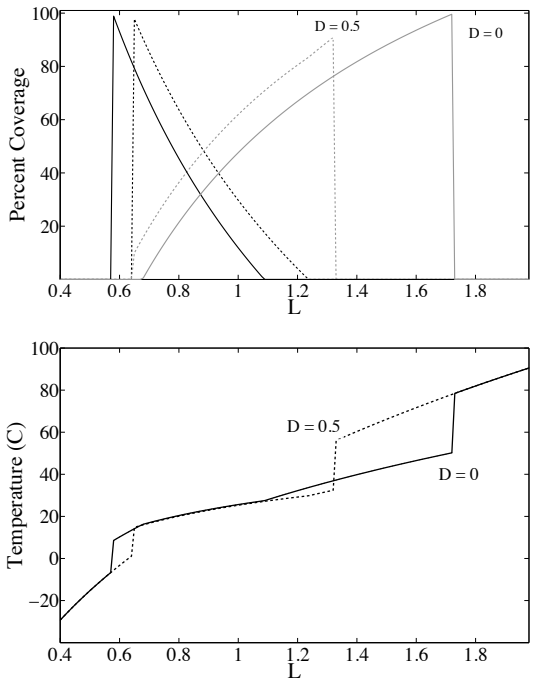


Figure 2: Two-Box Daisyworld $D = 0, D = 0.5$

Numerically computed results when $D = 0$ (solid lines) and $D = 0.5$ (dashed lines). The top plot shows daisy coverage over luminosity. When $D = 0$ the black daisies (black line) grow at lower luminosity and white daisies (grey line) grow at higher luminosity than when diffusion is set at an 'intermediate' value of 0.5. When $D = 0.5$ white daisies grow at lower and black daisies grow at higher luminosities. The bottom plot shows planetary temperature for the same results. The rate of change of planetary temperature when both daisy types are present is less than when only one daisy type is present.

Initializing daisy coverage to 1 for any luminosity removes hysteresis from the system. Hysteresis is recovered if, as in the original Daisyworld model, daisy coverage is allowed to 'evolve' by initializing new luminosity coverage with previ-

ous steady state luminosity coverage whilst making changes in luminosity very small.

Entropy production in Daisyworld

Previous studies have investigated entropy production in Daisyworld: (Pujol, 2002) within the original version, (Tonaizzo et al., 2004) with a version that allowed an arbitrary number of daisy types and a two dimensional cellular automata version in (Ackland, 2004).

The first two studies faced certain limitations due to the particular implementation of heat flux in the respective models. However they both concluded that when Daisyworld maximizes the rate of entropy production, there is an increase in the range of luminosity over which daisies grow. Rather than assuming Daisyworld is in a maximizing entropy production state, (Ackland, 2004) tests the contrary hypotheses that Daisyworld self-organises to either maximize entropy production or maximize the total amount of life for any given luminosity ('MaxLife'). It is found that a maximising life not maximising entropy principle is selected. However, due to the modelling assumptions of cellular automata Daisyworlds, computing the rate of entropy production via heat flux is not possible. Consequently it is the rate of biodiversity entropy that (Ackland, 2004) measures. See (Wood et al., 2008) for a more detailed discussion of these studies.

The two box formulation of Daisyworld is similar to energy balance climate models in which there is a flux of heat from warm to cool regions (North et al., 1981). Whereas in these models such heat gradients are produced via different amounts of energy received on the surface of the planet due to different latitudes, in the two-box Daisyworld model the difference in temperature is due to different albedo. However, heat will flow from warm to cool regions irrespective of how such a situation was produced, and the method of calculating entropy production budgets in energy balance box models such as (Lorenz et al., 2001) can be naturally employed in order to calculate entropy production in the two-box model. The rate of entropy production is a function of heat flux over the difference in temperature between the hot and cold boxes.

$$\frac{dS}{dt} = \frac{F}{T_w} - \frac{F}{T_b}. \tag{7}$$

The greatest rate of entropy production will be achieved with the greatest temperature difference and the greatest heat flux between the two daisy boxes. Attempting to increase entropy production by increasing heat flux via increasing diffusion may lead to a decrease in the temperature gradient and thus a decrease in thermal efficiency of the 'heat engine' and so a decrease in the rate of entropy production. Consequently maximizing entropy production is a balancing act with the value of D required to produce maximum rates of entropy production varying with the driving of the system. Figure 3 shows the unimodal function of entropy production over D with daisies present and fixed luminosity.

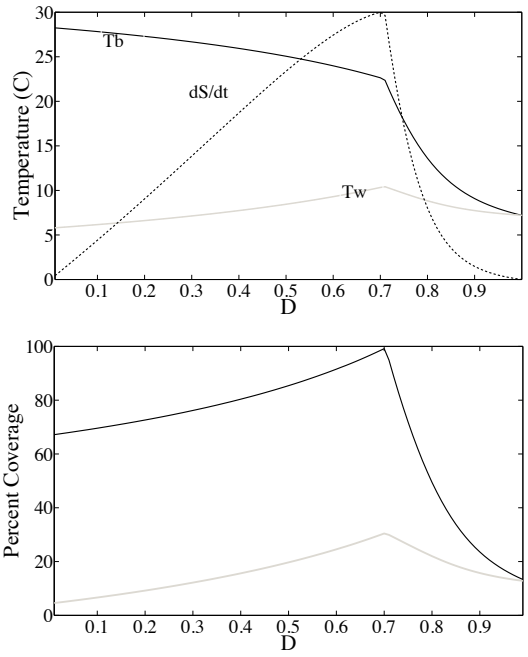


Figure 3: Two Box Daisyworld MEP results

The rate of entropy production and daisy box temperatures are shown in the top plot and black daisy coverage (solid black line) and white daisy coverage (solid grey line) is shown in the bottom plot for various values of heat diffusion where luminosity is fixed at 0.7. The greatest rate of entropy production is produced when $D \approx 0.7$. Increasing diffusion leads to a decrease in the temperature difference between the boxes and entropy production until the daisy boxes are isothermal with no entropy production and steady state coverage being that of a 'grey' daisy type with albedo of 0.5. The coverage of daisies undergoes a sharp decline as increasing heat flux drives the black and white box temperatures away from the optimal temperature of 22.5C. The value of D required to produce maximum entropy production will vary with L.

Maximising entropy production

It has been postulated that certain dissipative systems maximize the rate of entropy production. *How* these systems reach such non-equilibrium stable states is a different topic of enquiry. Here I will assume that the imaginary two-box planetary system, like the Earth, possesses an atmosphere with sufficient degrees of freedom to produce heat fluxes that lead to the maximisation of entropy production. In doing so we can construct a thought experiment analogous to Maxwell's Demon which is used to explore certain aspects of the second law of thermodynamics. Rather than monitor-

ing the velocity of air molecules, our demon monitors the flux of heat and temperature of the daisy boxes. It has a dial at its disposal that modulates the diffusivity of the atmosphere. For any luminosity value, the demon can alter the diffusion and in doing so find the rate of diffusion that maximizes equation 7. The behaviour of the demon can be implemented in a search algorithm that modulates D for any fixed L in order to find the value of D that produces the highest rate of entropy production.

Results

Figure 4 shows the effects of the maximising demon on the two-box Daisyworld. It can be seen that the 'any-daisy' range, the range of luminosity over which any daisy is present, is the same as when D is fixed at zero. It will be shown that this represents the greatest range of daisy growth possible. Planetary temperature is regulated with either daisy type present. The effects of increasing luminosity on planetary temperature are further reduced within the 'both-daisy' range, the range of luminosity over which black and white daisies are present. It will be shown that when the rate of entropy production is maximized the both-daisy range is also maximized and so the range of luminosity that sees the smallest rate of change of planetary temperature is maximized. By altering D to maximize the rate of entropy production we also maximize the any-daisy and both-daisy ranges and thus maximize the range of luminosity over which planetary temperature is regulated at all and regulated most effectively.

It is straightforward to show that maximising entropy production will lead to a maximisation of the any-daisy range. In order for there to be non-zero entropy production, there must be a temperature gradient between the two daisy boxes. Therefore there must be non-zero coverage in either or both boxes. This equates to black daisies growing at the lowest possible luminosity and white daisies growing at the highest possible luminosity. In order for black daisies to grow when the planet is cool, the amount of heat flux must be reduced. Consequently at the limits of the any-daisy range, $D \rightarrow 0$. The limits of the any-daisy range are also when the daisies reach maximum coverage. The intuition is that the external forcing drives the daisy coverage higher until the biota can no longer respond at which point any further increases lead to a population collapse. The daisy box temperature that produces maximum coverage is 295 degrees Kelvin. Setting $D = 0$ enables us to find the luminosity for the start and end of the any-daisy range. With the parameter and constant values as detailed in the Two-Box Daisyworld section, computing equations 8 and 9 give the start and end of the any-daisy range to be be 0.5764 and 1.729 respectively (to within 4 significant figures). These are the values returned with numerical results

$$L_{start} = \frac{295.5^4\sigma}{S(1-AB)}, \tag{8}$$

$$L_{end} = \frac{295.5^4\sigma}{S(1-AW)}. \tag{9}$$

Finding the limits of the both-daisy range is not a trivial exercise [1]. At lower luminosities we want to find that combination of luminosity and heat flux that increases the white daisy box to 5C. This will be achieved at lower luminosities with higher heat flux. But if heat flux is too high, the temperature of the black daisy box can be cooled so far as to lead to a collapse in the black daisy population. Similarly we want to find the greatest heat flux that can maintain the black daisy box to within 40C without increasing the temperature of the white daisy box beyond 22.5C and so its collapse. The hypothesis that maximising the both-daisy range is achieved when entropy production is maximized can be checked by computing steady states in which D is adjusted in order to maximize the both-daisy range. These produce values (to 4 significant figures) of 0.6285 for the start and 1.283 for the end of the both-daisy range which match the values for the both-daisy range when D is adjusted to maximize the rate of entropy production. These values are also returned when D is adjusted in order to maximize the total coverage of daisies for any luminosity (a Maxlife scenario). Results are shown in figure 5. An important difference between the MEPP and Maxlife models is that D reaches a maximal value in Maxlife such that the black and white boxes are isothermal and thus entropy production falls to zero; $T_b = T_w = T_{opt}$ and so $dS/dt = 0$.

Discussion

We have seen that when diffusivity is altered in the two-box Daisyworld model in order to maximise the rate of entropy production via latitudinal heat flux, the range of luminosity over which daisies grow is maximised. Care must be exercised when interpreting results from such simple models, especially in the absence of empirical data. The model presented in this paper is based upon established climate models and the observed real world phenomena of entropy maximisation via latitudinal heat flux. However the maximisation of entropy production was *imposed* on the model by a notional demon. This is not necessarily a limitation, but does need to be appreciated. MEPP models can be regarded as 'black box' models in that one need not know the details of the system's dynamics. Within the two-box model we have assumed that there are sufficient degrees of freedom within the processes that determine the diffusivity between the two boxes to afford the system the ability to configure itself into a state that maximizes the rate of entropy production. When the two-box model is in a MEP state with respect to latitudinal heat flux, it is not in a MEP state with respect to

[1] For this author.

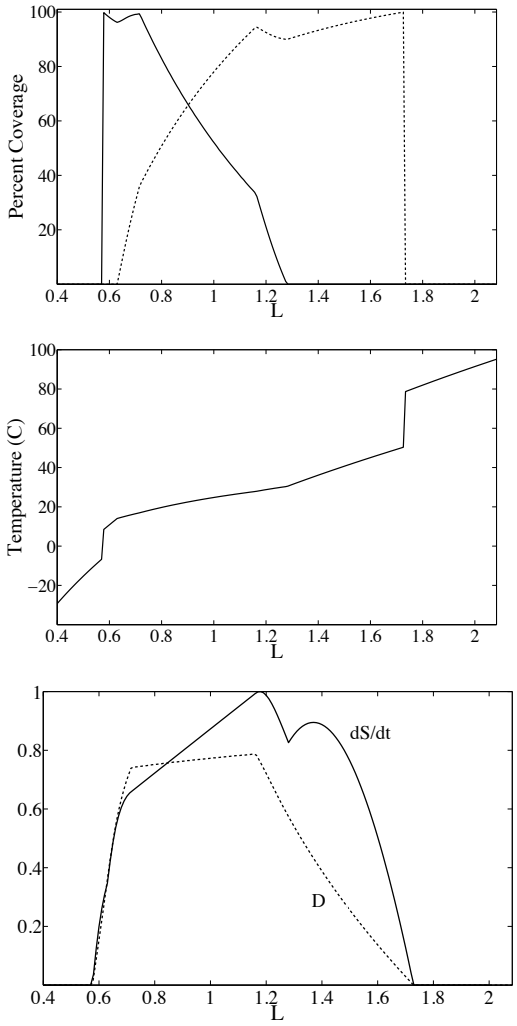


Figure 4: Two-box Daisyworld MEP results

Black (solid line) and white (dashed line) daisy percentage coverage are shown in the top plot. Planet temperature is shown in the middle plot. Normalised entropy production (solid line) and the diffusion parameter D (dashed line) are shown in the bottom plot. When adjusting heat flux to maximize the rate of entropy production, the any-daisy and both-daisy ranges are maximsed. Entropy production is greatest when $L \approx 1.2$.

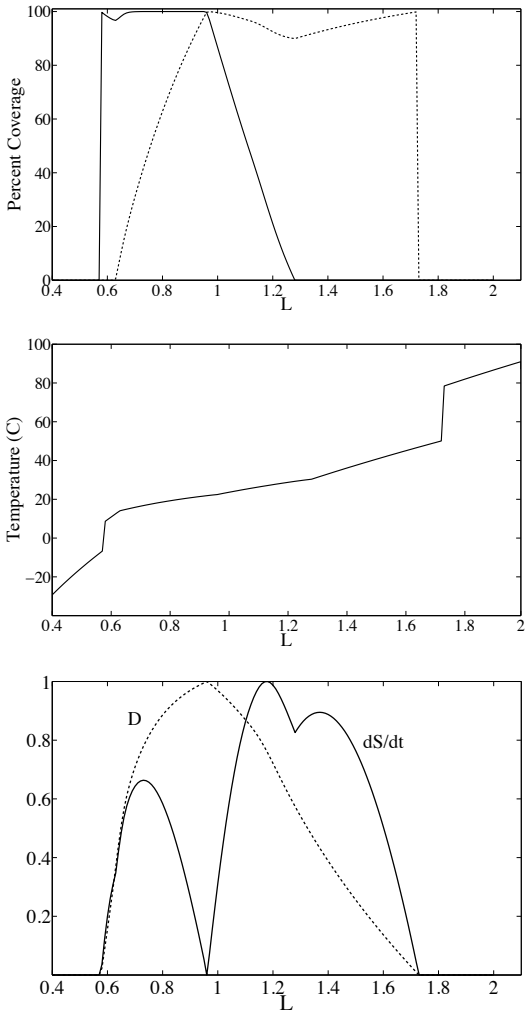


Figure 5: Two-box Daisyworld MaxLife results

Black (solid line) and white (dashed line) daisy percentage coverage are shown in the top plot. Planet temperature is shown in the middle plot. Normalised entropy production (solid line) and the diffusion parameter D (dashed line) are shown in the bottom plot. When adjusting heat flux to maximize the rate of entropy production, the any-daisy and both-daisy ranges are maximised. Diffusion reaches a maximum value and entropy production a minimum value when $L \approx 0.95$.

short wave to long wave radiative balance. This would be achieved by making Daisyworld as dark as possible thus absorbing as much of the star's energy and so converting the maximum amount of short wave radiation to long wave radiation. On Earth, the majority of entropy production is due to short wave radiation from the Sun warming the surface and atmosphere of the planet and then radiating this now long wave energy back into space. The Earth does *not* maximize the rate of this entropy production. This is due to the insufficient degrees of freedom the radiative mechanism possesses with MEP only to be expected in complex, turbulent, dissipative systems such as planetary climates (Ozawa et al., 2003). Arguably the utility of the results presented here will depend on the plausibility and conceptual coherence of the MEPP. If the MEPP is 'real' and applicable to a range of systems then it seems reasonable to investigate its effects on systems such as life-mediated climate models.

The effects of maximising entropy production in the two-box model was to maximize the any-daisy and both-daisy ranges and so maximize the range of solar forcing over which the system is self-regulating. It is tempting to claim that by maximizing entropy production we have maximized self-regulation. However the current results must not be overstated. For example the model's response to stochastic external or internal perturbations was not explored. We assumed that the rate of change of daisies was sufficiently faster than the star to allow the star to remain fixed whilst steady state coverage was found. We also assumed that diffusion was fixed whilst the system moved to steady state. A wide range of different model dynamics could well be produced by relaxing or altering these assumptions. That said, there are immediate intuitive connections between self-regulation and entropy production. The system can only regulate temperature when daisies are present. Latitudinal heat flux entropy can only be produced when daisies are present. If the two-box Daisyworld atmosphere had sufficient degrees of freedom to maximize entropy production, then it would maximize the range of luminosity over which daisies can grow.

Future Work

At a planetary level, it is not immediately obvious how individual organisms would self-organize in such a way as to lead to situations observed in this model. Given the 'choice' of evolutionary or thermodynamic maximizing principles, there seem to be no initial reasons to think that life would adhere to the latter. One very important mechanism absent from the two-box model is evolution. The daisies have fixed responses and effects on environmental variables. Finding real-world organisms in states that appear to maximise the rate of entropy production via metabolic processes may be the result of such states being more efficient than lower entropy producing states. A more efficient organism may be more fitter and so natural selection rather than thermodynamics would be the mechanism that explains how such states arose and persist. However, a recent study has shown how Daisyworld regulation can emerge via evolutionary dynamics with minimal assumptions (McDonald-Gibson et al., 2008). An intriguing next step would be to develop new models that incorporate evolutionary mechanisms into energy balance models in order to assess the relationships between entropy production, self-regulation and evolutionary dynamics. This would combine evolutionary and thermodynamic mechanisms and so build a potentially more complete picture of the Earth's climate.

Conclusion

A simple two-box Daisyworld model has been presented. It has been shown that this model can be regarded as an example of an energy balance climate model. Energy balance models have been used to explore the hypothesis that planetary atmospheres maximise the rate of entropy production via the transport of heat from the hot tropics to the cold poles. In the two-box Daisyworld model, the difference in zonal temperature is produced by a difference in albedo rather than latitude. It was found that when the amount of diffusion was adjusted to maximize the rate of entropy production, the range over which any and both daisies grow was maximised and consequently the viability range of life on the planet was maximised. Maximising the rate of entropy production led to a maximisation of the range of luminosities over which self-regulation is observed. It is speculated that developing new models that incorporate thermodynamic and evolutionary dynamics could produce new results that have direct applicability to the Earth and its climate.

Acknowledgements

The author would like to thank Inman Harvey for suggesting the maximising demon analogy and significantly supporting the development of a number of ideas contained herein and James Lovelock for stimulating discussions on Daisyworld and climate modeling.

References

Ackland, G. J. (2004). Maximization principles and daisyworld. *J. Theor. Biol.*, 227(1):121–128.

Dewar, R. (2003). Information theory explanation of the fluctuation theorem, maximum entropy production and self-organized criticality in non-equilibrium stationary states. *J. Physics A*, 36(631-641).

Dyke, J. G. and Harvey, I. R. (2005). Hysteresis and the limits of homeostasis: from daisyworld to phototaxis. In Capcarrere, M., Freitas, A., Bentley, J., Johnson, C., and Timmis, J., editors, *Proceedings of VIIIth European Conference on Artificial Life, ECAL 2005*, pages 241–246. Springer-Verlag.

Dyke, J. G. and Harvey, I. R. (2006). Pushing up the daisies. In Rocha, L., Yager, L., Bedau, M., Floreano, D., Goldstone, R., and Vespignani, A., editors, *Artificial Life X, Proceedings of the Tenth International Conference on the Simulation and Synthesis of Living Systems*, pages 426–431. MIT Press.

Harvey, I. R. (2004). Homeostasis and rein control: From daisyworld to active perception. In J., P., Bedau, M., P., H., T., I., and R.A., W., editors, *Proceedings of the Ninth International Conference on the Simulation and Synthesis of Living Systems, ALIFE'9*, pages 309–314. MIT Press, Cambridge MA.

Lorenz, R. D., Lunine, J. I., and Withers, P. G. (2001). Titan, mars and earth: Entropy production by latitudinal heat transport. *Geophysical Research Letters*, 28(3):415–418.

Lovelock, J. E. (1983). Daisy world - a cybernetic proof of the gaia hypothesis. *The Co-evolution Quarterly*, Summer:66–72.

Lovelock, J. E. (1995). *The Ages of Gaia - A Biography of Our living Earth*. Oxford University Press.

Martyushev, L. M. and Seleznev, V. D. (2006). Maximum entropy production principle in physics, chemistry and biology. *Physics Reports*, 426(1-45).

McDonald-Gibson, J., Dyke, J. G., Di Paolo, E., and Harvey, I. R. (2008). Environmental regulation can arise under minimal assumptions. *J. Theor. Biol.*, 251(4):653–666.

North, G. R., Cahalan, R. F., and Coakley Jr, J. A. (1981). Energy balance climate models. *Reviews of Geophysics*, 19:91–121.

Ozawa, H. A., Ohmura, A., Lorenz, R. D., and Pujol, T. (2003). The second law of thermodynamics and the global climate system: A review of the maximum entropy production principle. *Reviews of Geophysics*, 41(4):1018.

Paltridge, G. W. (1975). The steady-state format of global climate systems. *Q. J. R. Meterol Soc.*, 104:927–945.

Prigogine, I. (1962). *Introduction to non-equilibrium thermodynamics*. Wiley Interscience, New York.

Pujol, T. (2002). The consequences of maximum thermodynamic efficiency in daisyworld. *J. Theor. Biol.*, 217:53–60.

Tonaizzo, T., Lenton, T. M., Cox, P. M., and Gregory, J. (2004). *Non-equilibrium Thermodynamics and the Production of Entropy: Life, Earth and Beyond*, chapter Entropy and Gaia: Is there a link between MEP and self-regulation in the climate system?, pages 223–239. Springer.

Wood, A. J., Ackland, G. J., Dyke, J. G., Williams, H. T. P., and Lenton, T. M. (2008). Daisyworld: a review. *Reviews of Geophysics*, 46.

Group formation and social evolution: a computational model

Nicholas Geard and Seth Bullock

SENSe, School of Electronics and Computer Science, University of Southampton, SO17 1BJ
nlg@ecs.soton.ac.uk

Abstract

The tendency to organise into groups is a fundamental property of human nature. Despite this, many models of social network evolution consider the emergence of community structure as a side effect of other processes, rather than as a mechanism driving social evolution. We present a model of social network evolution in which the group formation process forms the basis of the rewiring mechanism. Exploring the behaviour of our model, we find that rewiring on the basis of group membership reorganises the network structure in a way that, while initially facilitating the growth of groups, ultimately inhibits it.

Introduction

Groups, it has been argued, are a "basic process of social interaction" (Turner et al., 1987). Individuals rely on groups to achieve ends they could not achieve alone, and as a means for defining their personal identity. Groups, meanwhile, exist only so long as individuals are interested in becoming members of them. Much attention has recently been devoted to the task of identifying and understanding groups and communities in social networks.[1]

In sociology, the significance of groups as an expression of human social interaction, and their importance as an object of study, have a long history (Turner et al., 1987; Wasserman and Faust, 1994). The application of analytic tools from physics revitalised the study of social systems from a network perspective (Newman et al., 2002), and groups were recognised as a significant structural phenomenon, though one not amenable to easy characterisation (Jin et al., 2001; Davidsen et al., 2002; Girvan and Newman, 2002).

From a network perspective, a community is a subset of individuals who have more connections to other individuals within their community than to individuals from outside their community. A significant proportion of the literature on community structure focuses on the challenge of identifying the presence of communities in large data sets, such as those obtained from email records, automatic recommendation systems and social networking sites (Fortunato and Castellano, 2008, provide a recent overview of developments in this area).

A smaller fraction of the literature is concerned with the question of how communities arise, and understanding the social dynamics that influence their formation and evolution (Jin et al., 2001; Skyrms and Pemantle, 2000; Grönlund and Holme, 2004; Backstrom et al., 2006). A common feature of these models is that community structure frequently emerges as a side effect of another process, such as introductions between friends, or a desire to differentiate oneself from a population average.

In many real world contexts however, groups do not appear passively. Rather, they are the outcome of an active recruitment process, which arises in response to some perceived need that can best be met by a combined effort (Olson, 1971). For example, companies organise lobby groups in order to more effectively have their concerns heard by government, workers form unions to increase their bargaining power in negotiations with employers, and social movements arise to engage in collective action for a variety of humanitarian, environmental and other causes. We focus here on individuals and their participation in social movements (McAdam and Paulsen, 1993; Della Porta and Diani, 2005; Hedström, 2006).

Social movements are groups of people who come together to act collectively in support or opposition of some political or social issue (Tilly, 1978; Della Porta and Diani, 2005). It is widely accepted that social ties between individuals are critical to the success of social movements in recruiting new members (Snow et al., 1980; Marwell et al., 1988; McAdam and Paulsen, 1993). While some choices of group affiliation are undoubtedly a product of an individual's intrinsic preferences, the affiliations of their social contacts also exert an influence (Della Porta and Diani, 2005). Prop-

[1] The terms "group" and "community" are often used interchangeably in the literature; in the remainder of this paper, we will use "group" to refer to a subset of individuals in a population who each identify as belonging to a particular organisation, and "community" to refer to the a subset of individuals in the social network who are more densely linked to each other than to the remainder of the network.

erties of the social network, such as the number and intensity of ties between individuals, the existence of central nodes, and resource heterogeneity are therefore important determinants of how effectively a social movement can grow, and hence its ability to achieve its aims (Marwell et al., 1988; Gould, 1993; Kim and Bearman, 1997).

At the same time, an individual's participation in activities associated with a particular social movement is likely to strongly influence the people they meet, and hence on the set of individuals with whom they may form social ties (Della Porta and Diani, 2005). Thus, there is a bidirectional relationship between the short term dynamic of group formation occurring on a social network, and the longer term dynamic of the evolution of the structure of that social network (Sayama, 2007; Gross and Blasius, 2008).

Existing studies of community structure in networks have typically focused on exploring how communities can emerge from individual level rules. The reciprocal influence that group formation dynamics may have on social network evolution has been hitherto neglected. We are not aware of any model that explicitly considers group formation as a process that may actively influence the evolution of social networks. However, several recent models of opinion formation and cooperation in networked systems do confront a similar issue with regard to the coevolution of network's structure and the dynamic processes occurring on that network (Guimerà et al., 2005; Holme and Grönlund, 2006; Santos et al., 2006; Kozma and Barrat, 2008).

Explicitly considering the relationship between group formation and social evolution raises two interesting questions: how does social network structure influence the effectiveness of group formation, and how does group formation influence the evolution of the social network? In this paper, we propose a simple model of group formation and social network evolution and investigate the extent to which a group formation process can bring about (or hamper) the emergence of structural conditions contributing to its success; that is, the speed and size with which a group can recruit members.

Model Description

We model a social network as a simple graph containing N vertices representing individuals, and M undirected edges representing social ties (*i.e.,* each vertex has $K = 2M/N$ neighbours on average). Each vertex is associated with a trait vector $\boldsymbol{a}$. Each component of this vector is a continuous variable in the range $[0, 1]$ reflecting some aspect of an individual's social character (Watts et al., 2002; Boguñá et al., 2004); for example, their tendency to adopt a liberal or conservative stance on a particular social or political issue. Two individuals with similar values in a particular component of their trait vector will tend to share similar opinions on a particular issue. Viewed together, the totality of an individual's views describes a vector in an abstract social space. The social distance between two individuals x and y may then be calculated either in terms of the Euclidean distance between the vectors $\boldsymbol{a}_x$ and $\boldsymbol{a}_y$, or, with respect to issue n, the absolute difference between a_{xn} and a_{yn}, where a_{xn} is the nth component of $\boldsymbol{a}_x$.

Our model is updated on two distinct time scales: a short time scale corresponding to group formation, and a longer time scale corresponding to social evolution, in which each step represents a complete iteration of the group formation process.

Group formation phase: The group formation process follows the following sequence of steps:

1. G individuals are picked uniformly at random to seed G different groups. These individuals are added to a set of *active* individuals, A.
2. A single individual i_x is randomly chosen from A. This individual issues invitations to all of their network neighbours who are not currently affiliated with any group to join their group. The individual i_x is then removed from A.
3. Each individual i_y who receives an invitation accepts it with a probability equal to $\alpha(1 - |a_{xn} - a_{yn}|)$, where α is a model parameter governing the base probability of acceptance, and n is the index of the group to which i_x belongs. Therefore, if the nth component of trait vectors associated with i_x and i_y are identical, the distance between them will be zero, and the probability of acceptance will be α. As the difference between traits increases, the probability of acceptance decreases linearly.
4. Individuals who accept invitations are added to A.
5. Steps 2–4 are repeated until there are no individuals remaining in A.

At this point, the network can be in one of two states: either all individuals are members of a group, or the group formation process has died out before spreading through the entire network because all individuals on the periphery of a group have had their initiations to join refused. The probability of this occurring will depend on the value of α and structural features of the network, such as the density of edges (Figure 1). In order to ensure some variability that can be ascribed to network structure, we typically chose values of K and α that placed the initial network in the boundary region of Figure 1, where the group formation process was able to spread some distance beyond the seed individual, but did not percolate across the entire network.

Social update phase: After group formation has concluded, individuals who have joined a group adjust their social ties. We assume that being a member of a group

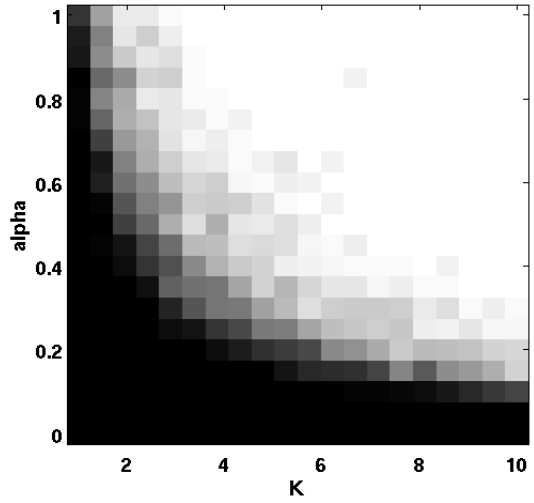


Figure 1: Proportion of network ($N = 2,000; G = 1$) becoming members of a group (black = 0%; white = 100%) for a range of values of α and K. Each parameter combination was repeated 200 times with randomly chosen seeds and the final group sizes were averaged. For low values of K and/or α, groups rarely grow beyond a few members. When both K and α are high, all individuals in the network join the group.

entails involvement in group-related activities that will result in an individual spending more time with members of their group (irrespective of whether they were previously known to them) and hence, given finite time, less time with current acquaintances who are not members of their group. We make the further assumption that individuals involved in groups are likely to update thier social neighbourhoods more frequently than unaffiliated individuals.

Each individual who is a member of a group therefore drops the edge connecting them to their least similar neighbour (irrespective of whether that neighbour is in their group or not) and creates a new edge connecting them to the member of their group, who is not currently a neighbour, to whom they are most similar. Similarity is again measured as the difference between the nth component of the respective trait vectors (*i.e.,* that corresponding to the group of which they are members).

After the social update phase has occurred, all groups are cleared, and the next iteration of the group formation phase begins on the new social network, with a new set of randomly chosen group seeds. Social movements often form in response to a particular issue, and either break apart or evolve into a new form as that issue becomes less relevant (Della Porta and Diani, 2005; Fuchs, 2006). Our decision to break apart all groups between each iteration of group formation is clearly a coarse approximation of this situation, but was chosen for initial simplicity.

Model Behaviour

To begin, we consider an initially random network with $N = 2,000$ and $K = 6$ in which only one group is formed during each iteration ($G = 1$) and trait values are drawn at random from a uniform distribution. In the simulations described here, these edges are initially randomly distributed between vertices following the Erdős-Rényi random graph model; however, other initial configurations are possible. This section describes the behaviour in an individual simulation run in detail, before exploring the sensitivity of this behaviour to K, α and the initial network structure.

The behaviour we are interested in observing is how the size of groups formed changes as the social network evolves. The size of the group formed depends not only on the global structure of the network, but also on the local neighbourhood of the seed individual. To obtain an indication of the general propensity of a particular network structure to facilitate group formation, we measure the average size of the group formed across fifty random seedings, with groups being erased after each. The social update phase is then carried out based on the group formed from the final of these seedings.

The structure of the social network passes through three distinct periods of evolution (Figures 2 and 3). Initially, the network is well connected but disordered (Figure 2, top panel), and the mean trait difference between neighbours is high (0.329). As a consequence, invitations have a low probability of being accepted and the resulting groups are small (12.42 members—0.62% of the population—on average over the first 10 iterations of the simulation). Mean clustering coefficient and path length both remain low (approximately 0.04 and 7.2 respectively), as is typical of a random graph.

However, the groups that do form enable their local regions of the network to become more ordered, by allowing individuals with similar trait vectors to increase the density of their interconnections, (Figure 2, middle panel). By doing so, they increase the probability of future invitations between individuals in this region being accepted and so assist the formation of groups in subsequent iterations.

Surprisingly, rather than produce a steady increase in the average size of groups as the social network becomes more ordered, a phase transition occurs at the point where a large proportion of the network simultaneously becomes well organised (Figure 2). Mean group size increases dramatically, peaking at 438.12 members (21.9% of the population) in iteration 75 (Figure 3). Mean clustering coefficient increases by an order of magnitude to approximately 0.5 by iteration 80, while mean path length remains relatively low and the degree distribution becomes more skewed, properties indicative of small world structure.

A side effect of larger groups forming is that the rate of network reorganisation increases (as each individual who is in a group updates one of their social ties). Furthermore,

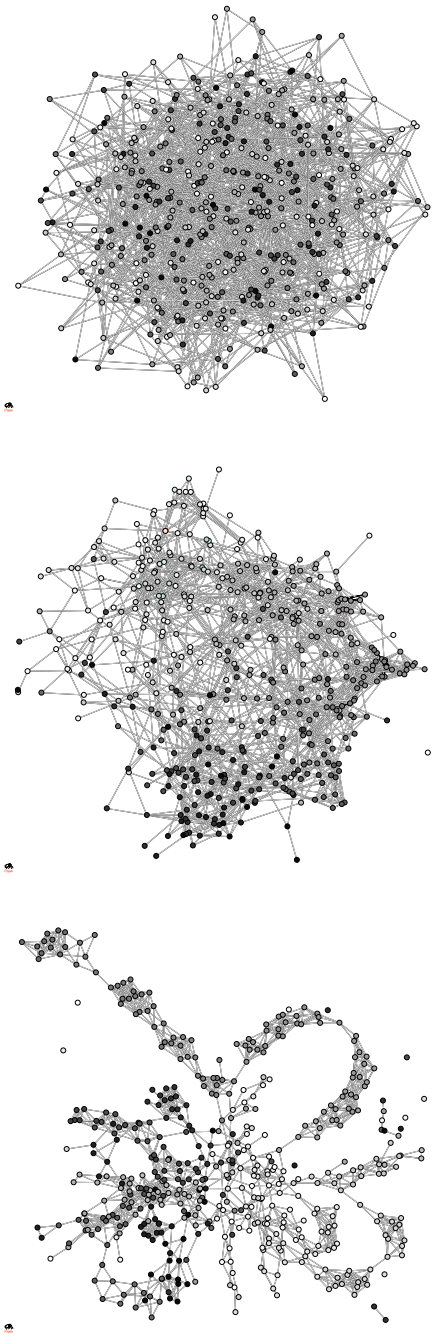


Figure 2: Network structure observed at different points in evolution: (a) the initial random network, with trait values dispersed throughout; (b) the network at the point of phase transition, when group formation spreads rapidly between neighbours with similar trait values; and (c) the network at the end of a run, with individuals clustered into weakly connected communities. Note that smaller networks ($N = 500$) are shown for clarity; however, their qualitative features are otherwise similar.

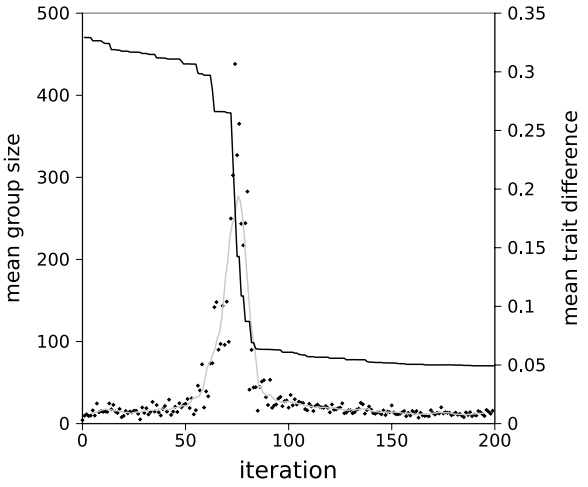


Figure 3: A representative run of the model initialised with a random network ($N = 2000, K = 6, \alpha = 0.25$). Each symbol represents the mean group size observed over 20 random seedings (as described in the text), together with a moving average calculated over 10 iterations (gray line) and mean trait difference between neighbouring nodes (black line). The three networks in Figure 2 correspond to networks observed prior to, during, and after, the spike in mean group size.

each individual is now able to select their new neighbour from a wider pool of potential candidates (their fellow group members). The mean trait difference between neighbours drops (to 0.049, Figure 3) and the network begins to partition into a number of weakly connected communities (Figure 2, bottom panel, and Figure 3). Around iteration 90, mean path length begins to increase steadily, reaching approximately 14 by iteration 200.

In the extreme case, the network may disintegrate completely into a set of disconnected components; however, this is not required in order for group size to fall: by iteration 200, 94.4% of individuals still belonged to a single connected component. The appearance of community structure sufficient to hamper the formation of groups by creating bottlenecks that impede the spread of invitations. If there is only a single link between two communities, then, even if it is between two very similar individuals, group membership has a chance of spreading at best equal to α.

This social evolution dynamic was observed across a range of parameter settings, with the primary differences being the time required for the network to organise, and the maximum size to which groups are able to grow (Figures 4 and 5). As K and/or α increase, the size of groups that form throughout each simulation run also increases, in line with the trend illustrated in Figure 1. For all combinations of K and α, the peak group size achieved is substantially greater

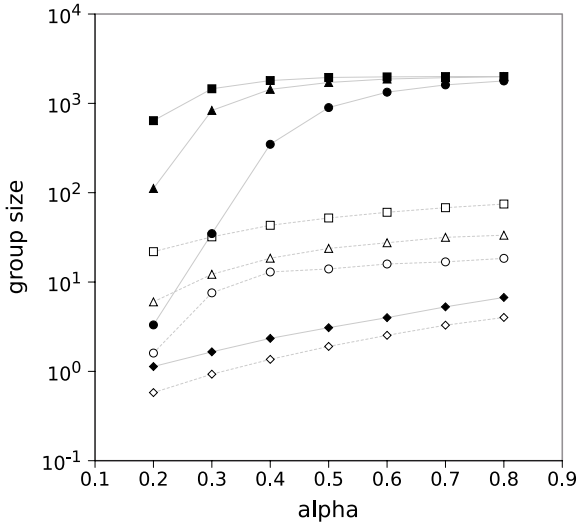


Figure 4: Peak (filled) and average (hollow) group sizes for various values of α and K (diamond: 2; circle: 4; triangle: 6; square: 8). Note that the Y-axis is log-scaled. Other model parameters: $N = 2,000; G = 1$. Each data point is averaged over 20 runs. The peak group size is that obtained during the phase transition in network structure. Average group size is that resulting after the phase transition has occurred.

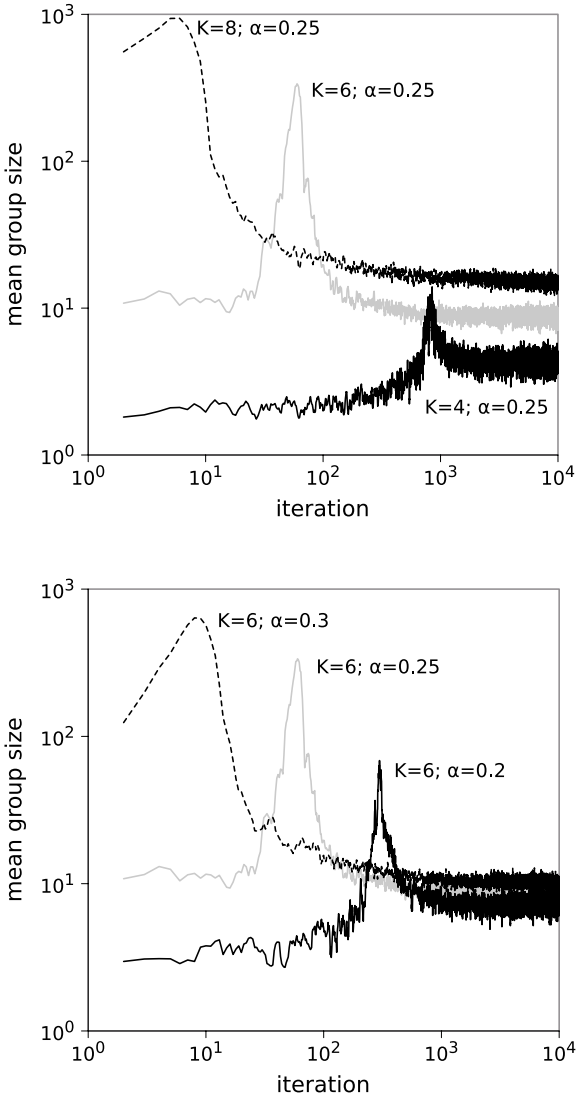


Figure 5: Mean group size trends for various values of K (top) and α (bottom). For clarity, a moving average over 5 iterations is shown, rather than individual data points. Note that both axes are log-scaled. Increasing either the density of connections (K) or the base probability of invitations being accepted (α) increases both the speed with which the network organises, and the peak group size that can be achieved.

than the average. Furthermore, increasing K and α results in the peak group size being obtained earlier in the simulation run.

We investigated the effect of the initial network configuration (Figure 6) on social evolution by varying the rewiring probability p used to create the initial network. In comparison to random graphs ($p = 1.0$), regular lattices ($p = 0.0$) with comparable N and M take considerably longer to organise and, at their peak, result in smaller groups. In many simulation runs (such as that shown in Figure 6), no peak phase occurs, and the network transitions directly to the disjoint community phase. Small world networks ($p = 0.05$) organise more slowly than random graphs, but otherwise behave similarly, and the occurrence of a peak phase is more reliable than in lattices.

We have also carried out preliminary investigations into the behaviour of the model when there is more than one group forming in each iteration ($G > 1$). In this case, competition between groups appears to lead to a "rich get richer" process, whereby large groups tend to increase in size, at the expense of smaller groups. The mechanism responsible for this is straightforward: once one group begins to increase in size with respect to the others, its members dominate the set of active individuals (A), and hence benefit from more frequent opportunities to recruit unaffiliated individuals.

Discussion

How can we interpret the pattern of social evolution observed in our model? As communities begin to emerge, the ability of groups to recruit large numbers of people initially improves. However, as these communities become stronger, they also become more homogeneous and detached from the wider social context in which they exist. This social isola-

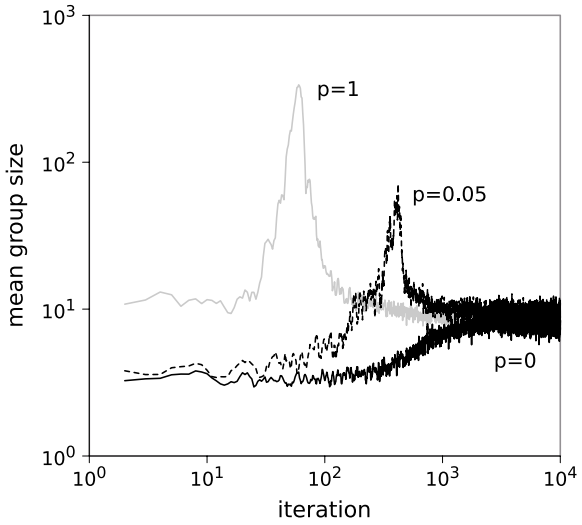


Figure 6: Mean group size trends for various initial network configurations (rewiring probabilities, $p = 0, 0.05$ and 1). For clarity, a moving average over 5 iterations is shown, rather than individual data points. Note that both axes are log-scaled. See text for discussion of trends.

tion severely limits the ability of groups to recruit new members and hence, potentially, to achieve their aims in an effective fashion (Snow et al., 1980). Networks with a higher density of social ties are more rapidly reorganised to facilitate, and later inhibit, group formation. Similarly, populations in which people have a strong predisposition toward joining groups reorganise more rapidly.

There is general evidence that segregation of social networks can arise despite the absence of any explicit preference for such an outcome (Schelling, 1971). Even when interaction structures are externally imposed, such as the hierarchical reporting relationships of a large organisation, there is evidence to suggest that the existence of communities can have a negative effect on global integration (Kilduff and Tsai, 2003). In the context of organisations, such an effect has led to the value placed upon *bridges*—individuals who fill structural holes in a network by linking otherwise disconnected components. Individuals in such positions often gain social capital from the role they play in mediating between different interest groups (Burt, 2002).

Social movements, too, can benefit from being linked together. Della Porta and Diani (2005) summarise extensive evidence suggesting that linkages between social movements allow sharing of information and resources, and facilitate cooperation and coordination of the aims of different movements. A key factor in linking movements is overlapping memberships—the existence of individuals who are members of two or more groups (Palla et al., 2005). This suggests that our assumption of exclusive group membership will require reevaluation. One promising direction for future work is to allow individuals to belong to multiple groups at the same time, and to explore the extent to which this enables the social network to organise in such a way that it facilitates the formation of groups without disintegrating into weakly connected components.

In summary, this paper has presented a novel model of group formation and social evolution that takes as its starting point two main ideas: first, that group formation is a process in which individuals actively seek to engage, and second, that this tendency has repercussions for the evolution of social network structure. The investigations reported here indicate that rewiring on the basis of group membership reorganises the network structure in a way that, while it initially benefits the growth of groups, ultimately inhibits it.

It is worth noting that, in order to remain as simple as possible, the model described here makes several assumptions that may limit its general applicability. For example, an individual's decision to join a particular group is based purely upon their similarity with the individual who has invited them, and does not take into account factors such as the alignment of their values with those of the group, or the opinions of their social neighbours (McAdam and Paulsen, 1993). Group membership is exclusive; that is, it is not possible for an individual to simultaneously be a member of more than one group which, as discussed above, is likely to be play a role in ensuring social cohesion (Della Porta and Diani, 2005; Palla et al., 2005). Despite these limitations of the model in its current formulation, we believe it to be a fruitful starting point for further exploration into the the co-evolution of topology and dynamics in social networks.

References

Backstrom, L., Huttenlocher, D., Kleinberg, J., and Lan, X. (2006). Group formation in large social networks: membership, growth, and evolution. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 44–54.

Boguñá, M., Pastor-Satorras, R., Díaz-Guilera, A., and Arenas, A. (2004). Models of social networks based on social distance attachment. *Phys. Rev. E*, 70(5):056122.

Burt, R. S. (2002). The social capital of structural holes. In Guillén, M. F., editor, *The New Economic Sociology: Developments in an Emerging Field*, pages 148–190. Russell Sage Foundation, New York, NY.

Davidsen, J., Ebel, H., and Bornholdt (2002). Emergence of a small world from local interaction: Modeling acquaintance networks. *Phys. Rev. Lett.*, 88:128701.

Della Porta, D. and Diani, M. (2005). *Social Movements: An Introduction*. Wiley Blackwell, Oxford, 2nd edition.

Fortunato, S. and Castellano, C. (2008). Community structure in graphs. To appear in Springer's *Encyclopedia of Complexity and System Science*. arXiv:0712.2716v1.

Fuchs, C. (2006). The self-organization of social movements. *Systemic Practice and Action Research*, 19(1):101–137.

Girvan, M. and Newman, M. E. J. (2002). Community structure in social and biological networks. *Proc. Nat. Acad. Sci. USA*, 99:7821–7826.

Gould, R. V. (1993). Collective action and network structure. *American Sociological Review*, 58(2):182–196.

Grönlund, A. and Holme, P. (2004). Networking the seceder model: Group formation in social and economic systems. *Phys. Rev. E*, 70(3):036108.

Gross, T. and Blasius, B. (2008). Adaptive coevolutionary networks: a review. *Journal of the Royal Society Interface*, 5:259–271.

Guimerà, R., Uzzi, B., Spiro, J., and Amaral, L. A. N. (2005). Team assembly mechanisms determine collaboration network structure and team performance. *Science*, 308:697–703.

Hedström, P. (2006). Explaining the growth patterns of social movements. In Elster, J., Gjelsvik, O., Hylland, A., and Moene, K. O., editors, *Understanding Choice, Explaining Behaviour*. Oslo Academic Press, Oslo.

Holme, P. and Grönlund, A. (2006). Nonequilibrium phase transition in the coevolution of networks and opinions. *Phys. Rev. E*, 74:056108.

Jin, E. M., Girvan, M., and Newman, M. E. J. (2001). The structure of growing social networks. *Phys. Rev. E*, 64:046132.

Kilduff, M. and Tsai, W. (2003). *Social Networks and Organizations*. Sage Publications, London.

Kim, H. and Bearman, P. S. (1997). The structure and dynamics of movement participation. *American Sociological Review*, 62(1):70–93.

Kozma, B. and Barrat, A. (2008). Consensus formation on coevolving networks: groups' formation and structure. Submitted to *J. Phys. A: Math. Gen.* arXiv:0801.4860v1.

Marwell, G., Oliver, P. E., and Prahl, R. (1988). Social networks and collective action: A theory of the critical mass III. *The American Journal of Sociology*, 94(3):502–534.

McAdam, D. and Paulsen, R. (1993). Specifying the relationship between social ties and activism. *The American Journal of Sociology*, 99(3):640–667.

Newman, M. E. J., Watts, D. J., and Strogatz, S. H. (2002). Random graph models of social networks. *Proc. Nat. Acad. Sci. USA*, 99:2566–2572.

Olson, M. (1971). *The Logic of Collective Action: Public Goods and the Theory of Groups*. Harvard University Press, Cambridge, MA, 2nd edition.

Palla, G., Derényi, I., Farkas, I., and Vicsek, T. (2005). Uncovering the overlapping community structure of complex networks in nature and society. *Nature*, 435:814–818.

Santos, F. C., Pacheco, J. M., and Lenaerts, T. (2006). Cooperation prevails when individuals adjust their social ties. *PLoS Comput. Biol.*, 2(10):e140.

Sayama, H. (2007). Generative network automata: A generalized framework for modeling complex dynamical systems with autonomously varying topologies. In *Proceedings of the First IEEE Symposium on Artificial Life*, pages 214–221. IEEE Press.

Schelling, T. C. (1971). Dynamic models of segregation. *Journal of Mathematical Sociology*, 1:143–186.

Skyrms, B. and Pemantle, R. (2000). A dynamic model of social network formation. 97(16):9340–9346.

Snow, D. A., Zurcher, L. A., and Ekeland-Olson, S. (1980). Social networks and social movements: A microstructural approach to differential recruitment. *American Sociological Review*, 45:787–801.

Tilly, C. (1978). *From Mobilization to Revolution*. Addison-Wesley, Reading, MA.

Turner, J. C., Hogg, M. A., Oakes, P. J., Reicher, S., and Wetherell, M. S. (1987). *Rediscovering the Social Group: A Self-Categorization Theory*. Blackwell Publishing, Oxford.

Wasserman, S. and Faust, K. (1994). *Social Network Analysis*. Cambridge University Press, Cambridge.

Watts, D. J., Dodds, P. S., and Newman, M. E. J. (2002). Identity and search in social networks. *Science*, 296:1302–1305.

A Computational Model of Gene Regulatory Networks and its Topological Properties

Ângela Gonçalves[2,3] and Ernesto Costa[1,2]

[1] Department of Informatics Engineering, University of Coimbra
[2] Centre of Informatics and Systems of the University of Coimbra
[3] European Space Agency, Rome
angela.goncalves@esa.int

Abstract

A new model for Gene Regulatory Networks (GRN) is proposed. The model is potentially more biologically sound than other approaches, and is based on the idea of an artificial genome from which several products like genes, mRNA, miRNA, non-coding RNA, and proteins are extracted. These products are connected giving rise to a heterogeneous directed graph. The topology of the obtained networks is studied using degree distributions. We make some considerations about the biological meaning of the outcomes of these simulations.

Introduction

Sequencing the human genome was a tremendous breakthrough, but today's great challenge is deciphering how genes determine the phenotypic traits of an organism and how the genome controls the development of organisms. Although biology's central dogma explains the basic process of gene expression into protein phenomena like cellular differentiation, the ability of cells with the same genetic information to behave differently according to their function in the organism, is not accounted for in the dogma. The answers to such questions lie in complex networks of interactions, known as regulatory networks, between genes and other molecules including proteins, the very products of gene expression. Regulatory networks are highly non-linear and have thousands of variables: finding a computational model for them is a difficult albeit important task. Various approaches for modeling gene regulatory networks (GRNs) appeared in the last decades focusing on regulation at transcription level, the best known form of regulation. However, recent studies revealed that regulation occurs at any stage of protein synthesis including transcription, RNA processing, mRNA decay, translation and post-translation. In this paper we propose a new model for gene regulatory networks, called HeRoN, that introduces a level of biological detail that not present in previous models, and study the topological properties of the networks using degree distributions.

The paper is organized as follows: in section 2, we will give a brief explanation of the biological concepts of gene regulation; a review of different approaches is given in section 3, and, in section 4, our own model is presented; section 5, will describe the experimental setup and the topological aspects of GRNs based on degree distributions; some concluding remarks are presented in the last section.

Gene Expression Regulation

Gene expression can be decomposed in three stages: transcription, processing and translation. In the transcription phase, a RNA molecule is created by complementing the DNA sequence of the gene starting in a place called the promoter of a gene. Transcription also ends when a particular signal is found. After transcription the RNA transcript is processed and certain non-coding sequences, called introns, are removed. The remaining sequences are joined and form a mature mRNA molecule. This mRNA molecule is then translated into a protein according to a known relation called the genetic code.

The central dogma posit gene expression as a one-way process where information flows from the genes to the proteins. What actually happens in organisms is that genes, proteins, mRNAs and other types of molecules in the cells, are able to interact with each other given rise to a regulatory process, which can occur at any stage of expression. One of the most well-known regulation mechanism acts at the transcription initiation stage. This type of regulation consists in the binding of certain proteins, called transcription factors, to particular sequences in the genome physically helping or making impossible the initiation of transcription. With this mechanism some genes are able to regulate the expression of other genes or even themselves. Other type of regulation occurs at the transcription termination and is influenced by many types of molecules in a cell. The regulation processes is highly non-linear and in Gonçalves and Costa (2007a) we studied the dynamics characteristics of GRN, namely the emergence of three types of behaviors: fixed, periodic and chaotic. One of the goals of this paper is to study the static topological properties of GRN, aiming at acquiring some insights about biological aspects of the process of genetic regulation.

State of the Art

In recent years several models for Gene Regulatory Networks have been proposed. The majority of models make the simplifying assumption that the control of gene expression resides only in the regulation of gene transcription. Due to lack of space we only briefly mention some of the known models. For an in-depth description (see Gonçalves and Costa, 2007b)

One early and influencing discrete approach adopted a complex system view of the genome (Kauffman, 1993). Using Random Boolean Networks, Kauffman represented the regulatory system as a network of logical components connected at random. Despite the interesting insights of Kauffmans model it was unable to give much explanation for the regulatory mechanisms and, to many, did not exhibit sufficient parallels with the real networks due to its abstract nature (Reil, 1999). Rather than using a network for a base level representation, another discrete and promising model, the Artificial Genome, originally proposed by Reil (1999) used a more biological framework being based on a DNA-like sequence representing the genome from which the network structure could be extracted. A similar model was proposed by Banzhaf (2003) and described by Hallinan and Wiles (2004), Watson et al. (2004) and Willadsen and Wiles (2003). In the original Artificial Genome model a random string of bases is generated to represent the genome of an organism. The string is searched for promoter sequences which are, by convention, '0101'. The six digits following the promoter will represent the gene sequence (see Figure 1A). The sequence between genes will be the regulatory region for the following gene. An operation is applied to the gene sequence to create a gene product. This operation represents the entire expression process and consists of incrementing each gene's digit by one, modulo 4 (the number of bases) (see Figure 1B). The resulting sequence is the gene product which will be used to search for matches in the regulatory regions of all genes (see Figure 1C). Each match represents a regulatory link between the gene that originated the gene product and the gene regulated by the region where the match occurred. Whether the regulation is inhibitory or excitatory depends on the value of the last digit of the gene product. After performing the matches a regulatory network can be extracted and displayed in the form of a graph. The Artificial Genome still comprise many simplifications, e.g., the merging of the entire process of gene expression, presenting no intermediate products and determining an arbitrary operation for the creation of a gene product.

Several models have been proposed that treat variables as continuous values and calculate them through differential equations. The Additive Regulation Model, and models based on the S-System power law are some examples. In the Additive Regulation Model (D'Haeseleer, 2000) variables are continuous and updated synchronously. This model can be represented as a matrix of positive, negative or zero connections (see Figure 2).

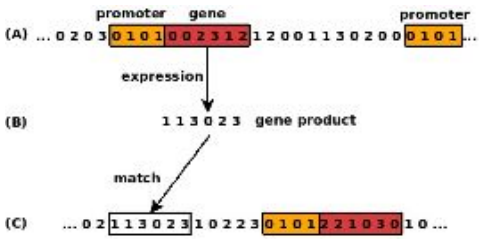


Figure 1: Artificial Genome model.

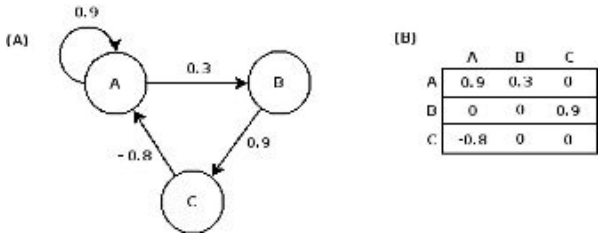


Figure 2: A Graph of a Regulatory Network

When a matrix entry is nonzero, there is a regulatory connection from gene product i to gene product j. If the entry is positive the regulation is enhancing and if it is negative the regulation is repressive. The expression level of each gene x_i could be given by the weighted sum of all variables:

$$\frac{dx_i}{dt} = S(\sum_j w_{ji}x_j + b_i) - D_i x_i \qquad (1)$$

with x_i the expression level of the ith variable, b_i a bias term that indicates if the gene is expressed in the absence of regulatory inputs, w_{ji} the weight in the matrix from gene j to gene i, $S()$ a sigmoidal function and D_i the decay rate of gene i.

Finally, an S-system is a parameterized set of nonlinear differential equations:

$$\frac{dx_i}{dt} = \alpha_i \prod_{j=1}^{n} x_j(t)^{g_{ij}} - \beta_i \prod_{j=1}^{n} x_j(t)^{h_{ij}} \qquad (2)$$

where x_i is the expression level of gene i, n is the number of network components, $\alpha_i \geq 0$, $\beta_i \geq 0$ are rate constants and g_{ij}, h_{ij} represent the interactive affectivity of x_j to x_i. The first product describes all influences that are excitatory (increase x_i) and the second product all influences that are inhibitory (decrease x_j). These systems have a rich structure but number of parameters that have to be estimated is large (Noman and Iba, 2005).

With the continuous models just described biologically plausible features such as decay rates of molecular products

(D'Haeseleer, 2000) can be included and reverse engineering/learning algorithms can be used to determine their parameters from real data (Ando and Iba, 2001; Sakamoto and Iba, 2001; Noman and Iba, 2005), however, as with the discrete models discussed, the blackbox approach of the process they use makes understanding the mechanisms of gene regulation at the various levels more difficult.

The above considerations prompted the creation of a new model called HeRoN with a string based framework similar to the Artificial Genome breaking the process down to its important steps and overcoming some of its simplifications. The networks derived by the HeRoN model can be represented by a graph where the nodes represent the different products involved in the process of gene expression, thus heterogeneous, and the arcs establish the interactions between the products.

HeRoN: a Model for a Heterogeneous Gene Network

The proposed model HeRoN takes a string from a four symbol alphabet representing the genome and derives from it various products such as genes, proteins and some more intermediate products. The expression algorithm is a six-step process that will now be described with some detail.

1. Generate the genome The genome, implemented as a string of integers, is randomly generated given a size parameter. Each integer corresponds to a base: 0 - T(U), 1 - A, 2 - G, 3 - C.

2. Search the genome for genes and create them The genome is searched for given sequences that represent the gene promoters. In real biological systems there are some promoter sequences that appear in most genes of many organisms, called consensus sequences, and the more a sequence in a genome resembles them, the more efficient the transcription. To achieve this, a threshold symbolizing the binding strength between a RNA polymerase and the genome, was set as a parameter. A sequence in the genome, with the same size as the given promoter sequence, is considered to be a valid promoter in the genome when its percentage of match with the given sequence is equal or above the threshold. Each time a valid promoter is found the genome is searched for a termination sequence. When such termination sequence, chosen to be a poly-A sequence of adjustable size, has been found a gene is created. Each gene consists of a promoter sequence, the coding sequence and the regulatory region. The coding sequence is the region located between the promoter and the termination sequence. The regulatory region is the region located between the end of the previous gene (after its termination sequence) and the promoter (see Figure 3A).

3. Generate RNA transcript from the genes The RNA transcript is generated by complementing the bases on the coding sequence of the gene according to the pairing A-T and C-G. In the four integer alphabet 1 and 0 are the complement of each other as are 3 and 2 (Figure 3A-B).

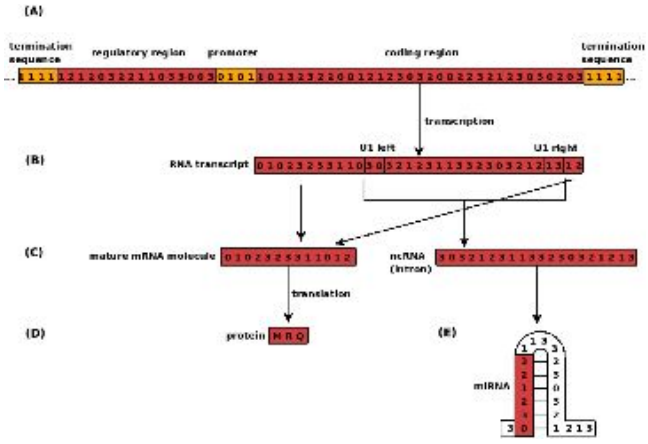


Figure 3: The HeRoN model. See text for an explanation.

4. Splice the RNA transcript which generates the mature mRNA and introns Splicing the RNA transcripts means that each RNA transcript is searched for introns that are removed from the sequence and stored into a list of components called ncRNAs. Introns are detected by means of two sequences, U1 left and U1 right, that simulate the role of U1 srRNA molecule that has two highly conserved consensus sequences complementary to the 5 and 3 ends of essentially all mRNA introns (Zhang and Rosbash, 1999). The new sequences created from the RNA transcripts with the introns removed are called mRNA (Figure 3B-C).

5. Translate the mRNA into proteins Each mRNA molecule is scanned for the start codon sequence (AUG). When this sequence is found the mRNA is read three bases at a time until a stop codon is found (UAA, UGA, or UAG). Each three bases are translated into one amino acid according to the genetic code table. The stop codon is not considered as part of the protein (Figure 3C-D).

6. Search the ncRNAs from miRNAs and create them The model incorporates a mechanism of RNA interference that regulates the stability of mRNA by triggering its degradation. This mechanism was added to the model when it was noticed that a large number of RNA transcripts did not produce proteins because they missed the start codon. Searches in biology literature for similar phenomena led to the subject of non-coding/junk DNA. Junk DNA has been a name given by researchers to large portions of DNA for which no function has yet been identified, including introns and large portions of intergenic sequences. Having found evidence that genes considered to be junk DNA have a regulatory influence (Martens et al., 2004) and that this kind of DNA makes up to 95% of chromosomes, researchers reversed their opinions on the usefulness of junk DNA, changing its name to non-coding DNA. In particular, the regulatory role of noncoding genes relates to the RNAi mechanism. This mechanism of transcriptional gene silenc-

ing is induced by the association between proteins and RNA. The resulting molecules are called small interfering RNA (siRNA), when they derive from exogenous sources (outside the cell), or are called microRNA (miRNA), when they are produced from non-coding genes in the cells own genome. miRNAs are short single-stranded RNA stretches of 21 to 23 nucleotides that are processed from primary transcripts known as pre-miRNA to short stem-loop structures called pre-miRNA and finally to functional miRNA (Gregory et al., 2006). The effect of this regulation mechanism is that while some genes are transcripted at a normal rate they are not expressed because they are degraded before they leave the nucleus. To incorporate this influence in the model it was determined that if the resulting protein has no sequence, because the mRNA misses the start codon, that mRNA molecule is considered to be non-coding and therefore is added to the ncRNA list where the introns were already stored. All ncRNAs are then scanned for hairpin loops with a minimum length. This indicates the presence of miRNAs that are then considered as another product in the model (Figure 3C-E).

From the expression algorithm to the network

The expression algorithm described above creates a list of products and stores their corresponding sequences and references to the products from which they derived. To extract the interaction network between these products it is necessary to determine the bindings between them, namely between proteins and genes and between miRNAs and genes. Finding the interactions between miRNAs and genes is simple since the two products are made of the same components, nucleotides, and their binding is a simple match between complementary sequences. The other type of binding involves elements that do not interact in a linear manner and are made up of different components, amino-acids and nucleotides. In biological systems the proteins ability to locate and bind with certain DNA sequences depends not only on the involved amino-acid and nucleotide sequences but also on the protein's three-dimensional structure and on the DNA double stranded structure. Many solutions exist that try to predict DNA-protein binding sites (Baker and Sali, 2001) and this is still an open topic in Bioinformatics. In addition to these approaches some authors find it important to examine the individual interactions between the amino-acids and the nucleotides since underlying the bindings are the discrete interactions between them (Hoffman et al., 2004). Databases such as the Amino Acid-Nucleotide Interaction Database (AANT) categorize amino-acid-nucleotide interactions from experimentally determined protein-nucleic acid structures. In our model a protein and a DNA sequence are perfectly aligned and the statistical table of the entire AANT database along with a binding threshold is used to determine if they bind. For each amino-acid in the protein its binding probability with the corresponding nucleotide in the DNA is given by the AANT statistic table. Given the interactions, one of four methods, called average, maximum, minimum and random, is used to compare them with the threshold (see Figure 4).

Aminoacid	A(%)	C(%)	G(%)	T(%)
Alanine (Ala, A)	24.2	17.3	24.0	24.6
Arginine (Arg, R)	19.6	24.1	35.7	12.2
Asparagine (Asn, N)	25.5	20.0	23.9	17.7
Aspartate (Asp, D)	13.3	34.2	37.0	1.5
Cysteine (Cys, C)	29.1	18.8	24.8	23.1
Glutamine (Gln, Q)	28.0	17.7	29.4	13.7
Glutamate (Glu, E)	19.1	34.8	33.0	4.8
Glycine (Gly, G)	20.1	22.9	32.1	17.0
Histidine (His, H)	25.3	16.2	37.7	14.2
Isoleucine (Ile, I)	21.4	26.4	30.8	11.4
Leucine (Leu, L)	9.5	31.1	30.2	19.4
Lysine (Lys, K)	23.7	22.8	30.7	16.3
Methionine (Met, M)	22.1	27.9	22.1	9.8
Phenylalanine (Phe, F)	17.7	24.1	40.5	17.7
Proline (Pro, P)	37.0	11.0	21.0	2.0
Serine (Ser, S)	28.2	20.9	27.2	19.7
Threonine (Thr, T)	24.6	20.2	27.8	23.1
Tryptophan (Trp, W)	14.4	30.2	24.8	21.8
Tyrosine (Tyr, Y)	28.4	27.4	23.6	15.0
Valine (Val, V)	25.0	35.3	20.0	1

Table 1: Statistical table of the entire AANT database. Along with the name of the amino-acids are the conventional three-letter and one-letter abbreviations.

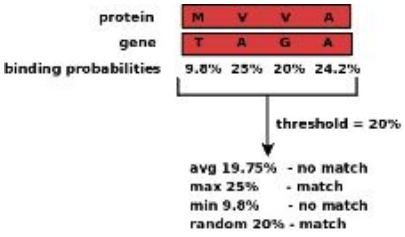


Figure 4: Protein binding example. Each amino-acid-nucleotide pair is searched for in the AANT statistic table (see Table 1 for a complete description).

If using the average method an average of all the probabilities is calculated. For the maximum and minimum methods, the respective maximum or minimum probability is chosen. For the random method the probability of a random amino-acid-nucleotide pair, from the sequence, is chosen. In the example of Figure 4 it was the V-G pair. The components are said to bind if the resulting probability is above or equal to the threshold.

The information gathered about the interaction between the components is then used to create a graph representa-

tion of the network where each gene creates many products, most of them ncRNAs and a single mRNA molecule. Each mRNA either creates a protein or a miRNA molecule, miRNA molecules can also derive from ncRNAs. All connections starting at a miRNA molecule end at an mRNA molecule and are repressive, while connections between proteins and genes can be either activating or repressive (see Figure 5).

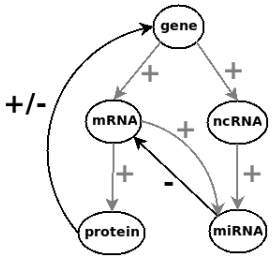


Figure 5: Activation/deactivation relations between the different products. The positive and negative signs near the edges represent, respectively, activation or deactivation of a product. The black colored edges represent the regulatory connections while the grey edges represent the "creation" of a product.

Experimental Setup and Results

Now that the model has been described it is time to present the experimental study that was carried out. Here we will be concerned only with the topological properties of GRNs.

Parameter	Used values
genome size	20000, 100000 and 500000
miRNA binding site size	4, 5, 6 and 7
inhibition rate	0, 0.25, 0.50 and 0.75
binding threshold	29, 32, 33 and 34
binding choice	avg, max, min and rand

Table 2: Variable parametrization

Table 2 shows the variable parameterization used throughout the experiments performed. The fixed parameters are: sequence 0101 for the promoter, promoter match of at least 75%, sequence 1111 for the termination sequence, and a binding site size of 6 for the proteins. Experiments were run for all possible combinations of the 'used values' mentioned in Table 2 . Each combination of the variable parameters was run 10 times. The initial set of active genes for each of the runs was randomly determined from a uniform distribution.

Topology of the obtained networks

The different topology classes of networks, i.e., regular lattice, small-world and random networks, arise from the different ways large sets of elements connect. A network where each node is connected to its nearest spatial neighbors is the so called regular lattice. Starting with a regular lattice and randomly rewiring a portion of the links creates small-world networks. At the extreme random networks are formed, where every pair of elements is connected at random. Like most social and biological networks, such as the World Wide Web, the immune system, the brain and ant colonies, to name just a few examples, genetic regulation networks possess certain non-trivial topological features. For instance, while nodes on regular lattices have constant degree and ordinary random networks have Poisson degree distributions, it is found that many real-world networks have degree distributions measurably different from these. This strongly suggests that there are features of such networks that would be missed if they were to be approximated by an ordinary random graph or lattice (Newman et al., 2001), thus many recent works on real-world complex systems focus on the subject of small-world and scale-free networks. While there are several statistical properties of graphs that may be used to characterize their topology (e.g., average path length, clustering coefficient), the work done on HeRoN concentrates on the degree distributions of the obtained networks. Most frameworks, with very few exceptions (Newman et al., 2001), for the study of graph statistical properties have been developed for unipartite, undirected graphs. It is, however, an important aspect for us to consider directed and heterogeneous graphs (graphs with nodes of different types), since this is the case of the network graphs obtained with the HeRoN model. One consequence of the graph being directed is that nodes have two different kinds of edges, the ones arriving at the node and the ones leaving the node - these will be referred to, respectively, as input and output connections. This is particularly important in analyzing the degree distribution of the nodes and therefore nodes of different kinds will be analyzed separately in relation to input and output connectivity.

Figure 6 and Figure 7 show the input and output degree distributions for each kind of node for a 20,000 base long genome on the left column, and for a 500,000 base long genome on the right column. Each column refers to the same network, obtained with a binding threshold of 29, 'avg' binding choice, miRNA binding size of 6 and an inhibition rate of 0. Table 3 gives the number of products of each kind in networks of different sizes with this parameterization.

#genome	#genes	#mRNA/prot	#ncRNA	#miRNA
	(8%)	(6%)	(80%)	(1%)
20000	57	46	556	9
100000	253	194	2993	50
500000	1361	1043	14531	259

Table 3: Number of each kind of product for different genome sizes, binding threshold = 29 and binding choice = avg. The number of proteins is the same as the number of mRNAs.

Whilst the 'genome size' parameter does not seem to

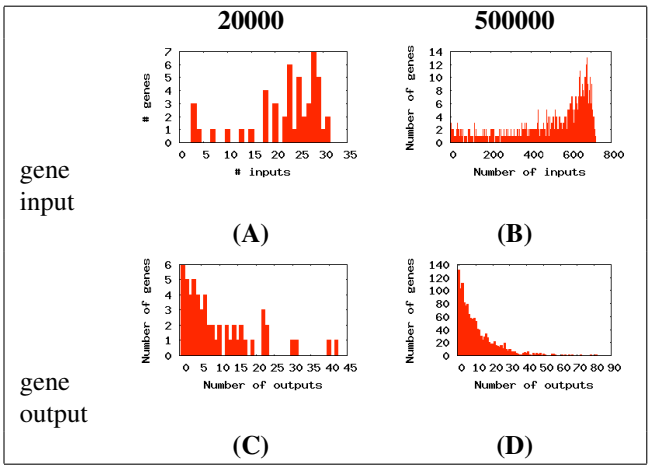


Figure 6: Histograms giving the degree distribution of gene input and output connectivities for a 20,000 base long genome, on the left column, and a 500,000 base long genome, on the right column. **(A)** and **(B)** Gene input connectivity distribution. **(C)** and **(D)** Gene output connectivity distribution.

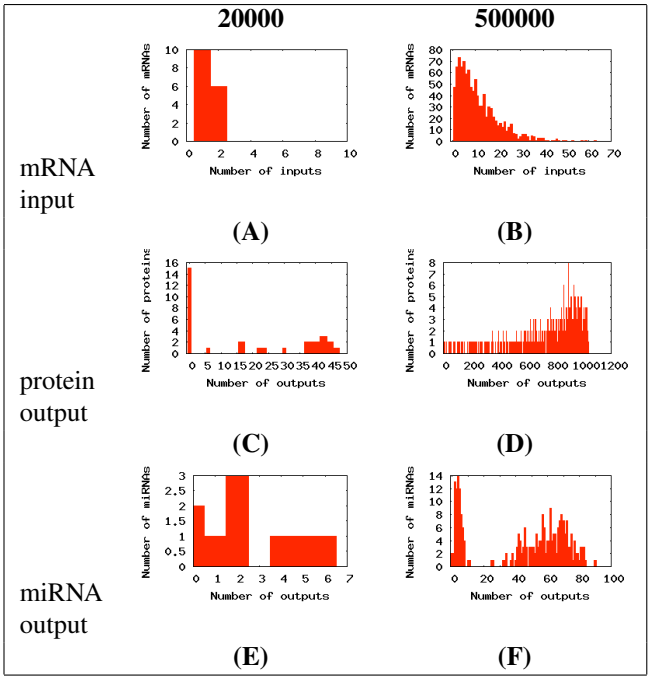


Figure 7: Histograms giving the degree distributions of the different species for a 20,000 base long genome, on the left column, and a 500,000 base long genome, on the right column. **(A)** and **(B)** mRNA input connectivity distribution. **(C)** and **(D)** Protein output connectivity distribution. **(E)** and **(F)** miRNA output connectivity distribution.

qualitatively alter the connectivity distributions, as can be seen by comparing the columns within Figure 6 and Figure 7, the two parameters 'binding threshold' and 'binding choice' determine the input connectivity distribution of genes and the output connectivity distribution of proteins. With a binding threshold of 29 and a 'max' or 'avg' binding choice, a heavily left skewed distribution with a fat tail is found for both the gene input and protein output connectivity (Figure 6B and 7D). These types of distributions are consistent with studies on other complex systems with directed graphs (Newman et al., 2001). The output distribution for the miRNA (Figure 7F) has two peaks. The left most one is the output distribution for the miRNA sequences that contain the subsequence 30 (U1 left), while the rightmost peak is the output distribution for the rest of the miRNA sequences. The reason for this is that the miRNA binds with mature mRNA sequences that have a low probability of containing the U1 left sequence because it is usually sliced with the introns. mRNA molecules have the U1 left sequence when this sequence is not followed by an U1 right sequence in the RNA transcript and therefore is not removed.

Regarding the gene output connectivity the shape of the distribution (Figure 6D) is always maintained, because the parameters that could alter it (promoter, promoter match, termination sequence, left and right u1 and u1 match) were kept fixed throughout the experiments. Figure 8 shows the linear-log and the log-log plots for the gene output. On the linear-log plot the distribution falls on a straight line, indicating an exponential decay of the distribution of connectivity. On the log-log plot the distribution decays faster than a power law would, since if the distribution had a power law tail it would fall on a straight line on this plot.

Some authors have shown evidence for the occurrence of three classes of small-world networks in real world networks: scale-free networks, characterized by a vertex connectivity distribution that decays as a power law; broad-scale networks, characterized by a connectivity distribution that has a power law regime followed by a sharp cutoff like an exponential or Gaussian decay of the tail; and single-scale networks, characterized by a connectivity distribution with a fast decaying tail, such as exponential or Gaussian. The question of why this range of possible structures for small-world networks exists is explained by the preferential attachment of new nodes that gives rise to the power law distributions. In the broad-scale and single-scale networks there are constraints limiting the addition of new links (Amaral et al., 2000). One constraint exists for the connection of new nodes to genes that could account for the faster decay of the tail of the gene output distribution. Genes have outputs to two different types of nodes: mRNA nodes and ncRNA nodes. While a gene only produces one mRNA, it can produce several ncRNAs and, as such, those connections are the most significant in terms of the overall degree distribution. Bigger genes have higher probability of producing several ncRNAs but their ability to produce them decays each time a ncRNA is produced because it shortens the sequence being

searched (search for ncRNAs continues after the last found ncRNA). As with the output of the genes, mRNAs receive two kinds of inputs: each mRNA receives one single input from a gene and possibly several inputs from miRNAs, so the shape of the distribution (Figure 7B) depends, mainly, on the miRNAs. Parameters that influence the input distribution connectivity of the mRNAs are the genome size and the miRNA binding site size. Figure 9 shows the linear-log and the log-log plots of the mRNA input frequency. Similar to the output connectivity of genes, the linear-log plot falls on a straight line, with an exponential decay and the log-log decays faster than a power law would, therefore indicating that there may be constraints limiting the addition of new links between the miRNAs and the mRNAs. Since bigger mRNAs have higher probability of having more inputs, the shape of the input distribution may be greatly influenced by the size distribution of the mRNAs. The scale-free nature of these networks is thus arguable.

Conclusions

The proposed model, HeRoN, introduces a new level of biological detail. The separation of the several processes and the representation of all the products involved in heterogeneous networks allowed, in particular, to extend the model to incorporate a RNA interference mechanism. From the networks obtained some interesting observations about their topology and dynamics can be made. From the static point of view, although many authors claim that the genetic regulatory networks have scale-free topologies, most of them (Geard, 2004; Hallinan and Wiles, 2004; Watson et al., 2004; Willadsen and Wiles, 2003) are not based on experimental results for this concrete type of network but rather on other biological networks, such as the protein-protein interaction networks and metabolic networks. Others (Liebovitch et al., 2006) use experimental mRNA concentration data to extract the networks thus ignoring all regulation other than regulation of transcription initiation. This could lead to misleading results since the presence of an mRNA does not mean that the protein it produces, which is a potential transcription factor, is actually synthesized, as other regulation mechanisms, such as the miRNA negative regulation, may be acting on the mRNA. A model that does not account for these mechanisms may incorrectly assume regulations between genes that are actually regulated by other products. Another question of

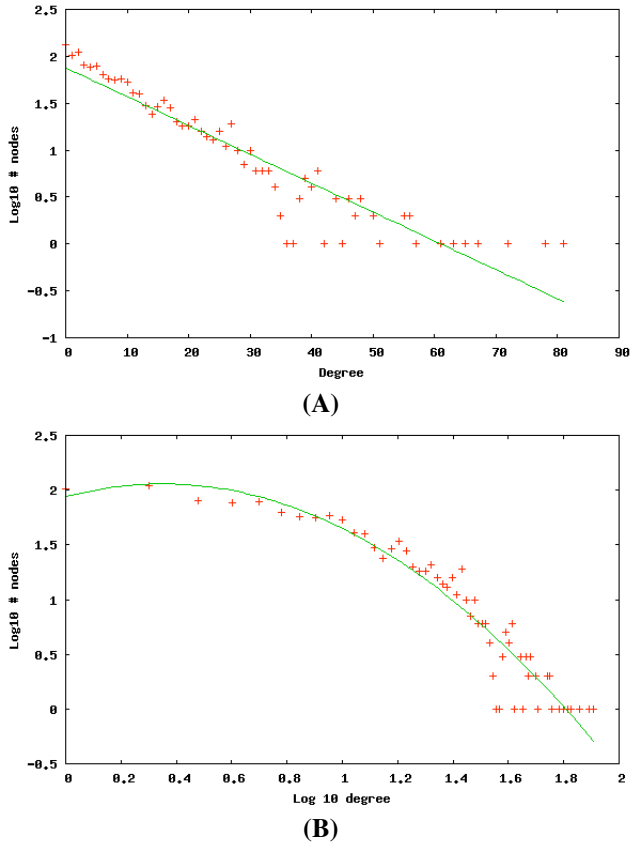


Figure 8: **(A)** Linear-log plot of the gene output connectivity. **(B)** Log-log plot of the gene output connectivity.

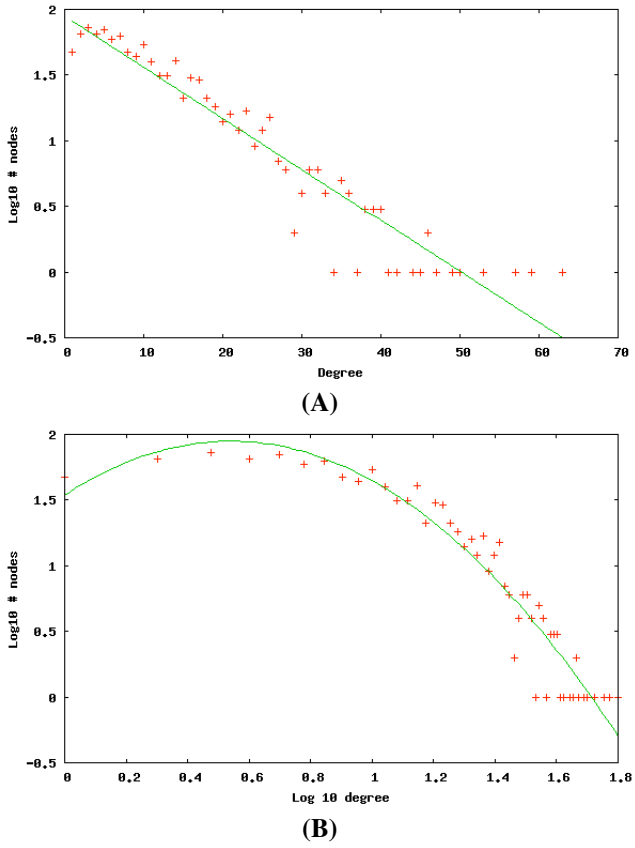


Figure 9: **A** Linear-log plot of the mRNA input connectivity. **B** Log-log plot of the mRNA input connectivity.

importance is that most models make a one-mode projection of an intrinsically heterogeneous network, i.e., they assume a network where all nodes are genes and the edges between them represent regulation relations. When such one-mode projection is made some information is obviously discarded (Newman et al., 2001). As was observed in real-world statistical data of other problems, real complex systems do not always have power law distributions because they are subject to constraints. In the HeRoN model we could not only find degree distributions that are constrained but we also introduced the study of degree distributions for some intermediate products.

This work, and the corresponding model, can be extended in several directions. An important issue that must be addressed is its scalability. Experiments with genomes of realistic dimensions should be performed. The genome of the E.Coli would be a good starting point since it is one of the smallest (4,600,000 bases long) and the most studied genome available. Then, the model could be improved and made more biological sound, by taking into account aspects such as the concentration of products (a continuous variable) and the time delays involved.

Finally, it would be interesting to observe how the alternative splicing of genes could alter the output degree distribution of genes, proteins and miRNAs. The HeRoN model would have to be extended to include this feature. Although several interesting observations were made by analyzing the degree distribution of the nodes, there are several other statistical properties that could be used to better understand them. Future work should include a study on the clustering coefficient, the average path length between nodes, the distribution of component (subgraph) sizes and the existence and size of a giant-component.

References

Amaral, L., Scala, A., Barthélémy, M., and Stanley, H. (2000). Classes of small-world networks. In *Proceedings of the National Academy of Sciences (USA)*, volume 97, pages 11149–52.

Ando, S. and Iba, H. (2001). Inference of gene regulatory model by genetic algorithms. In Kim, J.-H., editor, *Proceedings of Congress on Evolutionary Computation*, volume 1, pages 712–719.

Baker, D. and Sali, A. (2001). Protein structure prediction and structural genomics. *Science*, 294(5540):93–96.

Banzhaf, W. (2003). *Artificial Regulatory Networks and Genetic Programming*, pages 43–62. Springer Verlag.

D'Haeseleer, P. (2000). *Reconstructing Gene Networks fro Large Scale Gene Expressin Data*. PhD thesis, University of New Mexico (USA).

Geard, N. (2004). Modeling gene regulatory networks: fromn systems biology to complex systems. Accs draft technical report, University of Queensland (Australia).

Gonçalves, Â. and Costa, E. (2007a). A computational model for gene regulatory networks. Technical Report TR 2007/06, CISUC - University of Coimbra.

Gonçalves, Â. and Costa, E. (2007b). Heron: a computational model of gene regulatory networks. In Neves, J., Santos, M., and Machado, J., editors, *New Trends in Artificial Intelligence*, pages 288–299.

Gregory, R., Chendrimada, T., and Shiekhattar, R. (2006). Microrna biogenesis: isolation and characterization of the microprocessor complex. *Methods in Molecular Biology*, 342:33 – 47.

Hallinan, J. and Wiles, J. (2004). Evolving genetic regulatory networks usinf an artificial genome. In *Proceedings of the Seconf Conference on Asia-Pacific Bioinformatics*, volume 29, pages 291–296. Australian Computer Society.

Hoffman, M., Khrapov, M., Cox, J., Yao, J., Tong, L., and Ellington, A. (2004). Aant: the aminoacid-nucleotide interaction database. *Nucleic Acid Research*, 32:174–181.

Kauffman, S. (1993). *The Origins of Order: self-organization and selection in evolution*. Oxford University Press.

Liebovitch, L., Jirsa, V., and Shehadeh, L. (2006). *Complexus Mundi: emergent patterns in nature*, chapter Structure of genetic regulatory networks: evidence for scale-free networks. World Scientific.

Martens, J., Laprade, L., and Winston, F. (2004). Intergenic transcription is required to repress the saccharomyces cerevisiae ser3 gene. *Nature*, 429:571–574.

Newman, M., Strogatz, S., and Watts, D. (2001). Random graphs with arbitrary degree distributions and their applications. *Physical Review E*, 64.

Noman, N. and Iba, H. (2005). Inference in gene regulatory networks using s-system and differential evolution. In Beyer, H., editor, *Proceedings of the 2005 Conference on Genetic and Evolutinoary Computation*, pages 439–446. ACM Press.

Reil, T. (1999). Dynamics of gene expression in an artificial genome: implications for biolocal and artificial ontogeny. In Floreano, D., Nicoud, J.-D., and Mondada, F., editors, *Advances in Artificial Life: 5th European Conference on Artificial Life*, pages 457–466. Springer Verlag.

Sakamoto, E. and Iba, H. (2001). Inferring a system of differential equations for a gene regulatory network by using genetic programming. In Kim, J.-H., editor, *Proceedings of the 2001 congress on Evolutionary Computation*, volume 1, pages 720–726.

Watson, J., Geard, N., and Wiles, J. (2004). Towards more biological mutation operators in gene regulation studies. *BioSystems*, 76:239–248.

Willadsen, K. and Wiles, J. (2003). Dynamics of gene expression in an artificial genome. In Sarker, R., editor, *The 2003 Congress on Evolutionary Computation*, volume 1, pages 185–190.

Zhang, D. and Rosbash, M. (1999). Identification of height proteins that cross-link to pre-mrna in the yeast commitmem complex. *Genes Dev*, 13:581–592.

What Makes the Spatial Prisoner's Dilemma Game Sensitive to Asynchronism?

Carlos Grilo[1,2], Luís Correia[2]

[1]Dep. Eng. Informática, Escola Superior de Tecnologia e Gestão
Instituto Politécnico de Leiria
[2]LabMAg, Dep. Informática, Faculdade Ciências da Universidade de Lisboa
grilo@estg.ipleiria.pt

Abstract

We investigate aspects that control the Spatial Prisoner's Dilemma game sensitivity to the synchrony rate of the model. Based on simulations done with the *generalized proportional* and the *replicator dynamics* transition rules, we conclude that the sensitivity of the game to the synchrony rate depends almost exclusively on the transition rule used to model the strategy update by the agents. We then identify the features of these transition rules that are responsible for the sensitivity of the game. The results show that the Spatial Prisoner's Dilemma game becomes more and more sensitive for noise levels above a given noise threshold. Below this threshold, the game is robust to the noise level and its robustness even slightly grows, compared to the imitate the best strategy, if a small amount of noise is present in the strategy update process.

Introduction

Spatial evolutionary games are used as models to study, for example, how cooperation could ever emerge in nature and human societies (Smith, 1982). They are also used as models to study how cooperation can be promoted and sustained in artificial societies (Oh, 2001). In these models, a structured population of agents interacts during several time steps through a given *game* which is used as a metaphor for the type of interaction that is being studied. The population is structured in the sense that each agent can only interact with its neighbors. The underlying structure that defines who interacts with whom is called the *interaction topology*. After each interaction session, some or all the agents, depending on the *update dynamics* used, have the possibility of changing their strategies. This is done using a so called *transition rule* that models the fact that agents tend to adapt their behavior to the context in which they live by imitating the most successful agents they know. It can also be interpreted as the selection step of an evolutionary process in which the least successful strategies tend to be replaced by the most successful ones.

The discussion about using synchronous or asynchronous dynamics on these models started with a paper by Huberman and Glance (1993). Synchronous dynamics means that, at each time step, the revision of strategies happens for all agents simultaneously, while this is not the case for asynchronous dynamics. In that paper the authors contested the results achieved by Nowak and May (1992) who showed that cooperation can be maintained when the Prisoner's Dilemma game is played on a regular 2-dimensional grid by agents which do not remember their neighbors' past actions. Huberman and Glance criticized the fact that the model used in (Nowak and May, 1992) was a synchronous one, which is an artificial feature. They also presented the results of simulations where cooperation was no longer sustainable when an asynchronous dynamics were used. After this work, Nowak et al. (1994) tested their model under several conditions, including synchronous and asynchronous dynamics and showed that cooperation can be maintained for many different conditions, including asynchronism. However, the results are presented through system snapshot images, which render it difficult to measure the way they are affected by the modification from synchronous to asynchronous dynamics. Recently, in (Newth and Cornforth, 2007), a similar scenario was studied using various asynchronous update methods besides synchronous dynamics. The authors found that the synchronous updating scheme supports more cooperators than the asynchronous ones.

On the contrary, in (Grilo and Correia, 2007) we found that, in the Spatial Prisoner's Dilemma game, asynchronous updating supports, in general, more cooperators than synchronous updating. This conclusion was only possible because a large number of conditions was tested. Namely, we used small-world networks as interaction topologies so that the whole spectrum between regular and random networks could be explored. We also used the *generalized proportional* transition rule (see Section III), which allows us to tune the level of noise present in the strategy update process. We consider that there is noise when an agent fails to imitate the strategy of its most successful neighbor. We found that asynchronous updating is detrimental for cooperation only for very small noise values. That is, for the majority of the noise domain, asynchronous updating benefits cooperation. Also, as we go from regular to random networks, asyn-

chronous updating becomes beneficial to cooperation even for very small noise values. In (Grilo and Correia, 2008) we showed that the conclusions do not change if scale-free networks (Barabasi and Albert, 1999) are used. We also showed that the final outcome of the model is basically the same whether a deterministic or a stochastic asynchronous dynamics is used, which is in contrast with results reported in (Gershenson, 2002) for random boolean networks.

The proportion of cooperating agents eventually achieved in a spatial evolutionary game can be influenced by, for example, the game that is being used, the interaction topology, the transition rule or the update dynamics. The influence of some of these aspects has previously been studied. For example, in (Pacheco and Santos, 2005) the influence of the interaction topology is examined. Also, in (Tomassini et al., 2006) the influence of the interaction topology, the transition rule and the update dynamics in the Hawk-Dove game are studied.

But, as far as we know, prior to this work, there has been no explanation of the influence of the update dynamics in the outcome of spatial evolutionary games. This work is a step in that direction. Here, we identify the aspects that control the Spatial Prisoner's Dilemma game sensitivity to asynchronism. Based on previous simulations performed with the *generalized proportional* transition rule and new ones done with the *replicator dynamics* transition rule, we first conclude that the sensitivity of the Spatial Prisoner's Dilemma game to asynchronism depends almost exclusively on the transition rule. We then identify the features of these transition rules that are responsible for the sensitivity of the game.

The paper is structured as follows: in Section II we describe the model used in our simulations. In Section III we first compare the results achieved with the *generalized proportional* and the *replicator dynamics* transition rules and then we identify the features of these rules that influence the sensitivity of the model to asynchronism. Finally, in Section IV some conclusions are drawn and future work is advanced.

The Model

The Prisoner's Dilemma Game

In the Prisoner's dilemma game (PD), players can cooperate (C) or defect (D). The payoffs are the following: R to each player if they both play C; P to each if they both play D; T and S if one plays D and the other C, respectively. These values must obey $T > R > P > S$ and $2R > T+S$. It follows that there is a strong temptation to play D. But, if both play D, which is the rational choice or the Nash equilibrium of the game, both get less payoff than if they both play C, hence the dilemma. For practical reasons, the payoffs are usually defined as $R = 1, T = b > 1$ and $S = P = 0$, where b represents the advantage of D players over C ones when they play the game with each other. This has the advantage that the game can be described by only one parameter without losing its essence (Nowak et al., 1994).

Interaction Topology

We use *small-world networks* (SWNs) (Watts and Strogatz, 1998) as the interaction topology. We build SWNs as in (Tomassini et al., 2006): first, a toroidal regular 2-dimensional grid is built so that each node is linked to its 8 surrounding neighbors by undirected links; then, with probability ϕ, each link is replaced by another one linking two randomly selected nodes. Parameter ϕ is called the *rewiring probability*. Some works (Nowak et al., 1994) allow self-links because it is considered that each node can represent not a single agent but a set of similar agents that may interact with each other. Here, we do not allow self-interaction since we are interested in modeling nodes as individual agents. Repeated links and disconnected graphs are also avoided. The rewiring process may create long range links connecting distant agents. For simplicity, we will refer to interconnected agents as neighbors, even if they are not located at adjacent nodes. By varying ϕ from 0 to 1 we are able to build from completely regular networks to random ones. SWNs have the property that, even for very small values of the rewiring probability, the average path length between any two nodes is much smaller than in a regular network, maintaining however a high clustering coefficient observed in many real systems including social ones.

Interaction and Strategy Update Dynamics

On each time step, agents first play a one round PD game with all their neighbors. Agents are pure strategists which can only play C or D. After this interaction stage, each agent updates its strategy with probability α using a *transition rule* (see next section) that takes into account the payoff of the agent's neighbors. The update is done synchronously by all the agents selected to engage in this revision process. The α parameter is called the *synchrony rate* and is the same for all agents. This type of update dynamics is called *asynchronous stochastic dynamics* (Fatès and Morvan, 2005). It allows us to cover all the spectrum between synchronous and sequential dynamics. When $\alpha = 1$ we have a synchronous model, where all the agents update at the same time. As $\alpha \to \frac{1}{n}$, where n is the population size, the model approaches sequential dynamics, where exactly one agent updates its strategy at each time step.

Asynchronous stochastic dynamics models the fact that, at each moment, more than one agent, but not necessarily all of them, can update their strategy. Usually, asynchronism is understood as sequential dynamics. As an example, in all the works mentioned above, asynchronous dynamics means sequential updating. However, the reality seems to lie somewhere between synchronism and sequentiality and, so, both types of dynamics can be considered as artificial. In a population of interacting agents, many decision processes can

occur at the same time but not necessarily involving all the agents. If these were instantaneous phenomena we could model the dynamics of the system as if they occurred one after another but that is not usually the case. These processes can take some time, which means that their output is not available to other ongoing decision processes. Even if we consider them as being instantaneous, the time that information takes to be transmitted and perceived implies that their consequences are not immediately available to other agents. Asynchronous stochastic dynamics also models the fact that, at each time step, the number of agents updating their strategy is not always the same, which is a reasonable assumption. With this type of dynamics, this number follows a binomial distribution with mean α. Apart from these considerations, as we will see in the following sections, the fact that the α parameter allows us to explore intermediate levels of asynchronism is also useful in the analysis of the influence of this feature.

Simulations Setup

All the simulations were performed with populations of $50 \times 50 = 2500$ agents, randomly initialized with 50% of Cs and 50% of Ds. When the system is running synchronously, i.e., when $\alpha = 1$, we let it first run during a period of 900 iterations which, we confirmed, is enough to pass the transient period of the evolutionary process. After this, we let the system run for 100 more iterations and, at the end, we take as output the average proportion of cooperators during this period, which is called the *sampling period*. When $\alpha \neq 1$ the number of selected agents at each time step may not be equal to the size of the population and it may vary between two consecutive time steps. In order to guarantee that these runs are equivalent to the synchronous ones in what concerns to the total number of individual updates, we let the system first run until 900×2500 individual updates have been done. After this, we sample the proportion of cooperators during more 100×2500 individual updates and we average it by the number of time steps needed to do these updates. For each combination, 30 runs were made and the average of these runs is taken as the output.

Simulation Results

In our first simulations (Grilo and Correia, 2007, 2008), we used, a generalization of the *proportional* transition rule (GP) proposed in (Nowak et al., 1994). Let G_x be the average payoff earned by agent x, N_x be the set of neighbors of x and c_x be equal to 1 if x's strategy is C and 0 otherwise. According to this rule, the probability that an agent x adopts C as its next strategy is

$$p_C(x, K) = \frac{\sum_{i \in N_x \cup x} c_i (G_i)^{\frac{1}{K}}}{\sum_{i \in N_x \cup x} (G_i)^{\frac{1}{K}}}, \quad (1)$$

Parameter	Values
ϕ	0 (reg.), 1 (rand.), SW: 0.01, 0.05, 0.1
α	0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1
b	1, 1.1, 1.2, 1.3, 1.4, 1.5, 1.6, 1.7, 1.8, 1.9, 2
K	0, 1/100, 1/10, 1/8, 1/6, 1/4, 1/2, 1

Table 1: Parameter values used in the simulations.

where $K \in]0, +\infty[$ can be viewed as the noise present in the strategy update process. Noise is present in this process if there is some possibility that an agent imitates strategies other than the one used by its most successful neighbor. Small noise values favor the choice of the most successful neighbors' strategies. Also, as noise diminishes, the probability of imitating an agent with a lower payoff becomes smaller. When $K \to 0$ we have a deterministic best-neighbor rule such that i always adopts the best neighbor's strategy. When $K = 1$ we have a simple proportional update rule. Finally, for $K \to +\infty$ we have random drift where payoffs play no role in the decision process. For the moment, our analysis considers only the interval $K \in]0, 1]$. In this interval the decision process is strongly guided by the payoffs earned by the agents.

Each simulation is a combination of the ϕ, α, b and K parameters, and all the possible combinations of the values shown in Table 1 were tested. As Fig. 1 illustrates, when the GP rule is used, in situations where both cooperation and defection coexist, the level of cooperation can change significantly as we change α. For given α and b values, the levels of cooperation may be different when distinct ϕ and K values are used. Also, the exact way how the model reacts to α changes may change as well. However, no matter the ϕ and K values used, there is a common qualitative behavior: the model is sensitive to changes in the synchrony rate α. Due to this and space limitations we only show results for $\phi = 0.1$, inside the small world regime.

After experimenting with the GP rule, we also ran simulations with one of the most popular transition rules, the *replicator dynamics* rule (RD) (Hofbauer and Sigmund, 1998), which, when used on structured populations, is defined in the following way (Tomassini et al., 2006): the probability $p(s_x \to s_y)$ that an agent x, with strategy s_x and average payoff G_x, imitates a randomly chosen neighbor y, with strategy s_y and average payoff G_y, is equal to:

$$p(s_x \to s_y) = f(G_y - G_x) = \begin{cases} \frac{G_y - G_x}{b} & \text{if } G_y - G_x > 0 \\ 0 & \text{otherwise,} \end{cases} \quad (2)$$

where b is the largest possible payoff difference between two players in a one shot PD game. As Fig. 2 illustrates,

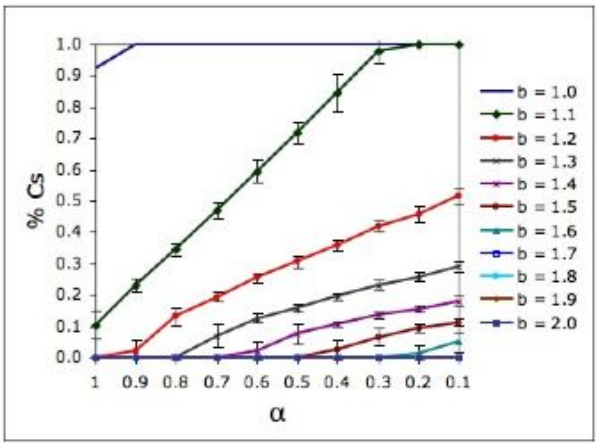


Figure 1: % of cooperators for $\phi = 0.1$ and $K = 1$ (GP rule).

when the RD rule is used, the level of cooperation is approximately constant as we change the synchrony rate α. As for the GP rule, the qualitative behavior of the model does not change no matter the interaction topology used.

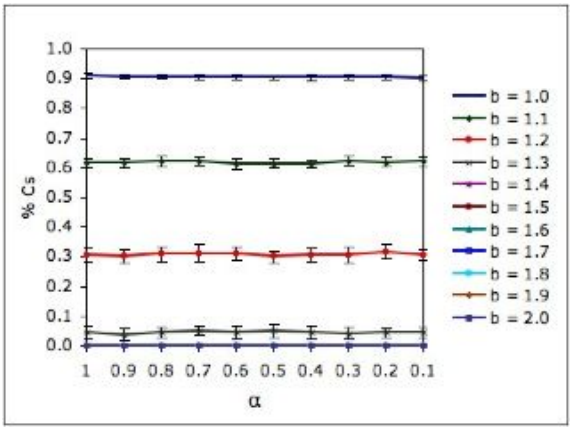


Figure 2: % of cooperators for $\phi = 0.1$ (RD rule).

From these results, it follows that the sensitivity of the model to the synchrony rate depends almost entirely on the transition rule that is used. This brings us to the question we try to answer with this work: which features of these transition rules are responsible for the Spatial PD's game sensitivity to the synchrony rate? After describing the function we use to measure the sensitivity of the model to the synchrony rate, we will start by looking to one of these features: *payoff monotonicity*.

Sensitivity Measure

We want to measure the sensitivity to the synchrony rate for situations like, for example, the one of Fig. 1, where ϕ and K are fixed. Let $C(\phi, R, b_i, \alpha_j)$ be the proportion of cooperators achieved for specific input parameters, where R represents the input parameter set of the transition rule (for example, for the GP rule $R = \{K\}$). We first compute, for each b value, the standard deviation of the proportion of cooperators achieved along all α values. We then sum these standard deviations, which gives us the overall sensitivity for a specific combination of ϕ and R values:

$$s(\phi, R) = \sum_{i=0}^{10} \sqrt{\frac{1}{10} \sum_{j=1}^{10} (C(\phi, R, b_i, \alpha_j) - \overline{C}(\phi, R, b_i))^2}, \quad (3)$$

where $b_i = 1 + 0.1i$ and $\alpha_j = 0.1j$. This measure compresses the results obtained for given ϕ and R parameters in a single value, which may lead to some loss of information. Therefore, whenever necessary, we will complement the results obtained with equation 3 with an analysis of the data from which the sensitivity values were derived.

Payoff Monotonicity

A transition rule is said to be *payoff monotonic* if it forbids the imitation of agents with smaller payoffs (Szabó, 2007). Looking at equations (1) and (2) we easily see that, while the RD rule is payoff monotonic, the GP rule is not (except when $K \to 0$). Given this, we first modified the RD rule in order to turn it into a non-payoff monotonic rule. The modified rule is as follows:

$$p(s_x \to s_y) = f(G_y - G_x, M) =$$

$$\begin{cases} (1 - \frac{1}{M})\frac{G_y - G_x}{b} + \frac{1}{M} & \text{if } G_y - G_x > 0 \\ \frac{1}{M} - \frac{1}{M}\frac{G_x - G_y}{b} & \text{otherwise,} \end{cases} \quad (4)$$

where $\frac{1}{M}$ is the probability that x imitates y when $G_x = G_y$. $M \in [1, +\infty[$ can be viewed as the payoff monotonicity degree: the bigger M, the smaller the probability that x imitates an agent with a lower payoff. We refer to this rule as *non-payoff monotonic* RD (NPMRD).

Fig. 3 shows the sensitivity of the model calculated as in equation 3. It shows that the sensitivity grows up to $\frac{1}{M} = 0.3$ and decreases after this value, although staying higher than the sensitivity of the standard RD rule. This means that the RD rule becomes sensitive to the synchrony rate only if it is non-payoff monotonic. But, if we look at Fig. 4, where the proportion of cooperators is depicted for $b = 1$, we can see that, for situations where cooperators and defectors coexist, the sensitivity continues to grow even for $\frac{1}{M} > 0.3$. That is, in these situations the influence of the synchrony rate in the output of the system grows as $\frac{1}{M}$ grows.

After this, we modified the GP rule in order to verify if its sensitivity to the synchrony rate is also due to the fact

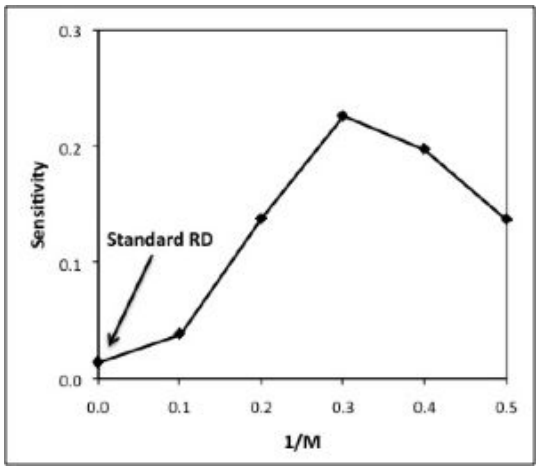


Figure 3: Sensitivity of the NPMRD rule to the synchrony rate as a function of $\frac{1}{M}$, for $\phi = 0.1$.

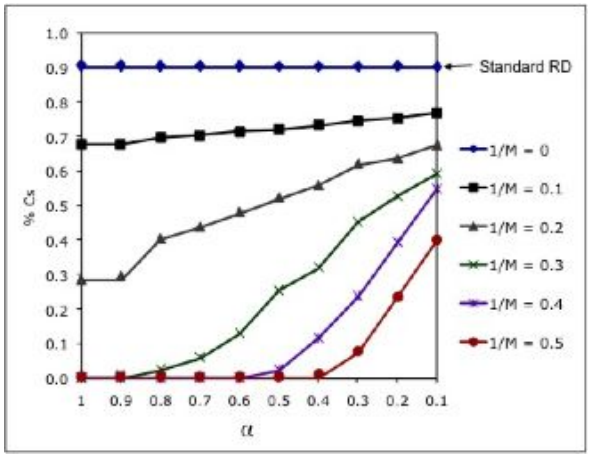


Figure 4: % of cooperators for $\phi = 0.1$ and $b = 1$ (NPMRD).

that agents can imitate a neighbor with a lower payoff. We will refer this rule as *payoff monotonic* GP (PMGP). Before describing PMGP, we recall that the GP rule takes the sum of the payoffs of C/D agents instead of treating the strategy/payoff of each neighbor individually. Putting it another way, the GP rule models a competition between two strategies (C and D) so that the winning probability is proportional to the sum of the payoffs of the agents using each strategy. The PMGP rule applies the original GP rule, eq. (1), only if one of the two following conditions is true:

$$\text{if } G_{Cs} < G_{Ds} \text{ and } s_x = C, \tag{5}$$

$$\text{if } G_{Cs} > G_{Ds} \text{ and } s_x = D, \tag{6}$$

where G_{Cs} and G_{Ds} are, respectively, the sums of the payoffs of C and D neighbors (including the payoff of the agent to be updated x), each one powered to $\frac{1}{K}$. Agent x keeps its strategy if none of these conditions is true.

Fig. 5 shows that the PMGP rule becomes much less sensitive to α changes than the original version for $K > 0.1$. Just as an example, compare Fig. 1 with Fig. 6: even taking into account some significant standard deviations in the payoff monotonic case, the difference in sensitivity between the two situations is clear. The divergence for $K > 0.1$ means that, above this value, payoff monotonicity also plays an important role in the insensitivity of the GP rule to changes in the synchrony rate, as it does with the standard RD rule. Stating it from the opposite perspective, when agents are allowed to imitate less successful strategies, the model's sensitivity grows as this possibility increases. Given that the probability of choosing less successful strategies grows with the noise level, this means that high noise levels increase the model sensitivity to the synchrony rate.

But, Fig. 5 also shows that, for $K <= 0.1$, the sensitivity of the PMGP rule stops diverging from the sensitivity of the original GP rule. It also shows that payoff monotonicity is not the only force that influences the sensitivity of the model to the synchrony rate. Notice that the sensitivity of the PMGP rule also varies as we change the noise level. That is, even when we prevent the imitation of less successful strategies, the model's sensitivity continues to vary with noise: it grows as the noise level decreases. Therefore, there must be another feature related to the noise level that also influences the model's sensitivity, although less than payoff monotonicity. We address this problem in the next section.

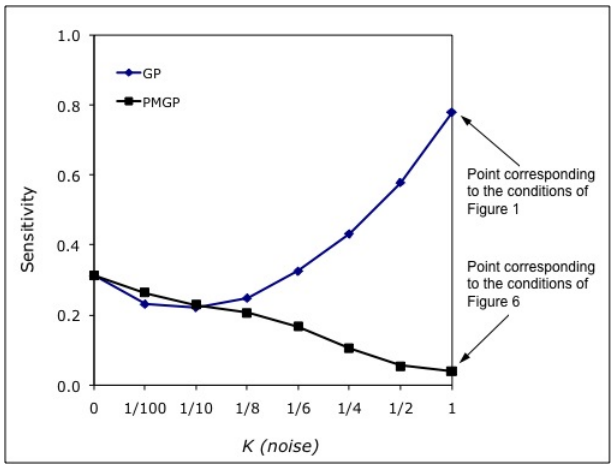


Figure 5: Sensitivity of the GP and PMGP rules to the synchrony rate as a function of K, for $\phi = 0.1$.

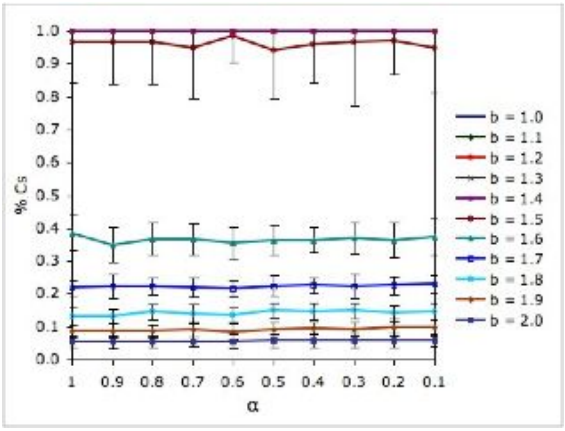


Figure 6: % of cooperators for $\phi = 0.1$ and K = 1 (PMGP rule).

Imitate the Best Tendency

Given that, with the PMGP rule, agents cannot imitate less successful strategies, what other forces influence the sensitivity of the Spatial PD game? If we analyze the original GP rule, we see that the probability of choosing a strategy with a lower payoff becomes very low as K approaches 0. That is, the payoff monotonicity degree increases as K decreases. On the other hand, as K decreases, the tendency to imitate the wealthiest neighbors is increased for both the original and the modified GP rule. Therefore, the two rules become more and more similar as K is decreased. In fact, when $K \to 0$, the two rules become one and the same deterministic rule: choose the strategy used by the best neighbor (see Fig. 5). This explains why the the two rules' sensitivities are similar for $K < 0.1$.

The above reasoning suggests that, besides payoff monotonicity, the "imitate the best tendency" level also influences the sensitivity of the Spatial PD game to the synchrony rate. More specifically, it suggests that the sensitivity of the model increases with the "imitate the best tendency" level. This could explain why the sensitivity of the model slightly increases for K values near 0 when the original GP rule is used (see Fig. 5). In order to verify this hypothesis, and given that it is based only on results achieved with the GP rule, we now turn our attention again to the RD rule. The goal is to verify if the "imitate the best tendency" level also influences the sensitivity of the model when this rule is used.

The first modification we have done to the RD rule was to change the way the neighbor y is chosen: each neighbor of the updating agent x has a given probability $0 < \theta \leq 1$ of entering a tournament. After this, the wealthiest agent in the tournament is selected and becomes the candidate neighbor y. θ represents the tendency of x to select its best neighbors. For example, when $\theta = 1$, y is always the wealthiest neighbor of x.

Once defined the way of choosing y, we still have no total control on x's "imitate the best tendency". Notice that, in the standard RD rule, $p(s_x \to s_y)$ only depends on the difference $G_y - G_x$. That is, we have no control on the sensitivity of x to the payoff difference between the two agents. Given this, we further modified $p(s_x \to s_y)$ in the following way:

$$p(s_x \to s_y) = f(G_y - G_x, S) = \begin{cases} (\frac{G_y - G_x}{b})^{\frac{1}{S}} & \text{if } G_y - G_x > 0 \\ 0 & \text{otherwise,} \end{cases} \quad (7)$$

where the sensitivity of x to $G_y - G_x$ is given by $S \in [1, +\infty[$: for the same payoff difference, the larger S, the bigger the probability that x imitates y. With these two modifications we can cover all the space between the best neighbor rule ($\theta = 1$, $S = +\infty$) and the standard RD rule ($\theta \approx \frac{1}{|N_x|}$, $S = 1$). We will refer to this rule as *extended* RD (ERD).

Fig. 7 shows the sensitivity of the ERD rule calculated as in equation 3. As can be seen in the chart, excepting some small fluctuations, the sensitivity of the model when the ERD rule is used grows as both θ and S are increased. This means that, as for the GP rule, a strong "imitate the best tendency" level also increases the RD's rule sensitivity to the synchrony rate.

Neighborhood Monitoring

There is yet another feature in which GP and RD differ: while the GP rule models a complete monitoring of the neighborhood (because all the neighbors' payoffs are considered), the RD rule models a partial neighborhood monitoring (only the payoff of one neighbor is considered). Notice that, despite the fact that the above described variant ERD allows a variable neighborhood monitoring, it considers only the payoff of one agent. Thus, we also modified the two rules in order to verify if this feature has some influence on the sensitivity to α.

The GP rule was modified in the following way: each neighbor of x has a given probability β of being considered in equation 1 (the updating agent x is always considered). The β parameter can be viewed as the neighborhood monitoring level. We will refer to this rule as *partial neighborhood monitoring* GP (PNMGP). Fig. 8 shows that the PNMGP rule is less sensitive to the synchrony rate than the original GP rule by a factor of approximately $1/2$, maintaing, however, a similar qualitative behavior.

The RD rule was modified so that, as in the case of the original GP rule, the payoff of all the neighbors contribute to the decision of the updating agent x. According to the *complete neighborhood monitoring* RD rule (CNMRD), the

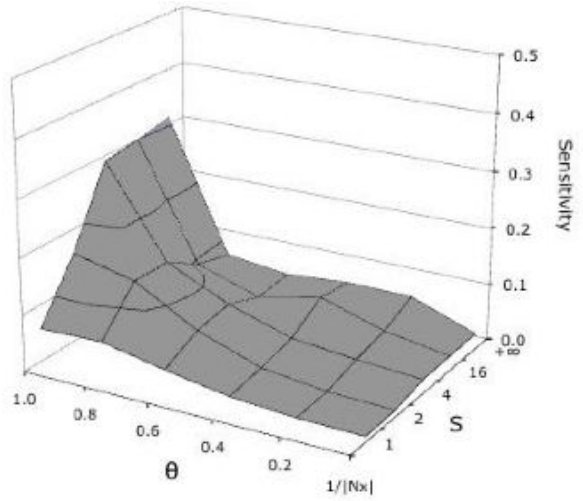


Figure 7: Sensitivity of the ERD rule for $\phi = 0.1$ as a function of θ and S. $\theta = \frac{1}{|N_x|}$ means that only a candidate neighbor y is randomly chosen as in the standard RD rule. Therefore, the point $s(\theta = \frac{1}{|N_x|}, S = 0)$ corresponds to the sensitivity of the standard RD rule (Fig. 2). $s(\theta = 1, S = +\infty) = 0.312$, which is very close to $s(K = 0) = 0.314$ of Fig. 5. Both points correspond to the best neighbor rule.

probability $p(s_x \rightarrow s_a)$ that an agent x, with strategy s_x, changes its strategy to an alternative strategy s_a, where $s_a = D$ if $s_x = C$ and vice-versa, is equal to:

$$p(s_x \rightarrow s_a) = \begin{cases} \frac{G_X - G_A}{bk_A} & \text{if } G_X - G_A > 0 \\ 0 & \text{otherwise,} \end{cases} \quad (8)$$

where G_X and G_A are the sum of the average payoffs earned by the neighbors of x playing, respectively, strategy s_x and s_a (including x), and k_A is the number of neighbors with strategy s_a. Fig. 9 shows the proportion of cooperators achieved with this rule when $\phi = 0.1$. The sensitivity to the synchrony rate for this situation is equal to 0.030, which is about the double of the sensitivity of the standard RD rule, 0.014, for the same situation (Fig. 2). This result is consistent with the one achieved with the PNMGP and GP rules. However, for the two situations, that is, for the GP *versus* PNMGP and the RD *versus* CNMRD rules, the difference in sensitivity is partly due to the fact that, with the complete neighborhood monitoring versions, there are more b values for which Cs and Ds coexist (compare, Fig. 2 and Fig. 9) than for the partial neighborhood monitoring versions. Therefore, more work must be done, namely exploring intermediate levels of neighborhood monitoring, in order to determine the real influence of the neighborhood monitoring level over the model's sensitivity to the synchrony rate.

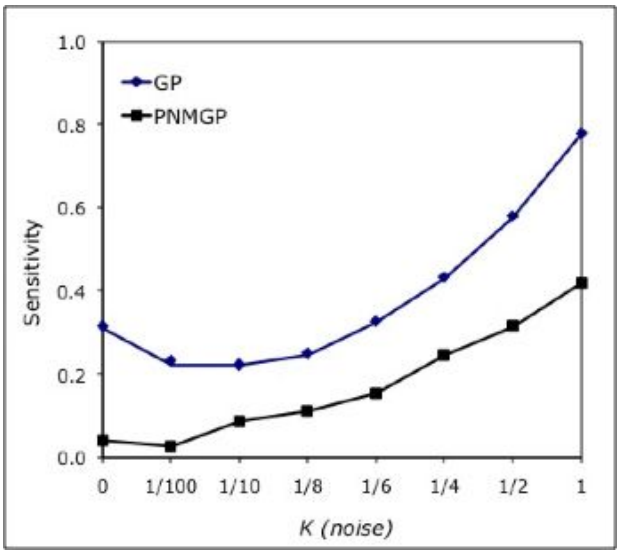


Figure 8: Sensitivity of the PNMGP rule to the synchrony rate as a function of K, for $\phi = 0.1$ and $\beta = 0.1$.

Conclusions and Future Work

In this work we identified the features that determine the sensitivity of the Spatial Prisoner's Dilemma game to the synchrony rate. We first found that the sensitivity of the model depends almost completely on the transition rule used to model the strategy update process. For this, we used the *generalized proportional* and the *replicator dynamics* rules which are, respectively, sensitive and insensitive to the synchrony rate no matter the interaction topologies used in the simulations. We then used some variants of these rules in order to identify the features that make them responsible for the sensitivity of the model.

The results can be summarized in the following way: the lower the payoff monotonicity degree and the higher the "imitate the best tendency" level, the more sensitive is the game to the synchrony rate. But, given that these are just consequences of the noise level, we can state the results in the following way: on the one hand, the Spatial Prisoner's Dilemma game becomes more and more sensitive for noise levels above a given noise threshold (0.1 in the GP transition rule). On the other hand, the game is robust to small noise levels, and its robustness even grows, compared to the imitate the best strategy, if a small amount of noise is present in the strategy update process. The line corresponding to the original GP rule in Fig. 5 illustrates this well. As far as we know, this is the first time such a result is achieved. We stress that these results are the same for all the interaction topologies we used in the simulations, which go from regular to random networks.

This result indicates that the noise level may play an important role in the robustness of real dynamical systems where social dilemmas exist. More precisely, it suggests that

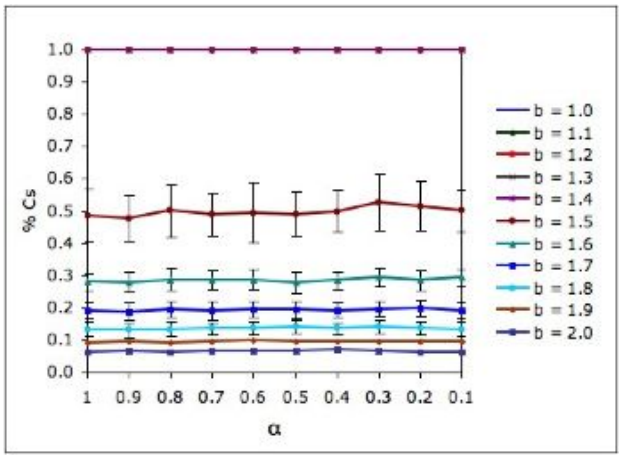


Figure 9: % of cooperators for $\phi = 0.1$ (CNMRD rule).

a moderate noise level can enhance the system's robustness to small variations on the underlying conditions. On the other hand, significant noise levels make a dynamical system too sensitive to small perturbations. More work must be done, however, in order to verify if this can be generalized to perturbations other than the ones related to the synchrony rate.

Future extensions to this work will explore *asynchronous stochastic dynamics* with other games in order to verify if the results achieved with the Prisoner's Dilemma game can be further generalized. The results achieved in (Tomassini et al., 2006) with the Hawk-Dove game, where the *best-neighbor* ($K \rightarrow 0$), the simple *proportional* ($K = 1$) and the *replicator dynamics* transition rules, as well as synchronous and sequential updating were used, seem to indicate that, also in this game, the transition rule is what determines the sensitivity of the model. However, only by exploring intermediate asynchronism and noise levels we can confirm this. Other transition rules, as the Sigmoid transition rule (Szabó, 2007) and interaction topologies, as the scale-free network model, will also be explored.

Finally, even if we now know that the noise level of the transition rule is the key feature in what concerns the sensitivity of the Spatial Prisoner's Dilemma game to the synchrony rate, we still do not know why it influences the sensitivity of the model as it does. Trying to explain this will be one of the main directions of our future work.

Acknowledgements

We thank the reviewers of the paper for their useful comments. We also thank the GruVA members Pedro Santana, Vasco Santos and Luis Simões for useful discussions and comments. This work was partially supported by FCT/MCTES grant No. SFRH/BD/37650/2007.

References

Barabasi, A.-L. and Albert, R. (1999). Emergence of scaling in random networks. *Science*, 286:509.

Fatès, N. and Morvan, M. (2005). An experimental study of robustness to asynchronism for elementary cellular automata. *Complex Systems*, 16(1):1–27.

Gershenson, C. (2002). Classification of random boolean networks. In *Proceedings of the Eight International Conference on Artificial Life*, pages 1–8. MIT Press.

Grilo, C. and Correia, L. (2007). Asynchronous stochastic dynamics and the spatial prisoner's dilemma game. In *Proceedings of the 13th Portuguese Conference on Artificial Intelligence, EPIA 2007*, pages 235–246. Springer-Verlag.

Grilo, C. and Correia, L. (2008). The influence of asynchronous dynamics in the spatial prisoner's dilemma game. In *Animals to Animats - 10th International Conference on the Simulation of Behavior (SAB'08)*. Springer-Verlag. To appear.

Hofbauer, J. and Sigmund, K. (1998). *Evolutionary Games and Population Dynamics*. Cambridge University Press.

Huberman, B. and Glance, N. (1993). Evolutionary games and computer simulations. *Proceedings of the National Academy of Sciences*, 90:7716–7718.

Newth, D. and Cornforth, D. (2007). Asynchronous spatial evolutionary games: spatial patterns, diversity and chaos. In *Proceeding of the 2007 IEEE Congress on Evolutionary Computation*, pages 2463–2470.

Nowak, M., Bonhoeffer, S., and May, R. M. (1994). More spatial games. *International Journal of Bifurcation and Chaos*, 4(1):33–56.

Nowak, M. and May, R. M. (1992). Evolutionary games and spatial chaos. *Nature*, 359:826–829.

Oh, J. C. (2001). Cooperating search agents explore more than defecting search agents in the internet information access. In *Proceedings of the 2001 Congress on Evolutionary Computation, CEC2001*, pages 1261–1268. IEEE Press.

Pacheco, J. M. and Santos, F. C. (2005). Network dependence of the dilemmas of cooperation. *Science of Complex Networks: From Biology to the Internet and WWW, CNET 2004*, 776:90–100.

Smith, J. M. (1982). *Evolution and the Theory of Games*. Cambridge University Press.

Szabó, G. (2007). Evolutionary games on graphs. *Phys. Rep.*, 446:97–216.

Tomassini, M., Luthi, L., and Giacobini, M. (2006). Hawks and doves on small-world networks. *Physical Review E*, 73(1):016132.

Watts, D. and Strogatz, S. H. (1998). Collective dynamics of small-world networks. *Nature*, 393:440–442.

Being Arranged in Advance: Quantum Entanglement and Biological Feedback

Taichi Haruna

Graduate School of Science & Technology, Kobe University, Kobe, JAPAN 657-8501
cheetha@kcc.zaq.ne.jp

Abstract

The categorical semantics of quantum protocols proposed by Abramsky and Coecke reveals that a prearranged quantum entanglement brings a strange quantum information flow in the quantum teleportation protocol. Their formal argument leads us to the distinction between an information flow sequence and a causal sequence on the same event. If this distinction is applied to information processing biological networks, we can claim that a prearranged biological feedback can play the same role as the quantum entanglement on the emergence of a specific local structure of networks. The aim of this paper is to provide a first step toward formal arguments on changes in biology without the external time parameter.

Introduction

If something is arranged in advance and if it works well, an apparently difficult or non-intuitive event can occur. A clear example of such a phenomenon is the quantum teleportation (Bennett et al., 1993; Nielsen and Chuang, 2000), in which a prearranged entangled pair of qubits allows an arbitrary quantum bit to be transferred from one site to the other site by a classical communication. In this paper we claim that biological feedbacks also play a role in the above mentioned type of phenomena. This paper is an engagement of two recent works. One is categorical semantics of quantum protocols by Abramsky and Coecke (2004), the other is algebraic study of biological networks by the author (Haruna, 2008; Haruna and Gunji, 2008), which is also based on category theory (Mac Lane, 1971).

Abramsky and Coecke (2004) clarifies the nature of quantum information flow by recasting the standard axiomatic quantum mechanics due to von Neumann (1932). What is the most relevant to us is that their formalism enables us to distinguish between an information flow sequence and a causal sequence on the same event. Our application of their result to biological networks immediately follows from this distinction.

The problem of prearrangement is central to changes in the realm of biology since biological changes including development and evolution are in general the process of imposing new constraints on the preceding constraints (Matsuno, 1989; Salthe, 1993; Kauffman et al., 2008). Matsuno (1989) argues that changes in biology could be described as the process of equilibration toward tentative final causes. Since the propagation speed of interactions in biological systems cannot be regarded as infinite, the tentative final causes can change as equilibration proceeds. Preceding equilibration constrains and triggers a new equilibration. Salthe (1993) considers evolution and development of hierarchical systems in terms of how the lower and upper levels constrain the dynamics of the focal level of a given system. Recently Kauffman et al. (2008) regards constraints as information for biological organizations to maintain themselves and evolve. Cascades of constraints lead to changes in biological organizations.

Biological feedback will be a typical example of prearrangement in biological systems. The term 'feedback' implies that succeeding events in a system have an impact on upstream processes in the system. Hence at least a 'path' for the feedback must be prearranged so that the feedback works effectively. We do not define what is prearrangement in biological systems in general but involve it with our formal argument implicitly. In particular we consider a biological feedback in information processing biological networks.

Our previous study (Haruna, 2008; Haruna and Gunji, 2008) on biological networks considers how to describe network motifs found in information processing biological networks. Network motifs are defined as local patterns that are found in real networks significantly more often than in an ensemble of suitably prepared random networks (Milo et al., 2002). They are considered to have certain biological functions (Alon, 2006, 2007). In information processing biological networks, each node in a network is considered to be an information processing unit. The direction of an arrow in a network indicates the direction of information flow. If a pattern of information processing is specified then we can deduce that how the information processing pattern constrains the local structure of networks (Haruna, 2008; Haruna and Gunji, 2008). However, it is not yet clear that how information flows at the network level is related to a causal sequence that brings the emergence of a network motif. We show that

a simple application of the category theoretical formalism of finite-dimensional quantum mechanics by Abramsky and Coecke (2004) can reveal this problem.

In this paper category theory is the main tool to argue the formal similarity between quantum entanglement and biological feedback. We believe that the generality of category theory is sometimes helpful to reveal unexpected common structure between different areas. The argument presented in this paper would provide a concrete example of such usefulness of category theory.

This paper is organized as follows. The next section is a brief overview of the quantum teleportation protocol and its categorical description by Abramsky and Coecke. In section III we review our algebraic study of network motifs. Section IV is the main part of this paper, where we show that how a causal sequence in an information processing biological network is reconstructed by a result obtained in the categorical description of quantum mechanics. In section V we give conclusions.

Categorical description of quantum teleportation

In this section we briefly review the quantum teleportation protocol (Bennett et al., 1993) and its category theoretical description (Abramsky and Coecke, 2004). The presentation here is minimal enough for the aim of this paper. For further details see the references. See also Coecke (2004).

The quantum teleportation protocol enables one to transfer an unknown quantum state from a source A to a remote target B by only two bits classical communication between them. The protocol involves three qubits a, b and c. Initially qubit a is in a state $|\varphi\rangle$ which is a unit vector in two-dimensional complex Hilbert space $\mathcal{H} = \{\alpha|0\rangle + \beta|1\rangle|\alpha, \beta \in \mathbb{C}\}$. Qubits b and c are prearranged as an entangled state, $\frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$, which is a unit vector in the tensor product $\mathcal{H} \otimes \mathcal{H}$. We abbreviate $|i\rangle \otimes |j\rangle$ as $|ij\rangle$ for $i, j = 0, 1$. Entangled states are defined as states that cannot be written in the form $|\phi_1\rangle \otimes |\phi_2\rangle$ for any choice of $|\phi_1\rangle$ and $|\phi_2\rangle$. Entangled states play important roles in the field of quantum information (Nielsen and Chuang, 2000).

We relocate the three qubits so that a and b are at the source A and c is at the target B. Now we perform so called *Bell-base measurement* on a and b. Each projector $P_i (i = 1, 2, 3, 4)$ associated with Bell-base measurement projects onto one of the one-dimensional subspaces spanned by the following vectors:

$$b_1 = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle),\ b_2 = \frac{1}{\sqrt{2}}(|01\rangle + |10\rangle),$$
$$b_3 = \frac{1}{\sqrt{2}}(|00\rangle - |11\rangle),\ b_4 = \frac{1}{\sqrt{2}}(|01\rangle - |10\rangle).$$

The four outcomes of the measurement occur with equal probability, $\frac{1}{4}$. We observe the outcome of the measurement and send it from A to B. This requires classical two bits. Based on this classical information, we 'correct' the qubit c by performing one of the following unitary transformation on it:

$$U_1 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix},\ U_2 = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix},$$
$$U_3 = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix},\ U_4 = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix}.$$

After the unitary correction, one can see that the state of c is $|\varphi\rangle$.

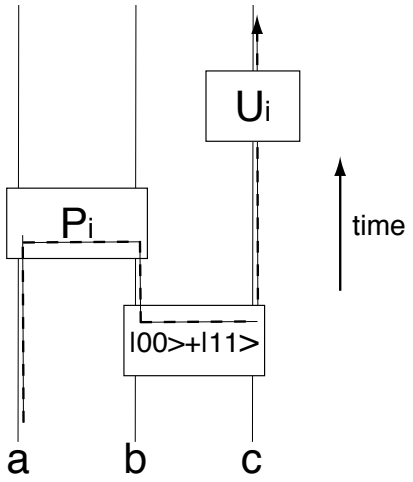


Figure 1: Quantum information flow in the teleportation protocol. The dashed arrow represents the quantum information flow.

For each observational branch, the quantum information flow seems to be 'acausal' as shown in Fig.1. Abramsky and Coecke (2004) proves that such a strange character of the quantum teleportation protocol can be captured at a more abstract level independent of the classical information flow by reformulating the finite-dimensional quantum mechanics from category theoretical point of view. A key point is that they distinguish two type of measurements appearing in the quantum teleportation protocol: one is the preparation of quantum states and the other is the indeterministic observation. These two type of measurements can be clearly distinguished by the notion of *compact closed category* (Kelly and Laplaza, 1980).

A *symmetric monoidal category* is a category $\mathcal{C}$ equipped with a *tensor product*

$$- \otimes - : \mathcal{C} \times \mathcal{C} \to \mathcal{C},$$

a *unit object* I and natural isomorphisms

$$l_A : A \cong I \otimes A,\ r_A : A \cong A \otimes I,$$
$$a_{A,B,C} : A \otimes (B \otimes C) \cong (A \otimes B) \otimes C,$$
$$s_{A,B} : A \otimes B \cong B \otimes A$$

for objects A, B, C in $\mathcal{C}$. These natural isomorphisms are required to satisfy certain coherence conditions (Mac Lane, 1971).

The definition of compact closed category by Kelly and Laplaza (1980) is as follows. A category $\mathcal{C}$ is a *compact closed category* if it is a symmetric monoidal category such that for each object A there are a dual object A^*, a *unit*

$$\eta_A : I \to A^* \otimes A$$

and a *counit*

$$\epsilon_A : A \otimes A^* \to I.$$

These data are required to satisfy the commutative diagram

$$\begin{array}{ccccc} A & \xrightarrow{r_A} & A \otimes I & \xrightarrow{1_A \otimes \eta_A} & A \otimes (A^* \otimes A) \\ {\scriptstyle 1_A}\downarrow & & & & \downarrow{\scriptstyle a_{A,A^*,A}} \\ A & \xleftarrow{l_A^{-1}} & I \otimes A & \xleftarrow{\epsilon_A \otimes 1_A} & (A \otimes A^*) \otimes A \end{array}$$

and the dual one for A^*. In other words, if a symmetric monoidal category $\mathcal{C}$ is seen as a bicategory with a single 0-cell, the 1-cells being the objects of $\mathcal{C}$ with the tensor product as their composition and the 2-cells being the morphisms of $\mathcal{C}$, the above conditions say that each object A of $\mathcal{C}$ has a right adjoint A^*. The required diagrams are 'triangular identities'.

The monoidal category of finite-dimensional vector space over a field is compact closed. This example corresponds to finite-dimensional quantum mechanics. The category of sets and relations with cartesian product $(\mathcal{R}el, \times)$ is also compact closed. This example is our main consideration in this paper. In $(\mathcal{R}el, \times)$, a one-point set $\{*\}$ is the unit object. For a set X, its dual X^* is itself, $X^* = X$. The unit for a set X is $\eta_X \subseteq \{*\} \times (X \times X)$ given by

$$\eta_X = \{(*, (x, x))|x \in X\}.$$

Similarly, the counit for X is

$$\epsilon_X = \{((x, x), *)|x \in X\}.$$

The *name* $\ulcorner f \urcorner$ and *coname* $\llcorner f \lrcorner$ of a morphism $f : A \to B$ in a compact closed category are defined by the following diagrams:

$$\begin{array}{ccc} A^* \otimes A & \xrightarrow{1_{A^*} \otimes f} & A^* \otimes B \\ {\scriptstyle \eta_A}\uparrow & & \| \\ I & \xrightarrow{\ulcorner f \urcorner} & A^* \otimes B \end{array}$$

$$\begin{array}{ccc} A \otimes B^* & \xrightarrow{\llcorner f \lrcorner} & I \\ \| & & {\scriptstyle \epsilon_B}\uparrow \\ A \otimes B^* & \xrightarrow{f \otimes 1_{B^*}} & B \otimes B^* \end{array}$$

In particular, we have $\eta_A = \ulcorner 1_A \urcorner$ and $\epsilon_A = \llcorner 1_A \lrcorner$.

In the following we will see that a name corresponds to a preparation of an entangled quantum state and a coname corresponds to an observational branch resulting from the indeterminism of quantum measurements (Abramsky and Coecke, 2004).

For a morphism $\rho : X \to Y$ in $(\mathcal{R}el, \times)$ we have

$$\ulcorner \rho \urcorner = \{(*, (x, y))|x \rho y, x \in X, y \in Y\},$$
$$\llcorner \rho \lrcorner = \{((x, y), *)|x \rho y, x \in X, y \in Y\}.$$

The compositionality lemma proved in (Abramsky and Coecke, 2004) is the most significant for our argument. It says that the following diagram commutes in any compact closed category:

$$\begin{array}{ccccc} A & \xrightarrow{f} & B & \xrightarrow{g} & C \\ {\scriptstyle r_A}\downarrow & & & & {\scriptstyle l_C^{-1}}\uparrow \\ A \otimes I & \xrightarrow{1_A \otimes \ulcorner g \urcorner} & A \otimes B^* \otimes C & \xrightarrow{\llcorner f \lrcorner \otimes 1_C} & I \otimes C \end{array}$$

The compositionality lemma captures the quantum information flow in the quantum teleportation protocol at an abstract level. The lemma yields the equation

$$U \circ (l_A^{-1} \circ (\llcorner f \lrcorner \otimes 1_A)) \circ ((1_A \otimes \ulcorner g \urcorner) \circ r_A) = U \circ g \circ f,$$

where all morphisms U, g, f have the same domain and codomain A (Fig.2). The original quantum teleportation protocol requires $U \circ g \circ f = 1_A$, however, any composition of morphisms in a compact closed category enjoys the inversion of the order of composition.

The right hand side of the above equation represents the sequence of quantum information flow on one hand, the left hand side represents the causal sequence of quantum measurements and a unitary transformation on the other hand. This distinction between an information flow sequence and a causal sequence is the essential point of our application of the compositionality lemma to biological networks in the following sections.

Algebraic description of network motifs

Network motifs are local patterns in networks that are considered to have certain biological functions (Milo et al., 2002; Alon, 2006, 2007). In particular, a four-nodes network motif called *bi-fan* (Fig.3) is found ubiquitously in information processing biological networks such as gene transcription regulation networks, signal transduction networks and neuronal networks (Milo et al., 2002; Alon, 2006). In this section we explain how the bi-fan motif emerges from an information processing pattern in a network (Haruna, 2008; Haruna and Gunji, 2008).

A node in an information processing network is considered to have an information processing ability. We assume that it has a specific internal structure that represents how

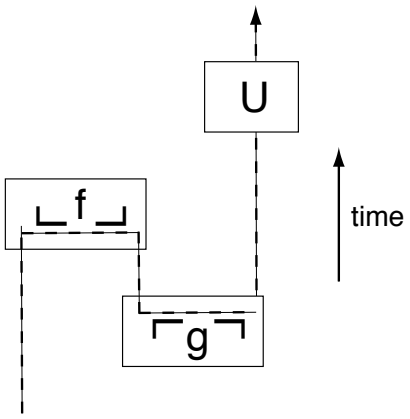


Figure 2: The essential feature of the quantum information fbw can be captured in any compact closed category.

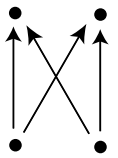


Figure 3: A network motif bi-fan.

it processes information. A simple but non-trivial internal structure considered here consists of two distinct nodes and an arrow between the two nodes:

$$\bullet \rightarrow \bullet$$

The source node, the arrow and the target node are considered to represent reception of information, transformation of information and sending of information, respectively. For example, each node in a gene transcription regulation network is a gene or a protein coded by the gene. They together represent a single node. Hence we can consider information processing in each node: possible regulations from other proteins (reception of information), synthesis of the protein from the gene via the transcription and translation processes (transformation of information) and possible regulations of other genes by the protein (sending of information).

If two nodes with this internal structure are connected by an arrow in the network, this connection by the arrow in the network is represented by a pattern shown below, in which the target in the internal structure of the source node is identifi ed with the source in the internal structure of the target node:

$$\bullet \rightarrow \bullet \rightarrow \bullet$$

Again in gene transcription regulation networks, each arrow in a network indicates a regulation from the source gene to the target gene. The protein synthesized by the source gene

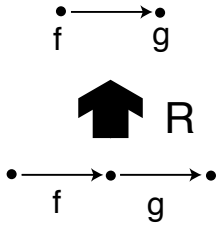


Figure 4: An arrow with its source and target nodes in a network is an image of M by R (See text).

is responsible for the regulation and is included in both sending of information at the source gene and reception of information at the target gene. This motivates us to introduce the above pattern.

We call the pattern *information processing pattern*, which is referred to as M in what follows. Thus an arrow with its source and target nodes in the network can be seen as the image of M by a graph transformation R defi ned as follows (Fig.4).

Let $G = (A_G, O_G, \partial_0^G, \partial_1^G)$ be a directed graph, where A_G is a set of arrows, O_G is a set of nodes and ∂_0^G and ∂_1^G are maps from A_G to O_G. ∂_0^G sends an arrow to its source. ∂_1^G sends an arrow to its target. We defi ne $RG = (A_{RG}, O_{RG}, \partial_0^{RG}, \partial_1^{RG})$ as

$$A_{RG} = \{(f,g) \in A_G^2 | \partial_1^G f = \partial_0^G g\},\ O_{RG} = A_G,$$
$$\partial_0^{RG}(f,g) = f,\ \partial_1^{RG}(f,g) = g.$$

The graph transformation R can be seen as a functor from the category of directed graphs $\mathcal{G}rph$ to itself. The directed graph RG is so called the line graph of G.

In general, we consider the constraint to a local pattern F in a network imposed by the information processing pattern M as that the local pattern F is isomorphic to an image of R, that is, we can write $F \cong RG$ for some G. It can be shown that the condition is equivalent to $\eta_F : F \cong RLF$, where L is a left adjoint to R and η is the unit of the adjunction (Haruna and Gunji, 2008). It is proved that for any information processing pattern M we can construct a corresponding adjoint pair (L, R) (Haruna, 2008). However, the condition $\eta_F : F \cong RLF$ is not equivalent to the condition $F \cong RG$ for some G in general. Here we do not go into the general argument but directly defi ne the left adjoint L.

For a directed graph $G = (A_G, O_G, \partial_0^G, \partial_1^G)$, a directed graph LG consists of the following data:

$$A_{LG} = O_G,\ O_{LG} = (O_G \times \{0,1\})/\sim,$$
$$\partial_0^{LG} x = [(x,0)],\ \partial_1^{LG} x = [(x,1)],$$

where $\sim$ is an equivalence relation generated by a relation ρ defi ned by $(x,1)\rho(y,0) \Leftrightarrow x \rightarrow y$ and $[(x,i)]$ is the equivalence class containing (x,i). We write $x \rightarrow y$ if there is an arrow from x to y in G.

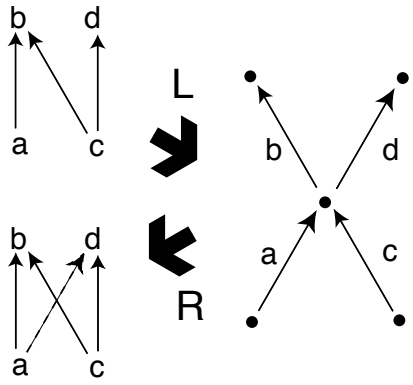


Figure 5: Explanation of the necessary condition for η_F : $F \cong RLF$.

Intuitively, L is a structuration of a pattern by the information processing pattern M because it replaces nodes with arrows. On the other hand, R is a de-structuration of a pattern with respect to M because it collapses an arrow to a node.

The necessary and sufficient condition for $\eta_F : F \cong RLF$ is that F is a binary graph (that is, there is at most one arrow between two nodes) and if $a \to b \leftarrow c \to d$ then $a \to d$ in F. We here explain that the latter condition is necessary. Suppose $a \to b \leftarrow c \to d$ in F (the upper left pattern in Fig.5). If L is performed on this pattern then we obtain the pattern at the right-hand side of Fig.5. If R follows then bi-fan emerges (the lower left pattern in Fig.5) as the dashed arrow is newly added as the fourth arrow.

One can see that $\eta_F : F \cong RLF$ is the condition that the information processing pattern M is fully developed or stabilized in a pattern F. What does happen in this developing process? The key point is what occurs at the central node in the right-hand side pattern in Fig.5. It is an equivalence class consisting of $(a,1),(b,0),(c,1)$ and $(d,0)$. The newly added fourth arrow from a to d appears since $(a,1)$ is identified with $(d,0)$. This identification process is due to the transitive relation $(a,1)\rho(b,0)\rho^{-1}(c,1)\rho(d,0)$. Since $(x,1)\rho(y,0)$ means $x \to y$ in the network level, $(b,0)\rho^{-1}(c,1)$ indicates the existence of feedback from b to c, the direction of which is opposite to the direction of the information flow at the network level.

If we try to interpret the process of emergence of bi-fan described above in terms of the information flows at the network level then an apparent difficulty arises. Since an arrow in information processing biological networks represents the direction of the information flow, it seems that there is no information flow sequence from a to d at the network level. Then how is it possible to construct a connection between a and d? This difficulty arises since the information flows at the network level only cannot treat the above mentioned feedback relation. The difficulty in the interpretation can be resolved when we consider an information flow sequence including the feedback relation and a causal sequence on it.

Reconstruction of causal sequence in biological networks

We reconstruct a causal sequence that brings the emergence of bi-fan through the compositionality lemma by Abramsky and Coecke (2004). We work in the category of sets and relations $(\mathcal{R}el, \times)$ which is compact closed.

We apply the compositionality lemma to the composition $\rho \circ \rho^{-1} \circ \rho$ which brings the fourth arrow in bi-fan. We regard the order of composition represents an information flow sequence including feedback relation ρ^{-1} from b to c, which should be distinguished from the information flows at the network level.

Given a directed graph $F = (A_F, O_F, \partial_0^F, \partial_1^F)$, we put $X = \{(x,0)|x \in O_F\} \cup \{(x,1)|x \in O_F\}$. The compositionality lemma applied to the right-hand side composition of $\rho \circ \rho^{-1} \circ \rho$ gives rise to the following commutative diagram:

$$\begin{array}{ccccc} & & & & X \\ & & & & \uparrow{\scriptstyle \rho} \\ X & \xrightarrow{\rho} & X & \xrightarrow{\rho^{-1}} & X \\ \downarrow{\scriptstyle r_X} & & & & \uparrow{\scriptstyle l_X^{-1}} \\ X \times \{*\} & \xrightarrow{1_X \times \ulcorner \rho^{-1} \urcorner} & X \times X \times X & \xrightarrow{\llcorner \rho \lrcorner \times 1_X} & \{*\} \times X \end{array}$$

The sequence of arrows from the upper left X to the upper right X along the lower side is interpreted as a causal sequence. The feedback relation ρ^{-1} is at the same position as the preparation of entangled qubits pair in the quantum teleportation protocol (Fig.6). The feedback relation ρ^{-1} between b and c is prearranged, so that the information flow from a to d occurs.

In the quantum teleportation case, the causal sequence is the sequence of our operations on the quantum system. A specific causal sequence of our operations enables an 'acausal' quantum information flow to occur. However, in our information processing biological network case, the relation between the information flow sequence and the causal sequence is reversed. We have to reconstruct a causal sequence from a given information flow sequence. Hence there may be an ambiguity in the reconstruction. Indeed, we can also reconstruct a causal sequence in a different way if we apply the compositionality lemma to the left-hand side

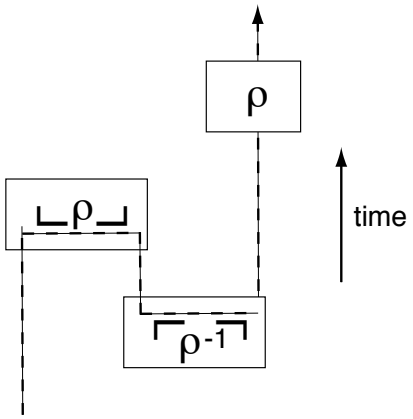


Figure 6: Reconstruction of a causal sequence on the emergence of bi-fan. The information flow structure is isomorphic to that of the quantum teleportation protocol.

composition of $\rho \circ \rho^{-1} \circ \rho$:

$$
\begin{array}{ccccc}
X & & & & \\
{\scriptstyle \rho}\downarrow & & & & \\
X & \xrightarrow{\rho^{-1}} & X & \xrightarrow{\rho} & X \\
{\scriptstyle r_X}\downarrow & & & & \uparrow{\scriptstyle l_X^{-1}} \\
X \times \{*\} & \xrightarrow{1_X \times \ulcorner \rho \urcorner} & X \times X \times X & \xrightarrow{\llcorner \rho^{-1} \lrcorner \times 1_X} & \{*\} \times X
\end{array}
$$

Contrary to the first reconstruction, the feedback relation ρ^{-1} is at the trail in the corresponding causal sequence (Fig.7). In order to determine a reconstructed causal sequence uniquely, we need a *selection rule*. Since ρ^{-1} is regarded as a biological feedback, it should appear as early as possible in the causal sequence. So the first reconstruction is desired in this respect. If we want to select the first reconstruction, the following rule is sufficient for this example:

ρ^{-1} must be transformed into $\ulcorner \rho^{-1} \urcorner$.

The general argument on how to define a selection rule is beyond the scope of this paper. It is left as a future work.

The difference between $(\mathcal{R}el, \times)$ and the category of finite-dimensional vector spaces over a field should also be noted. Both categories can implement full abstract quantum mechanics with some additional structures on one hand, the former cannot enjoy the full quantum teleportaion protocol since it has no Bell-base consisting of four vectors (Abramsky and Coecke, 2004). Hence the usage of the term 'information flow' in this paper is different from that in the references (Abramsky and Coecke, 2004; Coecke, 2004). They consider information flows including the conservation of contents of information. However, we never refer to contents of information but consider only the formal structure among information flows.

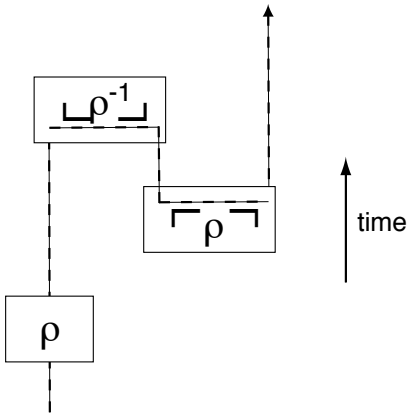


Figure 7: Another reconstruction of a causal sequence on the emergence of bi-fan.

Conclusions

Our argument in this paper is based on a specific example and has not yet developed with full generality. However, we can extract a general strategy to describe changes in biological systems without the explicit external time parameter:

(i) Make a distinction between information flow sequence and causal sequence.

(ii) Assume selection rules considering what should be prearranged.

(iii) Reconstruct the causal sequence from the information flow sequence based on the selection rules.

We hope that our argument presented in this paper will help to understand the universal role of information processing in natural phenomena ranging from quantum to biological regimes.

Acknowledgements

The author was supported by JSPS Research Fellowships for Young Scientists.

References

Abramsky, S. and Coecke, B. (2004). A categorical semantics of quantum protocols. In *Proceedings of the 19th Annual IEEE Symposium on Logic in Computer Science*. IEEE Computer Science Press, arXiv:quant-ph/0402130.

Alon, U. (2006). *Introduction to Systems Biology: Design Principles of Biological Circuits*. CRC Press, Boca Raton.

Alon, U. (2007). Network motifs: theory and experimental approaches. *Nature Review Genetics*, 8:450–461.

Bennett, C. H., Brassard, G., Crépeau, C., Jozsa, R., Peres, A., and Wooters, W. K. (1993). Teleporting an unknown quantum state via dual classical and einstein-podolsky-rosen channels. *Physical Review Letters*, 70:1895–1899.

Coecke, B. (2004). The logic of entanglement. arXiv:quant-ph/0402014.

Haruna, T. (2008). *Algebraic Theory of Biological Organization*. Doctral Dissartation, Kobe University.

Haruna, T. and Gunji, Y. P. (2008). Wholeness and information processing in biological networks: An algebraic study of network motifs. In *Postceedings of 2nd Internal Workshop on Natural Computing, Natural Computing 2007: Nagoya*. Springer-Verlag, to appear.

Kauffman, S., K., L. R., Este, R., Goebel, R., Hobill, D., and Shmulevich, I. (2008). Propagating organization: An enquiry. *Biology and Philosophy*, 23:27–45.

Kelly, G. M. and Laplaza, M. L. (1980). Coherence for compact closed categories. *Journal of Pure and Applied Algebra*, 19:193–213.

Mac Lane, S. (1971). *Categories for the Working Mathematician*. Springer-Verlag,New York.

Matsuno, K. (1989). *Protobiology: Physical Basis of Biology*. CRC Press, Boca Raton.

Milo, R., Shen-Orr, S., Itzkovitz, S., Kashtan, N., Chklovskii, D., and Alon, U. (2002). Metwork motifs: Simple building blocks of complex networks. *Science*, 298:824–827.

Nielsen, M. A. and Chuang, L. (2000). *Quantum computation and quantum information*. Cambridge University Press.

Salthe, S. N. (1993). *Development and Evolution: Complexity and Change in Biology*. MIT Press, Cambridge.

von Neumann, J. (1932). *Die Mathematische Grundlagen der Quantenmechanik*. Springer Verlag, Berlin.

Misrepresentations

Inman Harvey

Evolutionary and Adaptive Systems Group
Centre for Computational Neuroscience and Robotics
University of Sussex
Brighton, UK
inmanh@sussex.ac.uk

Abstract

The concept of “representations”, and particularly “internal representations”, can be controversial in Cognitive Science and AI. It is suggested here that much time-wasting confusion could be avoided if participants in such controversies came to recognize the variety of different senses, often incompatible, in which such terms are used. A hypothesis is presented as to why there is so much reluctance to recognize this. Once such fruitless controversies are swept aside through linguistic hygiene, there remain interesting real problems, which are eminently appropriate for being tackled by an Artificial Life methodology.

Introduction

There are many confusions and misunderstandings associated with the term "representation" in Cognitive Science and AI, and by extension in Artificial Life. It will be argued here that most of these problems can be solved (or dissolved) by careful linguistic hygiene. But there remain interesting and genuine problems that can be fruitfully approached using an artificial life, evolutionary robotics methodology.

Representation wars

Artificial life overlaps with AI, in that both tackle the problems involved in synthesising lifelike capabilities; there may be different emphases, perhaps on adaptive behaviour versus rational thought. AI overlaps with cognitive science. All of these are permeated by the positions researchers may take on philosophical issues: what is life, what is cognition, what is mind? Traditional GOFAI (Good Old Fashioned AI) approaches to these questions have often framed answers in terms of "representations", or "internal representations". These sort of notions made no sense to me, working in a GOFAI department, and some of our early artificial life experiments at the beginning of the 90s had as one motivation the intent to make such issues explicit (Cliff et al 1993; Harvey et al 1993). Using evolutionary robotics techniques, we evolved simple minimally cognitive agents to perform simple tasks, and then challenged the GOFAI theorists to try and identify just where these so-called "internal representations" were.

We used genetic algorithms to evolve the connectivity, the connection weights, and the temporal parameters for real-time recurrent artificial neural networks, that formed the "nervous systems" for agents interacting dynamically with their environment. This is within the Dynamical Systems approach to understanding cognition, and our motivation for this continuing line of research at Sussex (Harvey et al 1997, Harvey et al 2005) is very much in sympathy with similar research by Beer (Beer 1995, Beer 2000) and others. What were the responses of the GOFAI theorists?

Some of them claimed that there simply must be internal representations in these agents, though there was little agreement on just where to find them. Others claimed that maybe for the simple cognitive tasks internal representations were not necessary, but for more complex forms of cognition — "representation-hungry tasks" — they would be essential. Some claimed it was logically necessary for brains (or minds) to have representations; others claimed it was a pragmatic necessity — they simply could not imagine how one could design brains to work in any other way.

It became clear that there were many conflicting notions of representation being bandied about. I was prepared to offer my definition of the term (Harvey 1996), but to my surprise it was extremely difficult, if not impossible, to pin down the other protagonists in these representation wars to offer their own definition of the term. In the way that I use the term, I have never ever had any internal representation of any kind in my head — except in the most casual, metaphorical sense. When I say I have a map of Brighton in my head, this is shorthand for saying I can navigate as if I had a map in my hand, I can visualise the configuration of a map as if it were in front of me — but I certainly do not mean that there literally is a map in my head. Others disagreed; but what was the nature of this disagreement?

The lack of communication became so dispiriting that eventually it seemed to me a waste of time continuing these debates. It is perfectly possible to do artificial life style research into cognitive systems, from an enactive or Dynamical Systems perspective, without ever having to engage in such discussions on representations. However it

seems that many people are still being sucked into the same old futile confusions (Grush, 1997; Clark and Grush, 1999; Grush, 2004; Wheeler, 2005; Rowlands 2006; Gallagher, forthcoming), so that here we revisit the representation wars yet again.

The position presented here is that there ***are*** important interesting issues concerning representations that can be usefully tackled with an artificial life, evolutionary robotics approach. But that almost all the debate I see on this topic has nothing to do with such issues, but is rather symptomatic of confused and unclear talk, specifically: multiple incompatible usages for the term "representation".

The Pavement Problem

Imagine an international conference of urban traffic specialists. They meet for extended discussions on the question of what restrictions are desirable, for safety reasons, to place on bicycles riding on the pavement. The discussions are intense, confused and chaotic; there is a complete lack of agreement. The participants returned home demoralised, and astonished at the obtuseness of many of the suggestions that they had heard. It turns out that many of those present were not aware that "pavement" means the part of the street that cars drive on in North American English, and the side of the street (or sidewalk) where pedestrians walk in British English. Even worse, it turns out that some present actually *were* aware of this crucial ambiguity, but did not feel it necessary to comment that this lay at the root of all the confusion.

Fortunately — we hope — urban traffic experts are not so stupid. Their work is important and can save lives; they need to think and explain themselves clearly. What a pity that so many philosophers do not live up to these basic standards.

When people use the term "representation" in the context of cognitive science, it is not nearly so simple as having merely one or two well-defined meanings. There is a constellation of different, overlapping academic interests, from neuroscience to philosophy of mind, each with their own different perspectives on the term. Furthermore, the richness of the variety of usages of the term "representation" means that it often has multiple, incompatible referents within any one such academic discipline. In the rest of this paper I shall categorise some of these usages; and then I shall speculatively offer a possible explanation why people are so often so very reluctant to clarify which of these many possible senses they are using.

The Everyday Sense of the Word Representation

Before going into any technical sense in which the word might be used, let us look at its everyday meaning. Consulting a dictionary (www.dictionary.net, based on Websters) for the term "represent" (since "representations" include 'the act of representing, in any sense of the verb') we find a list of 8 variants of increasing sophistication. I shall return, below, to the last two listed, but can summarise briefly here the first six, as everyday meanings: to represent is to present again or anew, to present by means of something standing in the place of, to exhibit the counterpart or image of, to typify. This extends to serving as a sign or symbol of — words represent ideas or things.

In this sense, it seems to me that human beings are supremely representation users, ever since the dawn of cave art and of language. Our use of representations is, above all, what differentiates us from other animals; we can argue about grey areas such as chimpanzees' use of sign language, or the sexual displays of peacocks, but our human usage of representations is orders of magnitude more complex and comprehensive. We humans live in language and culture, as a fish lives in the water.

What can we do, now that we have this useful trick of representing? If we want to draw somebody's attention to a cat, instead of bringing a cat out of one's pocket and waving it around, one can draw a picture of a cat, or even more conveniently use the word "cat". This is a sophisticated and incredibly useful trick, so useful that we spend much of our childhood learning how to use and extend our capacities for representation. At the most sophisticated end of the spectrum, we can reason with mathematical symbols, and write computer programs so that machines can reason for us.

Representation is a relational term

In this basic sense, representation is a relational term like North or Twin. 'Brighton is to the north' is ambiguous without a context — north in relation to where? Brighton is north of Paris and south of London, so a disagreement as to whether it is or is not "to the north" can sometimes be easily settled by establishing this context. Likewise, twin is a relational concept. Any number of exhaustive tests on a child cannot settle the question of whether that child is or is not a twin. Twinness can refer only to its relationship with a second child. Very often the context — the relational partners to these relational terms — are implicit and obvious, leaving no scope for disagreement. In the representation wars, across different disciplines and across different sets of starting assumptions, it is crucial to recognize that there simply is no such universal agreement on the context.

A symbol **P** is used by a person **Q** to represent, or refer to, an object **R** to a person **S**. Nothing can be referred to without somebody to do the referring. Normally **Q** and **S** are members of a community that have come to agree on their symbolic usages, and training as a mathematician involves learning the practices of such a community. The vocabulary of symbols can be extended by defining them in terms of already-recognised symbols.

The English language and the French language are systems of symbols used by people of different language communities for communicating about their worlds, with their similarities and their different nuances and clichés. The languages themselves have developed over thousands of years, and the induction of each child into the use of its native language occupies a major slice of its early years. The fact that, nearly all the time we are talking English, we are doing so to an English-speaker (including when we talk to ourselves), makes it usually an

unnecessary platitude to explicitly draw attention to the community that speaker and hearer belong to.

Since symbols and representation stand firmly in the linguistic domain, another attribute they possess is that of some element of arbitrariness (from the perspective of an observer external to the communicators). When I raise my forefinger with its back to you, and repeatedly bend the tip towards me, the chances are that you will interpret this as `come here'. This particular European and American sign is just as arbitrary as the Turkish equivalent of placing the hand horizontally facing down, and flapping it downwards. Different actions or entities can represent the same meaning to different communities; and the same action or entity can represent different things to different communities.

In the more general case, and particularly in the field of connectionism and cognitive science, when talking of representation (in the sense outlined above) it is imperative to make clear who the users of the representation are. In particular it should be noted that where one and the same entity can represent different things to different observers, conceptual confusion can easily arise. When in doubt, always make explicit the **Q** and **S** when **P** is used by **Q** to represent **R** to **S**.

Of course, it is open for people to choose to use the word in some different, technical, sense that may not fit into this format; but then it is obligatory to make clear just how this different sense is defined.

The Homuncular Representation

We shall return to the more sophisticated meanings listed for "represent" in the dictionary further on, but let us first explore common extensions of the basic meanings. When we try and explain or describe complex systems, to other people or to ourselves, then it is standard practice, good common sense, to draw on metaphors from everyday life. We often represent the component parts of the mechanism in terms of homunculi, or little imaginary people, who are performing different functions in coordination with each other. The thermostat measures the temperature, and tells the central heating boiler when it should switch on — for many purposes this shorthand, that personifies (or "homuncularises") the different components, is so much clearer and more useful than any detailed mechanical description. Notice how there are two levels of representation going on here: at one level we are representing the thermostat (*to the reader*) as, in effect, a little homunculus; and at another level the signal travelling down the wire from the thermostat represents the temperature or the command that the thermostat-homunculus "intends" (metaphorically) to convey *to the boiler-homunculus*. Here, for clarification, I have *italicized* the different "recipients" **S** for the two different instances of representation.

Now what are the requirements for this signal to represent the temperature? Firstly, there must be some correspondence or covariance between variations in temperature and variations in signal. Secondly, the signal must play some functional role, in communicating with whatever plays the role of a receiver of the signal. There is, in my view, a third requirement that may be more controversial. There needs also to be a further metaphorical homunculus that acts as the sender of the signal. The metaphor requires one homunculus communicating with another one.

When we follow somebody on a walk in the countryside, they could leave an indication of which route to take at a fork in the path by drawing an arrow; this is a representation of the desired direction. Alternatively, we could just trace their footprints in the mud; but we would not call these a representation, or at any rate not in the same sense. Likewise, I would suggest, the homunculi metaphor implicitly requires the active connivance of both sender and receiver — the arrow counts, but the footprints do not.

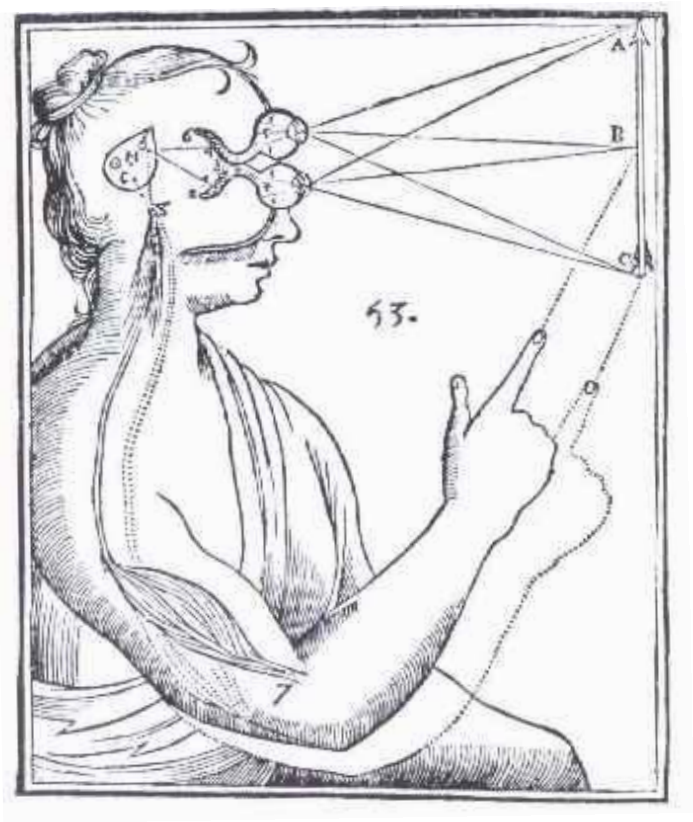
Not everyone will agree with this third requirement. Fair enough, but then you are using the word in a different sense from that which I propose. We should acknowledge and recognize that there are these different senses of the word representation, we should make explicit which sense we intend if we are to avoid confusion.

Is this signal travelling down the wire from the thermostat ***really*** a representation? Well yes, in this sense: the coin that I use on my chessboard to replace a lost piece ***really*** does represent the black king. Within the rules of the game it plays this role. When we used the homuncular metaphor there are these rules for the game.

I should stress that when I call this form of explanation "metaphorical" and "homuncular" this is certainly not intended to be disparaging, nor to mean that such explanations are any sense illegitimate or second rate. They are legitimate and invaluable forms of explanation, absolutely essential when we are talking in terms of functions, and we simply could not attempt to start to understand complex systems without employing them. We explain the strange and unfamiliar in terms of the familiar, and what could be more familiar than examples of inter-personal communication from our everyday life. But we should be aware of the baggage that this form of explanation carries with it.

More sophisticated senses of the term representation

Returning to definitions of "represent" in the online dictionary, items seven and eight were "to bring a sensation of into the mind or sensorium", and (Metaphorically) "to form or image again in consciousness, as an object of cognition or apprehension (something which was originally apprehended by direct presentation". Now personally I have trouble with these definitions. Apart from any other considerations, sensorium is not in my everyday vocabulary, and I have to go and look that up. My suspicion is that these definitions carry with them a certain baggage, certain philosophical assumptions that I may not share. For the purposes of this paper, however, I need only comment that these senses either do, or do not, share the basic properties outlined above. If they do share these properties, then to avoid confusion we



From Descartes (1662), *De Homine.* Visual information travels to the brain through hollow optic nerves. It continues on to the Pineal body, that regulates the flow of animal spirits to the nerves, and thence the muscles.

must be willing to spell out the **P**, **Q**, **R**, **S**. If they do not share these properties then we must acknowledge that these are different senses of the term.

As an aside on that final definition of represent above, we can notice the differences and similarities between the propositions "I imagined seeing a car" and "I saw an imaginary car". Superficially they appear to say the same thing, but the second version carries overtones of asserting the existence of "an imaginary car". With this second version, we might be more tempted to ask just where this imaginary car is. If it is not on the street, then where is it — in the consciousness, in the mind, in the brain? I would like to suggest that asking these questions is as foolish as asking where the twinness of a twin resides — is it in the DNA, where should we look for it? For the purposes of this paper, however I do not need to argue further in favour of my philosophical views or against other people's. I merely need to observe that different people use the term representation (and imagination) in different ways, and we must acknowledge that there are different senses of the terms; think "pavement" and "pavement".

Neuroscience

One role of a neuroscientist is to attempt to explain how the mechanisms of the brain allow us to see (and indeed even to imagine seeing) objects in the world around us. This is the same task as Descartes set himself, though perhaps nowadays we have less emphasis on the pineal gland and the immaterial spirit. Consider the image of an arrow as projected onto the back of the retina. In the picture shown, we have multiple representational layers of meaning. This is a reproduction — a representation — of the original woodcut first printed in Descartes' De Homini ('On Mankind'), 1662. We can see sketched on the right-hand side a representation of a vertical arrow. For the subject pictured here, of course, this is not a representation but rather it is a real arrow in front of it. If we perform our linguistic hygiene carefully, we should be able to avoid any confusion as to whether this is, or is not, a representation by clarifying the contextual relationship within which we are using the term. But then what about the image projected on the retina?

We, the observers of Descartes's woodcut, can see this projection, this representation. But the subject pictured there cannot see this, no more than they can see their own blindspot. It is the arrow in the world in front of it that the subject sees. We can, however, construct a metaphorical homunculus story where there is a chain of information flow, a pipeline, such that this retinal image is an intermediate carrier of information from the external world to some further internal mechanisms. Descartes certainly had one version of the story, and nowadays neuroscientists have different versions of such a story.

So the world (conceived as a homunculus) is conveying the information about the arrow in the world via this retinal image, this representation, to — to whom? To a posited receiving homunculus that, for the neuroscientist, may be some further subsystem of the brain. This could be a further staging post on the pipeline, as the pineal gland in Descartes's version. This is closely related to the example of the thermostat representing the room temperature, or the thresholded temperature, by an electrical signal sent to the central heating boiler. Is it a representation or not — that depends on which context you are spelling out; think "pavement" and "pavement".

A taxonomy of representations

I have spelt out above my own understanding of what I mean when I use the term representation in an everyday sense. I am aware that other people may disagree on some of these terms and conditions. When we moved to discussion of "internal representations" in cognitive science I personally am unwilling to change any of these terms and conditions (unless a clearly different technical sense of the term is defined and agreed upon). It follows necessarily that it makes no sense at all to talk of internal representations in the brain, except in the limited homuncular sense that I have outlined above. These are limited in that they are not representations "for me", rather they are representations "for some homunculus". Even though I can imagine seeing a unicorn, or visualise a map of

Brighton, this does not mean that there is a representation (for me) of either in my brain. There are structures or changes in my neural circuitry associated with my ability to imagine these things; but these structures or changes are simply not representational in the sense that I have spelt out.

Other people use the term in different ways, so let us make a start on classifying the different dimensions of meaning involved. To clarify, this is not so much an attempt to find the ***real*** meaning of the word (though I have my personal preferences), but rather an attempt to make visible the variety of possible senses that are used, just as road traffic experts should make explicit the two interpretations of "pavement".

1. Is representation a relational term or not?
2. If relational, is it essential that we should be able to contextualise the sender only, or the receiver only, or both or neither?
3. For internal representations, are they internal to the mind, the brain, or both or neither?
4. Is an internal representation a concept at a personal level, a sub-personal level, a neuro-anatomic level, or what?
5. Is it a concept within a functional explanation, or some other form of explanation?
6. Are such representations intentional or non-intentional concepts?
7. For internal representations, if you believe that there is a receiver, then is this receiver the person, or some substructure of the brain, or something else?
8. Is an internal representation of a cat in front of my eyes the same thing, or different from, the internal representation when the cat has gone but I am thinking of it?
9. Suppose somebody can navigate around Brighton without a map. Is the claim that they must have an internal representation of Brighton a logical claim that adds nothing to the previous statement; or an empirical claim that the nervous system underlying this cognitive capacity must be organised in some particular way?

In my experience, a room full of cognitive scientists and philosophers will, when challenged, produce a whole range of different responses to these questions. Not only are there these differences between different people, but also the same person may use the same word in different contexts with different senses. Yet typically, even when this is pointed out to them, this does not often seen to make them want to clarify their meaning in future discussions. I shall shortly suggest a hypothesis as to why this is so.

Presentations and Representations

Grush (1997: expanded in Grush 2004) does make a start at defining the terms he uses, and in particular makes distinctions between “presentations” and “representations”; also between “simulations” and “emulations”. To briefly summarise, he distinguishes between the two senses in item 8 listed above. He calls the former sense, as with direct sensory inputs of a cat, ‘presentational’, and reserves the term ‘representational’ for ‘counterfactual presentations’ such as considering the cat when it is not there.

Such a careful distinction is commendable, is consonant with the sentiments of this paper, and is regrettably all too rare amongst contributors to these debates. However Grush does not go far enough. In leading up to this, he states:

> If this second definition, and my gloss on it, are correct, then a representation is a part of a three-way relationship which also includes a user and a target. So far so good. Some may quibble over the need for a user, but that is not where the real problem lies. The real problem has been, and continues to be, the choice of states for which theorists attempt to give a representational analysis. Specifically, sensory states have been used as a model for representational states, the idea presumably being that sensory states represent the world to the subject. (Grush 1997)

So he is committed to a relational sense of the term, but less concerned about just how many partners there are in such a relationship, seeing this as a quibble and not where the real problem lies. Now my personal practice is normally to use the term with 4 such partners (**P**, **Q**, **R** and **S**), but I am prepared to engage in a discussion where the participants have agreed to use a different (e.g. 2 or 3-partner version) provided this is done openly and consistently. But it is simply unacceptable to leave it open to different participants in a discussion to have different senses in mind — and a sense that requires 2 partners must be a different sense to one that requires 3.

Representations as the billiard balls of cognitive science

We typically try to explain the complex and unfamiliar in terms that are simpler and more familiar. This is commendable and natural. Physicists will often try to explain atoms or other elementary particles in terms of billiard balls. We do not need to question how billiard balls will travel in a straight line, until they bounce off a wall or collide into each other. If we were perverse, however, we might demand to ask further questions. After all, billiard balls are made out of molecules and atoms, so is this not a circular explanation that does not bottom out anywhere? Well, in one sense *yes*, but in another sense *no,* because that is to misunderstand the role of an explanation. An explanation has to find some level of agreement, where no further questions are asked, and then try to reduce the complex and unfamiliar to this level. So for most purposes we can treat a billiard ball as an *explanans*, the explanatory premises, and this is incompatible with treating the billiard ball as an *explanandum*, that which is to be explained.

After many years of puzzlement, I have formed the tentative hypothesis that this lies at the root cause of cognitive scientists and philosophers being so reluctant to define what they mean by "representations" (Grush here being a partial exception, who has not gone far enough). For them, representations are the billiard balls of cognitive science. They are so familiar to them that they do not need to explain them further. Rather, they use them as part of the premises on which they build their cognitive theories. This makes them annoyed and irritated, just as physicists with billiard balls,

when one seeks an explanation for these premises.

This makes for difficulties if one thinks, as I do, that the capacity of human beings to represent things (common or garden everyday representations, in the external world) is supremely important and interesting; is the *explanandum*, not the *explanans*. In the phylogenetic history of human cognition, surely this counts as one of the Major Transitions (Maynard Smith and Szathmary, 1995). How did organisms start to create patterns in the world as representations of other objects or events in the world? Both the origin of, and the maintenance of, a capacity to represent things are surely amongst the main challenges for cognitive science. Artificial life techniques have a possible role to play here.

Minimal Cognition

What is the relationship between the physical mechanisms incorporated in the physical body of an organism and its behavioural capacities? The minimal cognition route aims to tackle this sort of question by starting at the bottom and building minimal models of minimally cognitive agents in some virtual environment. Evolutionary robotics allows one to evolve nervous systems to generate, we hope, the desired behaviour. When we come to analyse these, the advantage over real organisms is that we have full knowledge of the inner workings of all the mechanisms, and we can manipulate, alter constraints, lesion and otherwise experiment at will.

This kind of minimal cognition experiment (Harvey et al, 1997, 2005; Beer, 1995, 2000) typically has the status of a thought experiment, and an existence proof. If an artificial organism generates behaviour comparable to that of a real biological organism by use of mechanism X, this does not by itself prove that the real organism uses a similar mechanism X. It merely adds mechanism X to the list of possibilities; it is a separate scientific question as to which mechanism the real organism actually uses. The Artificial Life experiment can still be a very useful exercise, particularly if mechanism type X was previously thought unfeasible.

There have been many artificial life studies on communication, and language, but for the most part these have had built into them the possibilities of communication. To look at the major transition of the origin of representations one should perhaps start much earlier than this. Relevant work here is by Di Paolo (Di Paolo 2000) on the origin of social coordination and by Quinn (Quinn et al, 2002, 2003) on team behaviour. These studies look at communication between agents, two at a time (in simulation) in the first case, a group of three (both in simulation and on real robots) in the second case, where they have motives to influence each other's behaviour, and to do so via their actions. In the Quinn example, three agents or robots can sense each other through short range sensors, and move around on the plane. Their task is to travel across the plane in formation, which because of their sensors is only possible if they travel in a column of three. They are initially identical, so to achieve this task the first requirement is that they sort out between them the roles of leader, man in the middle, and tail. In some of the experiments a possible interpretation of what happens is that the symmetry is broken by the first one to make a stereotypical movement, which then determines its role. The others in some sense recognize this movement, and then take on the other roles.

It would be difficult (but not impossible) to claim that there is a fully-fledged representation going on here: "this stereotypical movement of mine, when suitably responded to by you, specifies or determines or represents what roles we should take". But if not fully-fledged, then arguably this is a transitional example. The stereotypical movement can be analysed either as (i) meaningless dynamics that nevertheless results in coordinated behaviour; or (ii) as a symbolic gesture from one agent to others. These two interpretations are not contradictory, they are framed at different levels of description. I suggest that these sorts of minimal cognition experiments are stepping stones on the way to evolving artificial agents where we can be more confident in calling them users of representations.

In Summary

There are both scientific and philosophical questions on the concepts of representations in cognitive science. Even the scientific questions cannot escape the philosophical issues of just what one might mean by the term. I am suggesting here, following Wittgenstein, that most of the philosophical problems and confusions come from poor linguistic hygiene. They are not issues of substance at all, merely the consequence of carelessness. On one interpretation of philosophy, I have not gone to any great lengths to argue for or against any particular philosophical position on representations. On another interpretation, this insistence on linguistic hygiene, to try to help people escape from the messes and confusions they make for themselves, is actually what philosophy is all about. Recall the difference between "pavement" and "pavement".

The common lack of care in defining terms may possibly, I have suggested, been partly because of the billiard balls role of representations in cognitive science; representations are so often treated as *explanans* rather than *explanandum*. It has always been irritating when those of us who share my opinion that we have no internal representations (as I understand the common sense of the term) in our heads have been branded as "Anti-Representationalists". I insist on calling myself a Representationalist, since I consider our human use of representations to be immensely important and interesting, and deservedly a focus of interest for Artificial Life studies. Representations as *explanandum*, not *explanans*.

This paper has been centred on the philosophical confusions, but there are proper scientific and technical questions to be asked. What are the minimal requirements for artificial agents to be capable of being representation-users? Understanding of this would seem to be a pre-requisite to the discussion of what are the minimal requirements for a sub-part (or module, or homunculus) within a brain or nervous system to be capable of being (metaphorically) representation-users in communicating with other modules. Artificial life methods have already made a tentative start to exploring these

questions, and we can hope for further progress. Artificial Life models give a superb arena in which these tricky, potentially ambiguous, terms can be given a demonstrably explicit sense, open to operational testing.

References

Beer, R.D. (1995). A dynamical systems perspective on agent-environment interaction. *Artificial Intelligence* **72**:173-215.

Beer, R.D. (2000). Dynamical approaches to cognitive science. *Trends in Cognitive Sciences* **4**(3):91-99.

Clark, A. and Grush, R. (1999). Towards a cognitive robotics. *Adaptive Behavior* 7 (1): 5-16.

Cliff, D., P. Husbands and I. Harvey (1993). Evolving visually guided robots. In: J.-A. Meyer, H. Roitblat and S. Wilson (eds.), *From Animals to Animats 2: Proc. of the Second Intl. Conf. on Simulation of Adaptive Behavior*, (SAB92), pp. 374--383. MIT Press/Bradford Books, Cambridge MA.

Descartes, R. (1662). *De Homini*

Di Paolo, E. A. (2000). Behavioral coordination, structural congruence and entrainment in a simulation of acoustically coupled agents. *Adaptive Behavior* 8:1. 25 - 46.

Gallagher, S. (forthcoming). Are minimal representations still representations? *International Journal of Philosophical Studies*.

Grush, R. (1997). The Architecture of Representation. *Philosophical Psychology* 10(1):5-25.

Grush, R. (2004). The emulation theory of representation: motor control, imagery and perception. *Behavioral and Brain Sciences* 27:377-442.

Harvey, I., P. Husbands and D. Cliff (1993). Issues in evolutionary robotics. In: J.-A. Meyer, H. Roitblat and S. Wilson (eds.), *From Animals to Animats 2: Proc. of the Second Intl. Conf. on Simulation of Adaptive Behavior*, (SAB92), pp. 364--373. MIT Press/Bradford Books, Cambridge MA.

Harvey, I. (1996): Untimed and misrepresented: connectionism and the computer metaphor. *AISB Quarterly*, no. 96, pp. 20--27.

Harvey, I., P. Husbands, D. Cliff, A. Thompson, N. Jakobi (1997): Evolutionary Robotics: the Sussex Approach. In *Robotics and Autonomous Systems,* v. 20 pp. 205--224.

Harvey, I., Di Paolo, E., Wood, R., Quinn, M, and E. A., Tuci, (2005). Evolutionary Robotics: A new scientific tool for studying Cognition. *Artificial Life*, 11(1-2), pp. 79-98

Maynard Smith, J., and Szathmary, E. (1995). *The Major Transitions in Evolution.* Oxford University Press.

Quinn, M., Smith, L., Mayley, G. and Husbands, P. (2002). Evolving teamwork and role allocation for real robots. In Standish, R.K., Bedau, M.A. and Abbass, H.A., editors, *Proceedings of 8th International Conference on Artificial Life*, pages 302-311.

Quinn, M., Smith, L., Mayley, G. and Husbands, P. (2003). Evolving controllers for a homogeneous system of physical robots: Structured cooperation with minimal sensors. *Philosophical Transactions of the Royal Society of London, Series A: Mathematical, Physical and Engineering Sciences*, 361:2321-2344.

Rowlands, M. (2006). *Body Language*. Cambridge, MA: MIT Press.

Wheeler, M. (2005). *Reconstructing the cognitive world: The next step*. Cambridge, MA:MIT Press.

NKα: Non-uniform epistatic interactions in an extended NK model

Tom Hebbron[1], Seth Bullock[1] and Dave Cliff[2]

[1] School of Electronics and Computer Science
University of Southampton
Southampton, UK
[2] Department of Computer Science
University of Bristol
Bristol, UK
tomhebbron@zepler.net

Abstract

Kauffman's seminal NK model was introduced to relate the properties of fitness landscapes to the extent and nature of epistasis between genes. The original model considered genomes in which the fitness contribution of each of N genes was influenced by the value of K other genes located either at random or from the immediately neighbouring loci on the genome. Both schemes ensure that (on average) every gene is as influential as any other. More recently, the epistatic connectivity between genes in natural genomes has begun to be mapped. The topologies of these genetic networks are neither random nor regular, but exhibit interesting structural properties. The model presented here extends the NK model to consider epistatic network topologies derived from a preferential attachment scheme which tends to ensure that some genes are more influential than others. We explore the consequences of this topology for the properties of the associated fitness landscapes.

Introduction

Recent advances in our understanding of natural genomes are beginning to reveal patterns in genomic organisation (Jeong et al., 2000; Barabási and Oltvai, 2004; Segrè et al., 2004). In particular, the epistatic networks that describe the manner in which genetically specified proteins interact with each other during cell metabolism have been shown to exhibit topologies that are scale-free in their degree distribution (Maslov and Sneppen, 2002; Fernández, 2007). In such networks, while the vast majority of proteins are involved in only a small number of protein-protein interactions, a few proteins are highly influential (Barabási et al., 1999).

Here, we explore the influence of this type of epistatic network topology on the structure of associated fitness landscapes using an extension of the NK model originally proposed by Kauffman (1989). In the canonical form of this model, the fitness associated with a particular genotype (i.e., the height associated with a particular point on the fitness landscape) is assessed by combining the fitness contributions of the binary alleles at each of its N loci. The fitness contribution of a locus, i, is determined by the allele at i and the alleles present at K additional loci. For each unique combination of $K+1$ alleles, a unique, but randomly determined fitness contribution is assigned. By considering the statistical properties of ensembles of NK landscapes, the generic influence of epistasis can be assessed.

Kauffman was able to demonstrate that the 'ruggedness' of a landscape increases with increasing K. For $K = 0$ landscapes, each locus contributes to fitness independently. The landscape is smooth, with the fitness of adjacent genotypes being highly correlated as a consequence of sharing $N - 1$ fitness components. An adaptive walk originating at any point on such a landscape will reach a single, unique optimum. Every step on such a walk will reduce the distance to the optimum by one as the allele at one locus mutates to a fitter variant. The mean length of such a walk is therefore $\frac{N}{2}$. By contrast, for landscapes where $K = N - 1$ a mutation at any locus has the side effect of changing the fitness contribution of the alleles at all other loci. Consequently, there is no correlation between the fitness of neighbouring genotypes, and the landscape is maximally rugged. A large proportion ($\frac{1}{N+1}$) of genotypes are now local optima, and adaptive walks tend to stall after $ln(N - 1)$ steps. Intermediate values of K give rise to intermediate levels of ruggedness, altering the average distance between local optima, the correlation amongst locally optimal genotypes, and the fitness distribution of local optima. For more details, see Altenberg (1997).

In the two most frequently explored forms of the model, the K loci that epistatically influence a particular locus, i, may either be randomly located on the genome, or may be the K nearest neighbour loci of i. In both cases, every gene influences the fitness contribution of (on average) the same number of other genes, ensuring that genes are equipotent in their contribution to genotypic epistasis. Many variants of this model have been considered, and its behaviour has been explored in various ways (Altenberg, 1997; Barnett, 1998; Geard et al., 2002; Gao and Culberson, 2002; Campos et al., 2002; Rivkin and Siggelkow, 2002; Verel et al., 2003; Skellett et al., 2005; Kaul and Jacobson, 2007).

Kauffman himself mentions briefly a variant of the model in which some genes are more influential than others (Kauffman, 1989, pp78). Here, we develop this idea and explore the implications of systematically manipulating the extent to which there is a particular scale-free non-uniformity in the degree of influence exerted by each gene on the fitness contribution of the remainder of the genome.

Scale-free degree distributions have been discovered to characterise connectivity in a wide variety of systems, from gene regulatory networks to scientific citation networks (Barabási et al., 1999; Rzhetsky and Gomez, 2001; Gisiger, 2001; Wolf et al., 2002; Barabási et al., 2002; Barabási, 2003). In each case, the frequency with which network nodes exhibit degree k is proportional to $k^{-\gamma}$, where $\gamma > 1$ (Barabási and Crandall, 2003). Scale-free networks of this kind may be grown via a process of 'preferential attachment' (Barabási et al., 1999; Newman, 2001; Caldarelli et al., 2002; Eisenberg and Levanon, 2003). Under such a scheme, nodes are added sequentially to an initial small graph. Upon being added to the graph, each node is allocated a number of edges linking it to existing nodes, where the probability of adding an edge to an existing node of degree k is proportional to k^{α}. Here, α is a model parameter governing the strength of preferential attachment.

Networks with a scale-free topology have some distinct properties.

Self similarity at different scales: properties of local areas of the network are echoed in the whole.

The small-world phenomenon: shortest paths between any pair of nodes are remarkably short (Watts and Strogatz, 1998; Albert et al., 1999; Lazer and Friedman, 2005; Giacobini et al., 2006).

Robust to random failure: removal of nodes at random has little effect on network structure. However they are vulnerable to attacks that target the highly connected hubs (Albert et al., 2000; Barabási, 2003; Barabási and Crandall, 2003).

This paper first specifies an extended NK model, NKα, and describes the metrics that will be used to characterise its fitness landscapes. Results from the novel model are then compared with those of the canonical NK models, and their implications discussed before, finally, some future work is suggested.

Methods

An extensible NK model was implemented using a variation on the hashing method described by Altenberg (Altenberg, 1994, 1997), and using an efficient hashing algorithm proven against funnelling effects (Jenkins, 1997). The model was validated against published data from several sources for the Kauffman local and random variants (Kauffman, 1989; Weinberger, 1991; Kauffman, 1993, 1995; Altenberg, 1997). The random number generator and hashing functions were tested using the NIST validation suite (Rukhin et al., 2001).

The network of epistatic interactions between loci was represented as an $N \times N$ Boolean matrix, A, with $A_{ij} = 1$ iff locus i influences the fitness contribution of locus j. Since each locus always contributes to its own fitness contribution, $A_{ii} = 1 \; \forall i$. Furthermore, $\sum_i A_{ij} = K + 1 \; \forall j$, since each row of A contains K entries in addition to the self-connection, corresponding to j's incoming edges. By contrast, the sum of each column of A corresponds to the out-degree of each locus, which, in general, may be free to vary such that $1 \leq \sum_j A_{ij} \leq N$. Under all schemes considered here $\sum_{i,j} A_{ij} = N(K + 1)$, i.e., the total number of edges in the network is conserved.

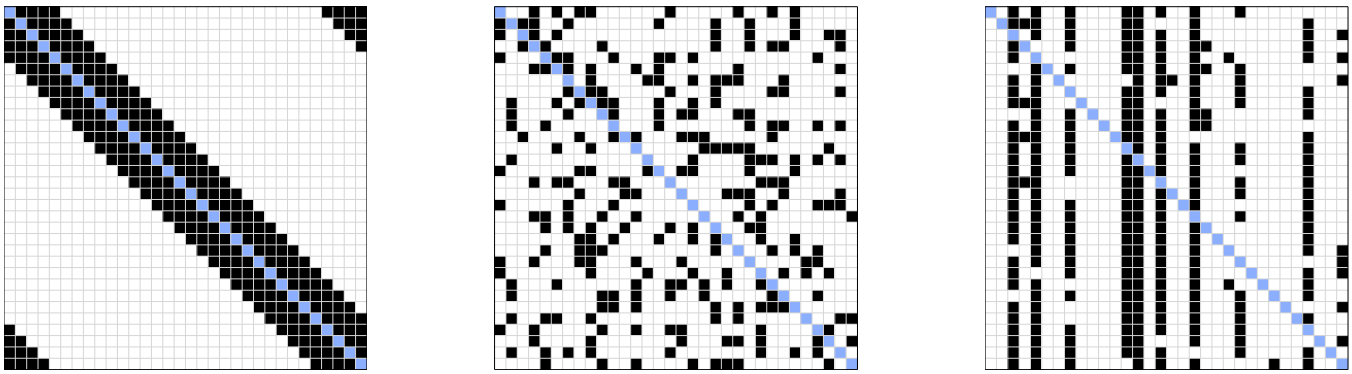
Kauffman's original NK model employed two schemes for allocating the epistatic links: local or random. In the former, each locus is influenced by its K nearest neighbours, giving rise to an epistatic network with a ring-lattice topology (see fig. 1a). In the latter, for each locus, K unique influential loci are chosen at random, giving rise to a random graph topology (see fig. 1b). Under the local scheme both in- and out-degree are uniform, whereas under the random scheme in-degree is uniform, but the out-degree is a Poisson distribution with a mean of K Newman et al. (2001).

Here, we introduce NKα, a variant of the NK model that employs a scale-free epistatic topology, parametrised by a single exponent, α. As before, the network contains $N(K + 1)$ edges, N of which are self-connections, and each locus has the same in-degree $(K + 1)$. However, the out-degree distribution approximates a power-law as a consequence of the following preferential attachment growth process.

Initially each locus is connected only to itself, giving a degree of 1. Subsequently, we perform K passes through the list of loci. Each pass visits each locus once in random order. On each visit, the visited locus is assigned one incoming edge from a random locus, i, chosen with probability $\propto (k_i)^{\alpha}$, where k_i is the out-degree of locus i and is updated after each visit, and the magnitude of α determines the strength of preferential attachment. This process assigns a total of $N(K+1)$ edges with a power-law like degree distribution, save that a ceiling threshold exists: no locus can have more than N connections (including its self-connection). With sufficiently high K or α, some loci will attract the maximum N connections deforming the power-law curve. When $\alpha = 0$, the resulting epistatic matrix is equivalent to the random map explored in the original model. Where $\alpha > 0$, increasingly skewed degree-distributions are generated, conferring increasing influence on a minority of loci (see Fig. 1c).

Measuring landscape properties

For the model introduced here, a triple (N, K, α) specifies an ensemble of landscapes that we sample and evaluate



(a) Local (b) Random (c) Scale-free: $\alpha = 1.8$

Figure 1: Epistatic maps for $N = 32$ with $K = 8$ using (a) local connectivity, (b) random connectivity, and (c) the NKα variant with $\alpha = 1.8$.

below. In addition to sampling the fitness distribution over each landscape as a whole by sampling 10,000 genomes at random on each landscape, we perform a number of walks across landscapes. Walks are of two types.

Adaptive walks were carried out by simple hill-climbers. At each step, a hill-climber calculates the fitness of all N single bit mutation neighbours of the current genotype, and selects one of the fitter neighbours at random to move to. If no fitter neighbour exists, then the hill-climber has reached a local optimum, and terminates. By undertaking multiple independent walks on the same landscape, an assay of available local optima can be compiled. Additionally, the length of adaptive walks is an indicator of a landscape's 'ruggedness'.

Random walks start from a random position in the landscape, and proceed by a series of random single-point mutations. Here, random walks were terminated after 2048 steps, as described by Weinberger (Kauffman, 1993). Such walks allow an assay of fitness distributions, and the correlation between the fitness of points separated by intervening genotypes.

Results

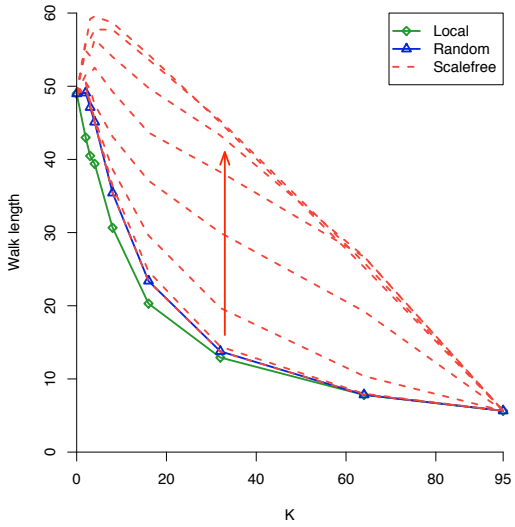
Unless otherwise stated, genotype length is held constant with $N = 96$, K ranges over $\{0, 1, 2, 3, 4, 8, 16, 32, 64, 81, 95\}$, and α ranges over $\{0.0, 0.5, 1.0, 1.5, 1.8, 2.0, 2.5\}$. By 'a full range of landscapes' we will mean all combinations of N and K for the local and random variants of the original NK model, and all combinations of N, K and α for the NKα variant. For the majority of results presented, the data is an aggregation of 10 repetitions of all combinations, each of these repetitions having a different seed and consequently different epistatic matrix.

Figure 2a shows the manner in which the average length of an adaptive walk varies with K for the landscapes considered here. While, in general, walk length increases with K, it is also apparent that walks tend to be longer for landscapes with higher values of α. Moreover, for low K and high α walks tend to involve a number of steps that exceeds $\frac{N}{2}$, the maximum average walk length observed for the local and random variants.

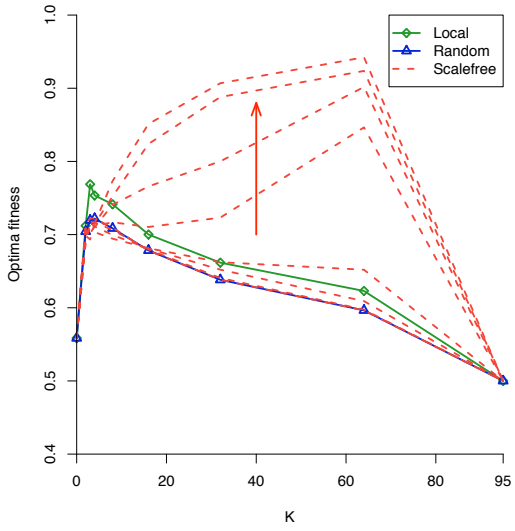
Figure 2b demonstrates that increasing α has a dramatic effect on the way in which K influences the mean fitness of landscape optima. For intermediate values of K, increasing α is associated with increasingly fit optima. In both figures we see that as α reaches high values, its influence asymptotes. This results from the ceiling effect mentioned above, which restricts the distribution of epistatic influences such that a few loci have the maximum influence, while the remainder have little or no influence at all.

For both the local and random variants of the NK model the correlation between the fitness at a local optimum and the fitness of its neighbours decreases with increasing K. Here, Figure 3 compares the distribution of fitness values of genotypes adjacent to a local optimum in a 'local' landscape with the distribution of fitness values adjacent to a local optimum on a landscape with relatively high α. The comparison is made for $N = 96$ and $K = 64$, but the qualitative results are characteristic of the comparison in general. For the local variant, fitness values adjacent to a local optimum are relatively tightly distributed around a value somewhat lower than the fitness at the local optimum. For the NKα variant, however, the distribution of adjacent fitnesses is much broader with many values close to the fitness of the local optimum, and many values far from it.

Figure 4a demonstrates that, for low K, the random NK model variant exhibits optima with a range of basin sizes and that there is a weak correlation between the fitness of an optimum and the size of its basin of attraction. Increasing K destroys this correlation, rendering every optima essentially equally attainable regardless of fitness (Figure 4b). However, the NKα variant gives rise to optima with a variety of



(a) Adaptive walk length

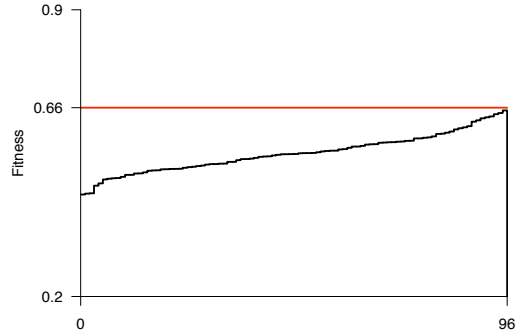


(b) Optima fitness

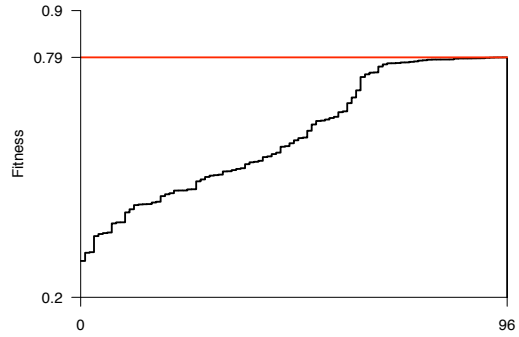
Figure 2: The mean length and mean local optimum fitness found for 100,000 adaptive walks for each (K,α) combination ($N = 96$). The arrow indicates increasing values of $0 \leq \alpha \leq 2.5$ for NKα variant.

basin sizes for high K landscapes, and here, optima fitness is strongly correlated with basin size. This accounts both for the fact that adaptive walks are taken on NKα landscapes tend to be longer than those carried out for equivalent landscapes from the local or random model variants, and that they tend to terminate at optima of higher fitness.

Kauffman used the term 'Massif Central' to describe a global structure he discovered in landscapes with small K. He used this term to refer to the tendency for high-fitness local optima to be located in the vicinity of the global optimum, rather than being randomly distributed as is the case for high K. The inverse correlation between fitness and distance in Figure 5a and its gradual erosion in Figures 5b and 5c reflect this observation. However, when we consider the NKα variant, we find a similar but stronger relationship with many fit optima close to the global optimum. Unlike for the original model variants, for the NKα variant this relationship between optima fitness and hamming distance is *strengthened* by increasing K.



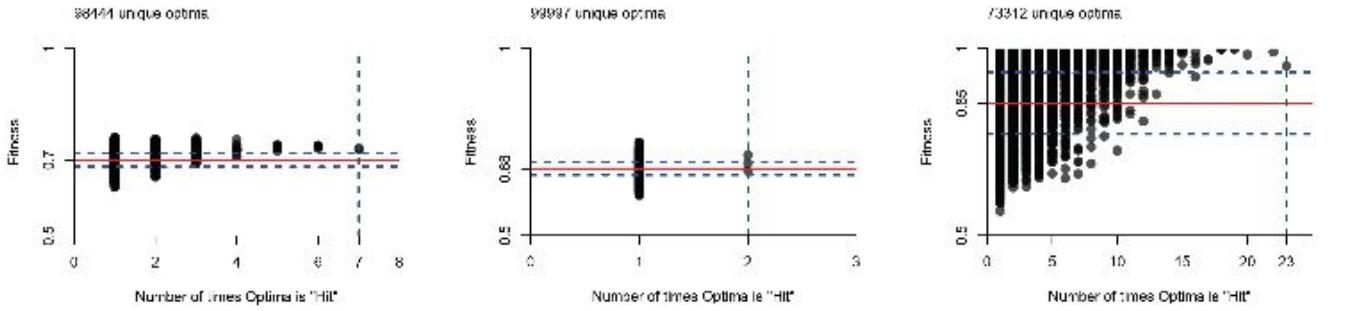
(a) Local



(b) Scale-free $\alpha = 1.8$

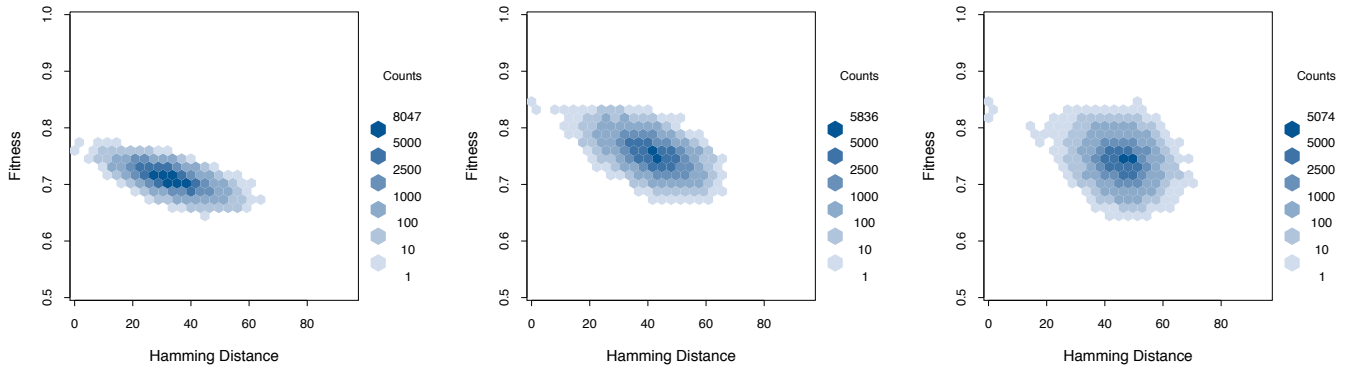
Figure 3: The fitness of neighbours of a local optimum are plotted for two landscapes ($N = 96$, $K = 64$), ordered by fitness. Solid horizontal lines indicate the local optimum fitness.

For the local and random variants of the NK model, loci are (roughly) epistatically equipotent. However, in the NKα model, some loci are more influential than others. How does this affect the rate at which different loci are mutated during an adaptive walk? At each step of an adaptive walk, plotting the out-degree of the mutated gene (how many loci it influences epistatically) reveals that for $\alpha > 0$, the most influential loci become fixed early in the adaptive walk. This effect increases with increasing α. For fixed α and N, increasing K increases the number of influential loci, and (if $\alpha > 0$) decreases the number of weakly connected loci. This lengthens the phase during which influential nodes are 'locked in.' By contrast, in the random variant (a), the variation in out-degree is much reduced, walks tend to be shorter, and there is no relationship between the out-degree of a locus and its tendency to be mutated early or late in an adaptive walk.

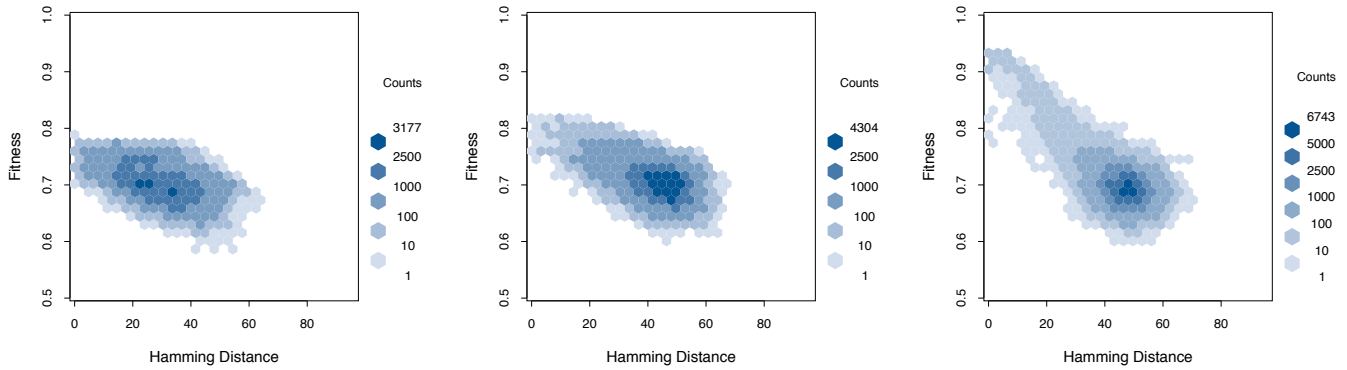


(a) Random: $K = 2$ (b) Random: $K = 16$ (c) Scale-free $K = 16, \alpha = 2.5$

Figure 4: The accessibility of local optima discovered by 10 repetitions of 10,000 independent adaptive walks on three classes of landscape. Solid horizontal lines indicates mean fitness with standard deviation indicated by dashed horizontal lines. Vertical dashed lines indicate the most frequently reached optima.

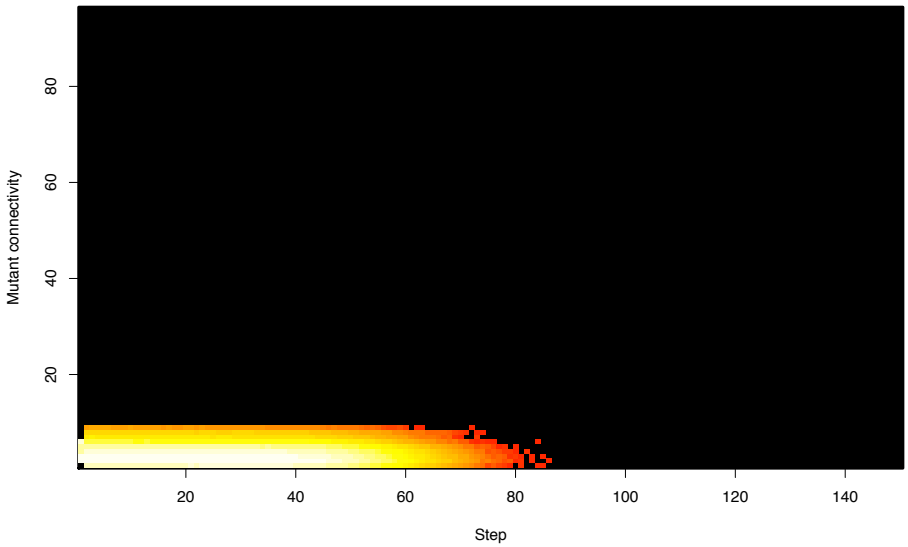


(a) Local: $K = 2$ (b) Local: $K = 4$ (c) Local: $K = 8$

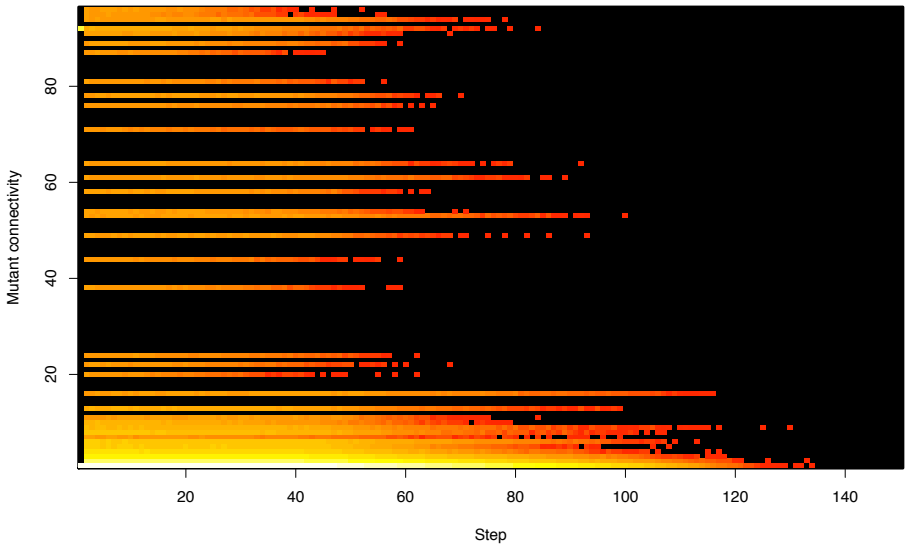


(d) Scale-free: $K = 2, \alpha = 1.5$ (e) Scale-free: $K = 4, \alpha = 1.5$ (f) Scale-free: $K = 8, \alpha = 1.5$

Figure 5: Each plot shows the hamming distance from the fittest optima found to the other optima found on 100,000 adaptive walks with $N = 96$ and $K = 2, 4, 8$. The data has been binned, and plotted with boundaries {0,1,100,1000,2500,5000,10000} to give a sense of the density of clustered optima. Darker tone indicates higher density of optima. The key on each plot shows only those bins used, with an upper limit of the most dense point in the plot. For the local variant (a-c) increasing K gradually erodes an inverse correlation between hamming distance and optima fitness. For the NKα variant, (d-f) increasing K strengthens a similar relationship.



(a) Random



(b) Scale-free $\alpha = 2.5$

Figure 6: When are influential loci mutated on an adaptive walk? For a sample of 100,000 adaptive walks, the heat map depicts the degree of mutated loci at each time step for (a) the random model variant, (b) the NKα variant. Here, $N = 96, K = 2$. The heat colour scheme is used: black indicates no mutation events, increasing to white indicating the most frequent mutation events.

Discussion

In general, imposing an increasingly scale-free structure on the network of epistatic interactions brings about a number of significant changes to the behaviour of adaptive walks on the associated fitness landscape: longer adaptive walks, higher fitness optima, more clustering of optima in the landscape and increased correlation between their fitness and the distance between them.

When the K epistatic influences of a locus are uniformly distributed, the resultant landscape is essentially isotropic. Statistical properties in one part of the landscape are largely predictive of the whole. Consequently, the effect of increasing K is to impose ruggedness globally. Conversely, when the same number of epistatic interactions are allocated non-uniformly, the genome is structured such that there exist a few influential loci and a majority of loci with little or no influence. This structure gives rise to a radically anisotropic landscape. Portions of the landscape exhibit properties that are very different from one another. More specifically, fixing alleles at influential loci confines an adaptive walk to a relatively correlated sub-landscape, while fixing the same number of low-influence loci confines an adaptive walk to a much less correlated landscape. Adaptive walks on such landscapes tend to initially spend time fixing influential loci, since mutating these alleles can bring about significant fitness changes. Once a satisfactory configuration of highly influential loci is discovered, low influence loci can be fixed relatively easily, since each is essentially independent from the others.

As yet it is unclear the extent to which one might describe NKα landscapes as modular. Are there multiple Massif Centrals on these landscapes, each characterised by a cluster of local optima of similar fitness? Or is there a more gross organisation of optima across the landscape as a whole? Relatedly, we have not considered assortativity in the network of epistatic interactions. While it has been known for some time that, for instance, the network of protein-protein interactions for yeast exhibits a scale-free degree distribution, recent work has shown that although the network for ancestral yeast has high degree proteins tending to interact directly with one another, the network for contemporary yeast is less assortative, with what has been interpreted as a more modular structure. For instance, precursors to modern yeast feature an epistatic network with a single hub related to the ribosome, whereas the modern yeast network exhibits two hubs, one ribosomal and the other related to signalling. These hubs are connected, but only via other poorly connected proteins, making the whole network appear modular (Fernández, 2007). Scale-free network topologies tend to be robust to failure unless the hubs are targetted (Albert et al., 2000; Barabási, 2003; Barabási and Crandall, 2003; Jeong et al., 2001), and a modular topology has the advantage of preventing the failure of one hub triggering the failure of another.

The preferential attachment algorithm used here defines an epistatic network topology with a scale-free out degree, which has significant effects on the resulting fitness landscape. However, the class of networks with scale-free degree distribution encompasses a range of topologies. In future work, we will extend the current NKα variant to consider the influence of assortative epistatic network topologies; the fitness landscapes and evolutionary dynamics to which they give rise.

Conclusions

The human genome project revealed a far lower number of genes than anticipated, increasing the significance of the study of their interactions. By extending an existing model, the paper demonstrates how a scale-free epistatic network topology alters the properties of a fitness landscape in a way that makes adaptive dynamics on it much more liable to discover high-fitness optima despite strong epistasis. To the best of our knowledge, and also to our surprise, this is the first systematic study of how the standard NK results vary when a preferential attachment scheme is used for determining the epistatic linkages between loci.

References

Albert, R., Jeong, H., and Barabási, A. (1999). The diameter of the world wide web. *Arxiv preprint cond-mat/9907038*.

Albert, R., Jeong, H., and Barabási, A. (2000). Error and attack tolerance of complex networks. *Nature*, 406(6794):378–382.

Altenberg, L. (1994). Evolving better representations through selective genome growth. In *Proceedings of the First IEEE Conference on Evolutionary Computation, IEEE World Congress on Computational Intelligence*, pages 182–187. IEEE Press.

Altenberg, L. (1997). NK fitness landscapes. In Bäck, T., Fogel, D. B., and Michalewicz, Z., editors, *Handbook of Evolutionary Computation*, pages B2.7.1–B2.7.10. Institute of Physics Press and Oxford University Press, New York.

Barabási, A. (2003). *Linked: How Everything Is Connected to Everything Else and What It Means for Business, Science, and Everyday Life*. Plume.

Barabási, A., Albert, R., and Jeong, H. (1999). Mean-field theory for scale-free random networks. *Physica A*, 272(1):173–187.

Barabási, A. and Crandall, R. (2003). Linked: The new science of networks. *American Journal of Physics*, 71:409.

Barabási, A., Jeong, H., Néda, Z., Ravasz, E., Schubert, A., and Vicsek, T. (2002). Evolution of the social network of scientific collaborations. *Physica A*, 311(3-4):590–614.

Barabási, A. and Oltvai, Z. (2004). Network biology: Understanding the cell's functional organization. *Nature Reviews Genetics*, 5(2):101–113.

Barnett, L. (1998). Ruggedness and neutrality: The NKp family of fitness landscapes. In Adami, C., Belew, R., Kitano, H., and Taylor, C., editors, *Artificial Life VI: Proceedings of the*

Sixth International Conference on Artificial Life, pages 18–27. MIT Press, Cambridge, MA.

Caldarelli, G., Capocci, A., De Los Rios, P., and Munoz, M. (2002). Scale-free networks without growth or preferential attachment: Good get richer. *Arxiv preprint cond-mat/0207366*.

Campos, P., Adami, C., and Wilke, C. (2002). Optimal adaptive performance and delocalization in NK fitness landscapes. *Physica A*, 304(3-4):495–506.

Eisenberg, E. and Levanon, E. (2003). Preferential attachment in the protein network evolution. *Physical Review Letters*, 91(13):138701.

Fernández, A. (2007). Molecular basis for evolving modularity in the yeast protein interaction network. *PLoS Comput Biol*, 3(11):e226.

Gao, Y. and Culberson, J. C. (2002). An analysis of phase transition in NK landscapes. *Journal of Artificial Intelligence Research*, 17:309–332.

Geard, N., Wiles, J., Hallinan, J., Tonkes, B., and Skellett, B. (2002). A comparison of neutral landscapes—NK, NKp and NKq. In Fogel, D. B., El-Sharkawi, M. A., Yao, X., Greenwood, G., Iba, H., Marrow, P., and Shackleton, M., editors, *Proceedings of the 2002 Congress on Evolutionary Computation*, pages 205–210. IEEE Press.

Giacobini, M., Preuss, M., and Tomassini, M. (2006). Effects of scale-free and small-world topologies on binary coded self-adaptive CEA. In *Evolutionary Computation in Combinatorial Optimization*, pages 86–98. Springer, Berlin/Heidelberg.

Gisiger, T. (2001). Scale invariance in biology: Coincidence or footprint of a universal mechanism? *Biological Reviews*, 76(02):161–209.

Jenkins, B. (1997). Hash functions. *Dr. Dobbs Journal*, 9709.

Jeong, H., Mason, S., Barabási, A., Oltvai, Z., et al. (2001). Lethality and centrality in protein networks. *Nature*, 411(6833):41–42.

Jeong, H., Tombor, B., Albert, R., Oltvai, Z., Barabási, A., et al. (2000). The large-scale organization of metabolic networks. *Nature*, 407(6804):651–654.

Kauffman, S. (1989). Adaptation on rugged fitness landscapes. In Stein, D., editor, *Lectures in the Sciences of Complexity*, volume 1, pages 527–618. Addison-Wesley, Redwood City, CA.

Kauffman, S. A. (1993). *The Origins of Order: Self-organization and Selection in Evolution*. Oxford University Press, Oxford.

Kauffman, S. A. (1995). *At Home in the Universe: The Search for Laws of Self-organization and Complexity*. Oxford University Press, Oxford.

Kaul, H. and Jacobson, S. (2007). New global optima results for the Kauffman NK model: Handling dependency. *Mathematical Programming*, 108(2):475–494.

Lazer, D. and Friedman, A. (2005). The parable of the hare and the tortoise: Small worlds, diversity, and system performance. KSG Working Paper No. RWP05-058, John F. Kennedy School of Government, Harvard University.

Maslov, S. and Sneppen, K. (2002). Specificity and stability in topology of protein networks. *Science*, 296(5569):910.

Newman, M. (2001). Clustering and preferential attachment in growing networks. *Physical Review E*, 64(2):25102.

Newman, M. E. J., Strogatz, S. H., and Watts, D. J. (2001). Random graphs with arbitrary degree distributions and their applications. *Phys. Rev. E*, 64(2):026118.

Rivkin, J. and Siggelkow, N. (2002). Organizational sticking points on NK landscapes. *Complexity*, 7(5):31–43.

Rukhin, A., Soto, J., Nechvatal, J., Smid, M., Barker, E., Leigh, S., Levenson, M., Vangel, M., Banks, D., Heckert, A., Dray, J., and Vo, S. (2001). A statistical test suite for the validation of random number generators and pseudo random number generators for cryptographic applications. Special Publication 800-22, NIST.

Rzhetsky, A. and Gomez, S. (2001). Birth of scale-free molecular networks and the number of distinct DNA and protein domains per genome. *Bioinformatics*, 17(10):988–996.

Segrè, D., DeLuna, A., Church, G., and Kishony, R. (2004). Modular epistasis in yeast metabolism. *Nature Genetics*, 37:77–83.

Skellett, B., Cairns, B., Geard, N., Tonkes, B., and Wiles, J. (2005). Maximally rugged NK landscapes contain the highest peaks. In Beyer, H. G., editor, *Proceedings of the Genetic and Evolutionary Computation Conference*, pages 579–584. ACM Press, New York, NY.

Verel, S., Collard, P., and Clergue, M. (2003). Where are bottlenecks in NK fitness landscapes? In Sarker, R., Reynolds, R., Abbass, H., Tan, K. C., McKay, B., Essam, D., and Gedeon, T., editors, *Proceedings of the 2003 Congress on Evolutionary Computation*, pages 273–280. IEEE Press.

Watts, D. and Strogatz, S. (1998). Collective dynamics of 'small-world' networks. *Nature*, 393(6684):409–10.

Weinberger, E. (1991). Local properties of Kauffman's NK model: A tunably rugged energy landscape. *Physical Review A*, 44(10):6399–6413.

Wolf, Y. I., Karev, G., and Koonin, E. V. (2002). Scale-free networks in biology: New insights into the fundamentals of evolution? *Bioessays*, 24(2):105–9.

Why Should You Care?
An Arousal-Based Model of Exploratory Behavior For Autonomous Robot

Antoine Hiolle and Lola Cañamero
Adaptive Systems Research Group
School of Computer Science
University of Hertfordshire
College Lane, Hatfield, Herts AL10 9AB, UK
{A.Hiolle,L.Canamero}@herts.ac.uk

Abstract

The question of how autonomous robots could be part of our everyday life is of a growing interest. We present here an experiment in which an autonomous robot explores its environment and tries to familiarize itself with the features available using a neural-network-based architecture. The lack of stability of its learning structures increases the arousal level of the robot, pushing the robot to look for comfort from its caretaker to reduce the arousal. In this paper, we studied how the behavior of the caretaker influences the course of the robot exploration and learning experience by providing certain amount of comfort during this exploration. We then draw some conclusions on how to use this architecture together with related work, to enhance the adaptability of autonomous robots development.

Introduction

The question of how autonomous robots could be part of our everyday life is of a growing interest. To approach this goal, many questions remain unanswered, from what kind of hardware would be needed, to what kind of architectures would be appropriate in order to promote socially situated robot that would fit in our environment. We are especially interested in the latter issue.

To design such an ideal robot, it is argued that taking an epigenetic approach would be a suited solution (Cañamero et al., 2006). Indeed, this approach would help the robot discover and learn affordances in the environment in which it is situated, including the agents it interacts with, as opposed to an approach where the designed architectures would need prior knowledge about the environment. The concern that arises with this approach is to find what sort of built-in mechanism a robot needs to be able to develop its cognitive and social capacities. To be precise, what are the inner drive(s) and basic principle(s) which will push the robot towards situations in which it will learn what it needs to in order to be fully operational in the given environment. This problem has many of similarities with the development of infants. Psychological evidence suggests that caretaker-infant attachment bonds are vital to the cognitive and emotional development of infants (Hofer, 2006), especially during the first years of life. Indeed, as John Bowlby (1969) discovered during his studies on mother-infant interactions, the primary caretaker, usually the mother, is utilized by the infant as a secure base in his/her early life, especially during stressful and/or unusual episodes. Furthermore, as stressed in (Schore, 2001), if the primary caretaker doesn't act accordingly to the infant's demands in term of interactions, the mental development of the child can be impaired, leading to emotional and cognitive disorders. Therefore, identifying the factors that are particularly relevant during these interactions, as well as their dynamics, is important to understand how the development of a child can lead to many different and uneven outcomes.

Our work also took inspiration from work done in the autonomous robotics research area, such as (Avila-Garcia and Cañamero, 2004), for affective (hormonal) modulation of behavior selection in the case of action selection in a competitive scenario; and especially (Blanchard and Cañamero, 2006; Cañamero et al., 2006), modeling the caretaker in the case of a perception used to modulate the robot's affect and thus its behavior. Drawing on these ideas, we have developed a robotic architecture to explore a new environment and learn from it using the robot's caretaker as a secure base, i.e. providing "comfort" to reduce the robot's distress. Numerous scenarios in terms of caretaking style are then possible to try to enhance the robot's experience and especially its learning process.

In the remainder of this paper we introduce an experiment illustrating how a caretaker can help to modulate the arousal of an infant-like robot by interacting with it and providing it with comfort. The architecture used here allows the robot to discover and learn information about its environment, more specifically getting used to meeting certain patterns of stimuli and classifying them in a stable manner. During that exploration, its arousal is stimulated by the novelty and the lack of stability of the patterns it senses. When this arousal level is high, the robot looks for comfort from the caretaker. The arousal thus modulates the behavior of the robot, and the caretaker modulates its arousal.

Robotics Model

Our architecture can be described in three main steps. The robot first learns the features encountered in its exploration of the environment, and gets habituated to them and classifies them. Then the convergence and stability of these structures are evaluated to calculate the arousal level; this arousal level reflects the degree of surprise and mastery of the robot in the current sensorimotor situation. Finally, an appropriate action is selected and executed.

Exploring and Classifying the Environment

To explore and categorize the environment, our architecture uses two different learning systems. First, a Hopfield-like associative memory neural network is used to learn the patterns of stimuli encountered during the experiment. The system is based on models of associative memory (Davey and Adams, 2004). The network is a two-dimensional grid of N binary neurons, with a state or output S_i, locally connected to their four nearest neighbors and randomly connected to four other units of the network with a symmetric connection matrix of weights w_{ij}. The connectivity is a blend of the two configurations represented in Fig. 1. This model is a modification of the standard Hopfield network. The local field h_i of a unit *i* is given by:

$$h_i = \sum_{i \neq j}^{N} w_{ij} S_j$$

then the next state of the unit i is calculated as:

$$S_i = \begin{cases} 1 & \text{if } h_i > 0 \\ -1 & \text{if } h_i < 0 \\ 0 & \text{if } h_i = \theta_i \end{cases}$$

In our network we use asynchronous random-order updates. Then to learn the presented input pattern vector, we use a modified version of the following procedure from (Davey and Adams, 2004):

Begin with a zero-weight matrix

Repeat either until all local fields are correct or for M time steps

Set the state of the network to one of the input patterns ξ

For each unit i *in turn*

Calculate $h_i \xi_i$

If this is less than a threshold T, then change the weights between unit i and all

other connected units j, according to:

$$\forall j \neq i \quad w'_{ij} = w_{ij} + \frac{\xi_i \xi_j}{N}$$

The point in which our algorithm differs from the original (Davey and Adams, 2004) is the repetition until all local fields are correct. In our experiment the number of steps used to learn the current pattern is fixed (10 steps in the current settings). Therefore, the pattern is learned correctly and completely if the robot stays in its current position, in front of the exact sensory input pattern; if all the local fields are correct before ten time steps, then learning stops as described above.

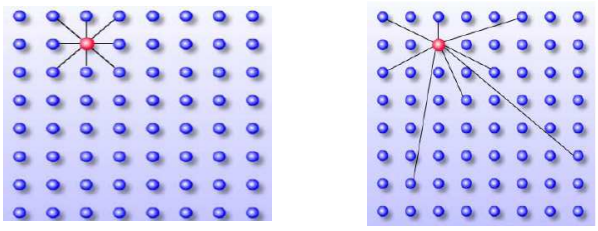


Figure 1: Associative memory network connectivity (locally connected on the left and randomly connected on the right, from (Calcraft et al., 2007))

The second learning algorithm is a classical Kohonen Self-Organizing map (Kohonen, 1997). The goal of this module is to classify the patterns of stimuli encountered during exploration. We used the classical algorithm, but here we don't have a decreasing learning rate or neighborhood size over time; therefore, the map is constantly learning but has nevertheless a satisfying stability for already encountered patterns and keeps its plasticity.

Arousal Model

To compute the arousal of the robot we use two different contributions. First, we evaluate the discrepancy between

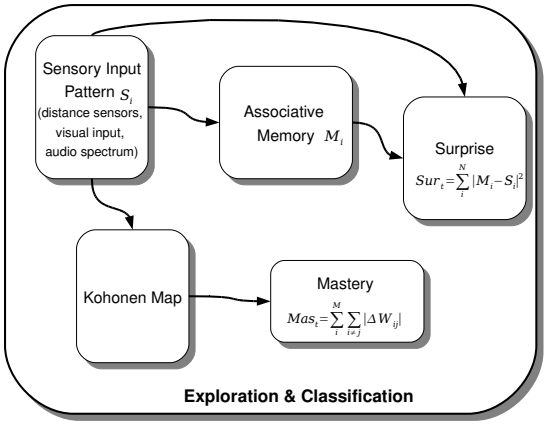


Figure 2: The robot explores and classifies the environment using a Hopfield-like associative memory and a Kohonen Map.

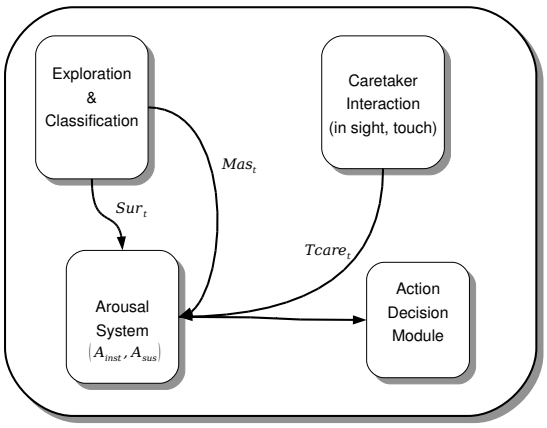


Figure 3: Entire Architecture



Figure 4: Our Experimental Setup

the current pattern of stimuli and the output of the associative memory, a value we call surprise Sur_t, since it decreases as a function of the familiarity of the current pattern of stimuli. Indeed, since the associative memory has a fixed number of time steps to learn the pattern, more than one presentation is needed. When a pattern is familiar enough, the network converges fast and the surprise value is close to zero. We also use Mas_t, a value we call *Mastery*, which is the sum of the variations of the weights of the Kohonen map. This value shows the ability of the robot to classify the current pattern and how these classes evolve. Formulas of how these values are calculated are displayed in Fig. 2. At each time step, the arousal of the robot is computed as:

$$A_t = \begin{cases} \frac{Sur_t + Mas_t}{2} & \text{if } T_{Care} = 0 \\ A(t-1) - \alpha \cdot T_{Care} & \text{otherwise} \end{cases}$$

where T_{Care} takes the value 0.5 when the caretaker is in sight, 0.8 when he/she touches the back sensors, 1 when both conditions are met, and 0 otherwise. Here α is the decay rate of the instantaneous arousal when the caretaker is interacting (set to 0.2). $A(t)$ is then used to evaluate a smoothed value of the arousal that we call *instantaneous arousal*, as follows:

$$A_{inst}(t) = \frac{\tau_a \cdot A_{inst}(t-1) + A(t)}{\tau_a + 1}$$

This value allows us to calculate an average of this arousal, called *sustained arousal*,

$$A_{sus}(t) = \begin{cases} \frac{\tau_{sus} \cdot A_{sus}(t-1) + A_{inst}(t)}{\tau_{sus}+1} \\ \text{if } T_{Care} = 0 \text{ and } A_{inst}(t) > 0.4 \\ 0 \qquad \text{otherwise} \end{cases}$$

where $\tau_a = 30$ is the time window on which the instantaneous arousal is calculated, as an average of $A_{inst}(t)$, and $\tau_{sus} = 10$, the time window on which the sustained arousal is calculated, as an average of the instantaneous arousal.

Choice of Actions

The actions the robot takes are based on the levels of both, instantaneous and sustained arousal. The robot can turn in only one direction, to discover a new pattern of stimuli when the arousal is low and the robot is in a "bored state". If the arousal is neither low nor high the robot remains still and tries to learn the current pattern of stimuli. If the arousal level is high, the robot barks to attract the caretaker's attention, and if the arousal is high and sustained, the robot looks for the caretaker by moving is head from top to bottom and left to right, trying to attract the caretaker in sight. Numerically speaking, the actions described above are taken when the conditions below are met:

$$\begin{cases} \text{if } A_{inst} < 0.25 & \Rightarrow \text{turn to explore} \\ \text{if } A_{inst} > 0.25 \text{ and } A_{inst} < 0.7 & \Rightarrow \text{stay still and learn} \\ \text{if } A_{inst} > 0.7 & \Rightarrow \text{bark to attract attention} \\ \text{if } A_{inst} > 0.7 \text{ and } A_{sus} > 0.6 & \Rightarrow \text{search for the caretaker} \end{cases}$$

Experimental setup and Results

In our experiments we used an Aibo robot on play mat, adding three cylindrical objects of different colors, as shown in Fig. 4. The robot uses three sensory modalities: color (the main color in the center of its visual field projected into the RGB color space), distance (the distance measurements provided by three distance sensors located in front of the robot), and contact (from one contact sensor on the top of its head and three on its back). Each sensor value (including the 3 RGB components of the color of the centre of its visual field) is discretized and projected into a vector containing ten binary elements. To summarize, the robot has to habituate to a vector aggregating all the element of the sensory space, i.e. 100 binary elements (30 for the color, 30 for the distance sensors, 30 for the back sensors, and 10 for the head sensor). The caretaker can provide comfort to the robot either by appearing in its visual field and staying in sight or by touching the sensors on its back. The robot recognizes the caretaker

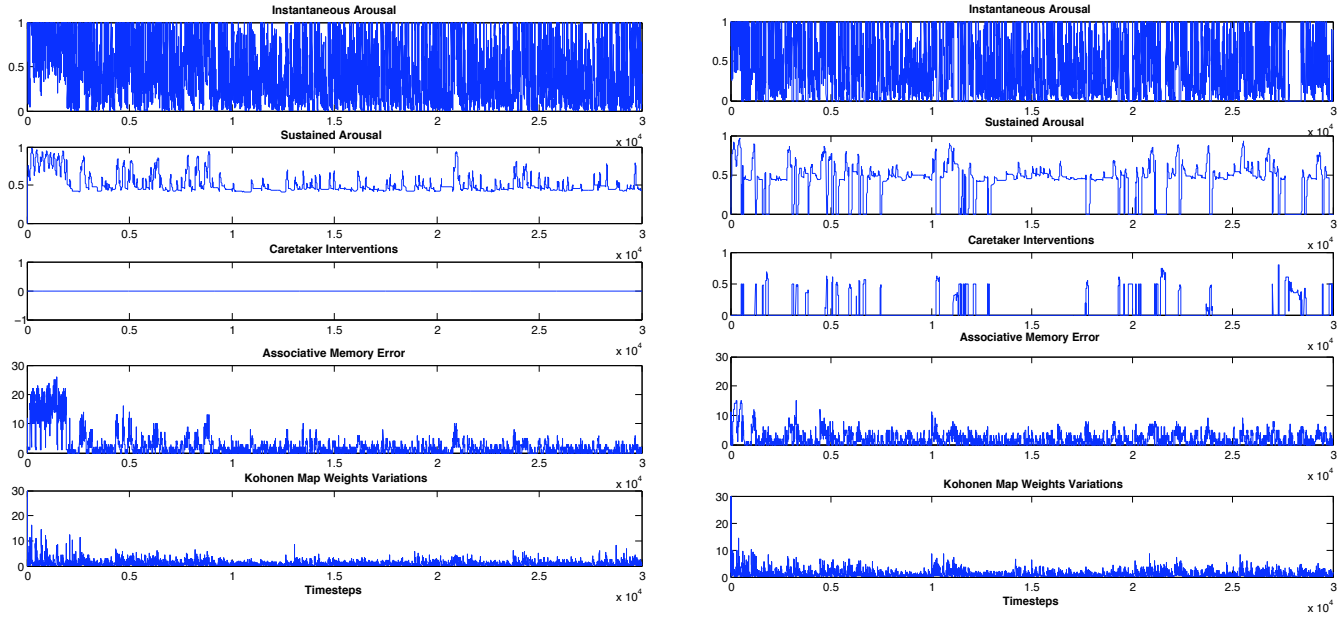


Figure 5: Evolution of (from top to bottom): instantaneous arousal, sustained arousal, caretaker interventions, associative memory error, and variations of the Kohonen map's weights. The graphs on the left-hand side correspond to an experiment with a caretaker only available at the beginning of the experiment whereas the ones on the right-hand side correspond to an active caretaker often providing comfort to the robot.

using the color of its clothes (this is hardcoded in this experiment, the caretaker is wearing a black top as it is the only color absent from the experiment room). At every time step, we recorded the values described in the model section, namely instantaneous arousal, sustained arousal, caretaker interventions, associative memory error, and variations of the Kohonen map's weights.

We have represented the results of two typical experiments in Fig. 5 with two different caretaking styles: an active caretaker, responding almost constantly to the robots demands (results on the right-hand side of the figure), always staying on the right of the robot to appear in sight every time the robot is looking for him/her, and a caretaker who only interacts at the beginning and then leaves the robot on its own and only intervenes few times (once every two minutes). The beginnings of both experiments are the same. When the robot is put on the play mat, it is almost instantly asking for the caretaker, since all the features are new and highly stimulating its arousal. Then the caretaker appears in sight and touches its back sensors to calm it down. We can observe on the graphs that for both caretaking styles, the Mastery value and Surprise value are high and sustained in the case of the non-caring caretaker, since the "non-caring" caretaker then backs away immediately after putting the robot down. Whereas for the other type of caretaking, the experimenter stays close during the whole experiment. In the case of the non-intervening caretaker, the robot is surprised and quickly stimulated by the new environment, and the levels of arousal (sustained and instantaneous) urge it to look for the caretaker quickly. By doing this, the robot actually sees the colors of the upper environment, which are novel stimuli, and tries to learn them, and this results in an even higher increase of its arousal levels. As for the experiment with an active caretaker, since he interacts and provides comfort, the arousal levels are lower and the robot can explore without . To find out how the two different caretaking styles differ, in terms of stability and performance of the exploration and classification system, we ran our experiment 10 times for each of the scenarios. The results for the average values and standard deviations for Mastery (Kohonen Map weights variations), Surprise (associative memory error)and Sustained arousal for the entire experiment are presented in Table 1. These values are used as a measurement of the quality of the learning process, to evaluate how each caretaking style

Style	$\bar{Mas}\sigma$	(Mas)	$\bar{Sur}\sigma$	(Sur)
Caring	0.5987	0.0355	0.3456	0.0565
Not Caring	0.6427	0.0407	0.6455	0.0324

Table 1: Results for 10 runs for each caretaking style

affect the learning experience of the robot. Each run lasted 50,000 timesteps and started from the exact same position. We can see that in terms of the Kohonen Map stability (the Mastery value), the caring caretaker behavior does not outperform the non-caring one by a large difference. However, there is a large difference in terms of Surprise (the associative memory's performance) between the different caretaking styles. The sustained arousal gives coherent results since the robot without the caretaker has to deal on its own to reduce its arousal by mastering the situation and getting habituated to the patterns. We can only conclude with this small sample that both behaviors are not optimal and that finding the correct trade-off between staying close and not caring needs further investigations. As an end result, in all our runs the robot had learned and classified all the encountered patterns meaning therefore its arousal always remained under the lowest threshold and kept turning fast in the arena in the "bored state", looking for new features to learn.

Discussion

The architecture we used in our experiment allows a robot to explore an unknown environment as a function of the dynamics of its interactions with the caretaker and the behavior of this latter. We have seen that even using such a simple architecture, the outcomes of every experiment are different depending on the type of interactions. The developmental approach we have followed reproduces mother-infant interactions. However, what needs to be underlined is the difficulty experienced during tuning the parameters of the architecture, namely the decay rates of the arousal levels. Indeed, to obtain a behavior oscillating between exploring, learning, and demanding the caretaker's presence, we needed to explore several configurations of the parameters. Nevertheless, these results show how using the caretaker as an arousal — and indirectly as a behavioral— modulator is actually possible without having a complex architecture. Furthermore, apart from these two opposite caretaking styles, our architecture allows to actively choose whether a situation, pattern of stimuli, has to be learned or avoided. Indeed, if the caretaker wants the robot to really learn the pattern, he/she can provide a small amount of comfort for the robot to have its instantaneous arousal in the middle level, between the two thresholds. This way the robot remains in its current position, without looking for the caretaker or moving away. In the opposite case, the caretaker can provide comfort to the robot so that it continues to look for another situation, keeping the instantaneous arousal below the lowest threshold, so that the robot does not learn one situation that is judged irrelevant by the caretaker.

As for the related work, a comparable model of arousal modulation and mother-infant interaction, although, not applied to robotics, can be found in (Smith and Stevens, 1996, 2002). In these contributions, the authors used a similar approach to modulate arousal based on neurophysiological data (Hofer and Sullivan, 2001) regarding how endogenous opioids modulate arousal in infants. However, their architecture did not have any cognitive system related to the interactions and their qualities, but was focused on the dynamics of the dyadic interaction. Another contribution can be related to this work. In (Likhachev and Arkin, 2000), the notion of comfort and object of attachment is used by a robot to remember its "comfort zones". What differs between the work presented here is that the object is a person, and also the comfort of the robot is not a function of the distance between the robot and the object of attachment.

Finally, in (Thomaz and Breazeal, 2007), an interesting experiment is described showing how a human can help a robot learn a certain task. In this contribution, a robot can explore and learn on its own but has also the opportunity to use human guidance to adapt to new tasks, changes in the environment, and to generalize one task to similar ones. The robot communicates its internal state with basic facial expressions and gestures. This "Socially Guided Exploration" presents similar features with the work presented here; in both experiments the interactions with a human are used to enhance the learning process, and also in both cases the human teacher/caretaker has to pay attention to the feedback from the robot in order to intervene to help and guide the robot. However, what differs between the two experiments is the modalities the human uses to interact with the robot. In the experiment presented in this paper, the human caretaker orients the robot's behavior by touching its back sensor to reduce its arousal level in order for the robot to move to another sensorimotor context, or appear in sight, whereas in the contribution discussed here, the human teacher can either point with his/her finger to a certain region of the environment or even give verbal instructions to the robot. We argue that the simple non-verbal way of interacting we used in our experiment is sufficient to bias the behavior and improve the learning process of an autonomous robot.

Conclusion and Future Work

In the experiments described above, we have shown how it is possible to modulate the exploratory behavior of an autonomous robot using notions like surprise and mastery to take into account its cognitive development, and especially using a caretaker as a secure base to provide comfort and reduce its arousal. To provide a more autonomous and adaptive solution, we could use material from previous work, modeling the imprinting phenomenon, using a perception or a compound of them as "desired perceptions" (Blanchard and Cañamero, 2005; Hiolle et al., 2007). These perceptions could be the voice of the caretaker and his/her face. We could then to add to our architecture the possibility for the robot to learn how to attract the attention of the caretaker and keep him/her close enough, as has been done in (Hiolle and Cañamero, 2007). However, finding the correct parameters for the architecture to obtain a balanced be-

havior was not easy and we experienced what was stressed in (Kaplan, 2001):"Fixing the satiation level and the speed of decay in order to obtain the right behavior remains the tricky thing". We think that using even earlier experiences of the robot could help evaluate these parameters. Using this as grounding for an early shaping of the personality of the robot would help us build a more realistic robot, and assess its attachment style using an Ainsworth-like Strange Situation Test (Ainsworth, 1969). To improve the autonomy of our robot's development, adding a curiosity drive (Oudeyer et al., 2007) would guide the robot's exploration towards more interesting situation, acting in order to increase its "learning progress". Another possibility would be to modify our architecture using the arousal, or a variable related to it (first derivative for instance), to directly modulate the cognitive abilities of the robot. More precisely, this value could modulate the learning rate, and/or the neighborhood of the Kohonen map. The robot could then exhibit various behaviors depending on the situation, and the dynamics of the system would certainly be different, perhaps closer to what happens with infants. On another level, what also needs to be done is to come up with accurate and consistent metrics to qualify and even quantify the behavior of the caretaker. We would also like to measure how a caretaker is interacting and possibly assess the effects of the different caretaking styles. We could then even point out what definitely should not be done based on the behavior and personality of the robot. We would also like to investigate how a robot could develop bonds with several caretakers and exhibit preferences for a given caretaker as a function of the given context or situation.

Acknowledgements

We are grateful to Neil Davey and Rod Adams for discussions about their neural networks model of associative memory. This research is supported by the European Commission as part of the FEELIX GROWING project (http://www.feelix-growing.org) under contract FP6 IST-045169. The views expressed in this paper are those of the authors, and not necessarily those of the consortium.

References

Ainsworth, M. D. S. (1969). Object relations, dependency, and attachment: A theoretical review of the infant-mother relationship. *Child Development*, 40(4):969–1025.

Avila-Garcia, O. and Cañamero, L. (2004). Using hormonal feedback to modulate action selection in a competitive scenario. In Schaal, S., Ijspeert, J., Billard, A., Vijayakumar, S., Hallam, J., and Meyer, J. A., editors, *From Animals to Animats 8: Proceedings of the 8th Intl. Conf. on Simulation of Adaptive Behavior*, pages 243–252. Bradford Book.

Blanchard, A. and Cañamero, L. (2006). Modulation of exploratory behavior for adaptation to the context. In Kovacs, T. and J., M., editors, *Biologically Inspired Robotics (Biro-net) in AISB'06: Adaptation in Artificial and Biological Systems.*, volume II, pages 131–139.

Blanchard, A. and Cañamero, L. (2005). From imprinting to adaptation: Building a history of affective interaction. *Proc. of the 5th Intl. Wksp. on Epigenetic Robotics*, pages 23–30.

Bowlby, J. (1969). *Attachment and loss*, volume 1:Attachment. New York : Basics Books.

Cañamero, L., Blanchard, A., and Nadel, J. (2006). Attachment bonds for human-like robots. *International Journal of Humanoïd Robotics*.

Calcraft, L., Adams, R., and Davey, N. (2007). Optimal connection strategies in one- and two-dimensional associative memory models. In *Proceedings of the 7th Intl. Wksp. on Information Processing in Cells and Tissues, Oxford*.

Davey, N. and Adams, R. (2004). High capacity associative memories and connection constraints. *Connection Science*, 16(1):47–65.

Hiolle, A. and Cañamero, L. (2007). Developing sensorimotor associations through attachment bonds. In Prince, C., Balkenius, C., Berthouze, L., Kozima, H., and Littman, M., editors, *Proc. 7th Intl. Wksp. on Epigenetic Robotics*, pages 45–52. Lund University Cognitive Studies.

Hiolle, A., Cañamero, L., and Blanchard, A. (2007). Learning to interact with the caretaker: A developmental approach. In Paiva, A., Prada, R., and Picard, R., editors, *Proc. of the 2nd Intl. Conf. on Affective Computing and Intelligent Interactions*, pages 422–433.

Hofer, M. (2006). Psychobiological Roots of Early Attachment. *Current Directions in Psychological Science*, 15(2):84–88.

Hofer, M. and Sullivan, R. (2001). Toward a neurobiology of attachment. *Handbook of developmental cognitive neuroscience*, pages 599–616.

Kaplan, F. (2001). Artificial attachment: Will a robot ever pass ainsworth's strange situation test. In *Proceedings of Humanoids 2001 : IEEE-RAS Intl. Conf. on Humanoid Robots*, pages 125–132.

Kohonen, T. (1997). *Self-Organizating Maps*. Springer-Verlag.

Likhachev, M. and Arkin, R. (2000). Robotic comfort zones. In *SPIE Sensor Fusion and Decentralized Control in Robotic Systems III*, pages 27–41.

Oudeyer, P.-Y., Kaplan, F., and Hafner, V. (2007). Intrinsic motivation systems for autonomous mental development. *IEEE Transactions on Evolutionary Computation*, 2(11).

Schore, A. (2001). Effects of a secure attachment relationship on right brain development, affect regulation, and infant mental health. *Infant Mental Health Journal*, 22(1-2):7–66.

Smith, T. and Stevens, G. (1996). Emergence, Self-Organization, and Social Interaction: Arousal-Dependent Structure in Social Systems. *Sociological Theory*, 14(2):131–153.

Smith, T. and Stevens, G. (2002). Hyperstructures and the Biology of Interpersonal Dependence: Rethinking Reciprocity and Altruism. *Sociological Theory*, 20(1):106–130.

Thomaz, A. and Breazeal, C. (2007). Robot learning via socially guided exploration. In *Proceedings of the 6th Intl. Conf. on Development and Learning, London*.

The Effects of Payoff Preferences on Agent Tolerance

Enda Howley and Colm O'Riordan

Department of Information Technology
National University of Ireland, Galway
enda.howley@nuigalway.ie
colm.oriordan@nuigalway.ie

Abstract

An objective of multi-agent systems is to build robust intelligent systems capable of existing in complex environments. These environments are often open, noisy and subject to rapid, unpredictable changes. This paper will explore how agents can bias their interactions and choices in these complex environments. Existing research has investigated how agents can bias their interactions based on factors such as similarity, trust or reputation. Unfortunately, much of this research has ignored how agents are influenced by their preferences for certain game payoffs. This paper will show that individual payoff preferences have a significant effect on the behaviors that emerge within an agent environment. We argue that agents must not only determine with whom to interact, but also the levels of benefit or risk these interactions should represent. This paper presents a series of game theoretic simulations examining the effects of agent payoff preferences within an evolutionary setting. Our experiments show that these factors promote tolerance throughout the population. We provide an experimental benchmark using an almost identical game environment where payoffs are not considered by agents. Furthermore, we also present simulations involving noise, thereby demonstrating the ability of these more tolerant agents to cope with uncertainty in their environment.

Introduction

Agent interactions are often heavily biased through certain group structures. These structures are often defined by factors such as geographical location (Axelrod, 1984), kin selection (Hamilton, 1963), choice and refusal (Stanley et al., 1995), or trust and reputation (Dellarocas, 2003). In the real world, individuals often bias their interactions through factors such as their need for certain services, or preferences for particular goals. As a result, we must acknowledge that not all agent interactions are identical and are driven by individual preferences and needs. Therefore in this paper we will examine through game theoretic simulations how agent payoff preferences influence the overall agent population. Some researchers have examined the payoffs commonly used in the Prisoners Dilemma and concluded that certain payoffs promote cooperation (Fogel, 1993). The implications of agent payoff preferences when determining their peer interactions remain to be fully explored and understood.

This paper shows how game payoff preferences directly influence the levels of tolerance and reciprocity throughout an agent population. Existing research has not examined the significance of these agent payoff preferences. For example, in a multi-agent environment an agent may only trust one of its peers. Yet, in order to satisfy its individual preferences this agent may choose to interact with a less trusted peer. This decision could be based on a payoff preference or similarly a preference for a service being offered. In short, within a game theoretic model, agent interactions should reflect that agents are free to bias their interactions based on their preferred peers and also their preferred games. As a result we will simulate an environment where agents may offer and choose games based upon their preferred payoff values. Some agents are risk takers and prefer games which have higher risk payoffs, while others are more risk averse and prefer game payoffs which hold lower risk.

Previous research of this type has focused primarily on IPD games which remain constant throughout the population. Recent research has started to explore the effects of allowing the game payoffs to change (Taylor and Nowak, 2006; Howley and O'Riordan, 2006a, 2008). The need to study these interactions stems from the fact that real world interactions rarely remain identical indefinitely. In reality, an agents interactions will be determined through its bias towards preferred individuals and its bias towards achieving specific goals. This paper will investigate aspects of this statement and in particular address the following two research questions:

1. What are the effects of game payoff preferences on the overall agent population and the strategy genes?

2. What strategies are most successful in this variable payoff environment when noise is introduced?

In the following section of the paper we outline our motivations and aims. We will then discuss various aspects of background and related research from the subject domain. This will involve existing research in the areas of spatial, tagging and trust models. Subsequent sections will

describe our evolutionary algorithm, simulator design, parameters and experimental setup. The results section will present a series of experiments showing the behavior of the overall population when alternative interaction models are used. Stemming from these results we will outline our conclusions.

Aims and Motivations

The primary goal of this paper is to propose a simple extension to the traditional Iterated Prisoner's Dilemma and analyse the effects of this salient extension on agent interactions. Previous research from the authors has examined a number of agent interaction models. In previous work, we examined the underlying dynamics of tag mediated interaction models. We identified the significance of tag group size on the levels of cooperation that emerge within the agent population. We also examined the effects of openness on trust between agents (Howley and O'Riordan, 2006b). We found that alternative forms of openness and change effected agents in very different ways. We also identified the dependencies that can emerge between agents within a trust model when exposed to openness. More recent research has investigated payoff variances and preferences among agents (Howley and O'Riordan, 2008). We outline how agents like to avail of higher payoffs through defection but this results in the agent population exploiting each other into choosing games with low risk payoffs. These strategy preferences emerged to dominate the agent populations.

This paper examines the agent strategies at gene level and presents a more detailed examination of their preferences for certain game payoffs. Furthermore we hope to clarify the heightened levels of tolerance present in these game environments. We hope to ascertain the scale of these differences and the reasons behind them. Also we hope to test these levels of tolerance through introducing noise into the game interactions. This differs from simulating reduced population viscosity, or mutation which would serve to undermine cooperation in cooperative groups. In this work we are more interested in identifying the strategy traits that emerge when these factors are not present.

We have extended the Prisoner's Dilemma to reflect variable payoffs, thereby scientifically capturing the effects of agent payoff/risk preferences. Agents' who express their game preference based on a games 'temptation to defect' are in effect specifying a unique game with an associated degree of risk. We provide a more detailed description of this extension in later sections.

Through the results presented in this paper, we aim to extend our previous research while also demonstrating the important dimensional space which has largely been ignored by existing research in multi-agent systems. The differences shown in this paper present many implications for the domain of multi-agent research. The most important of these involves the need to delineate between agent environments where all interactions are of equal value and those where interactions are not equal.

Background Research

In this paper our main topic of concern involves how agents bias their interactions. Previous research on this topic has examined techniques such as spatial, tagging, kin selection and trust. In this section we will discuss some of the existing research on these topics. We will also introduce and discuss the Iterated Prisoner's Dilemma (IPD), which is used throughout our simulations.

Spatial, Tagging and Kin Selection

In relation to the emergence of cooperation, one of the most important considerations involves how agents bias their interactions towards cooperative peers and away from non-cooperative peers. In this paper we are only concerned with the latter. Kin selection is one such interaction mechanism involving groups of related individuals (Hamilton, 1963). Another more common interaction model involves agents located on a spatial topology such as a grid (Axelrod, 1984; Nowak and May, 1993). Agents bias their interactions and therefore play peers located on adjacent cells of the grid. Tag-mediated interaction models are based on a similar premise. These models locate agents on an abstract topology and then bias interactions based on players proximity to each other (Holland, 1993; Riolo, 1997). Arbitrary tags are similar to visible markings or labels which may be used by agents to bias their interactions based on their preferences. Some real world examples of tags could include football fans recognising each other from their jerseys or travelers recognising each other abroad through their native accents. Tags can provide a more general representation of agent interactions than spatial models. Later in this paper we will outline how tag-mediated selection may be used to structure interactions based on players individual preferences for certain games.

Iterated Prisoner's Dilemma

The Prisoner's Dilemma (PD) is a simple two player, non-zero sum, non-cooperative game. Each player must make a decision to either cooperate (C) or defect (D). Both players decide simultaneously and, therefore, have no prior knowledge of what the other has decided. If both players cooperate, they receive a specific payoff. If both defect, they receive a lower payoff. If one cooperates and the other defects then the defector receives the maximum payoff and the cooperator receives the minimum.

For this game to be classified as a dilemma in all cases, certain constraints must be adhered to. The following is the first constraint:

$$\lambda 2 < \lambda 4 < \lambda 1 < \lambda 3 \quad (1)$$

Table 1: Prisoner's Dilemma Payoff Matrix

Players Choice	Cooperate	Defect
Cooperate	$(\lambda 1, \lambda 1)$	$(\lambda 2, \lambda 3)$
Defect	$(\lambda 3, \lambda 2)$	$(\lambda 4, \lambda 4)$

These conditions result in $\lambda 2$ being the sucker's payoff, $\lambda 1$ is the reward for mutual cooperation, $\lambda 4$ is the punishment for mutual defection, and $\lambda 3$ provides the incentive or temptation to defect. The second constraint is the following:

$$2\lambda 1 > \lambda 2 + \lambda 3 \tag{2}$$

This constraint prevents players taking alternating turns receiving the sucker's payoff ($\lambda 2$) and the temptation to defect ($\lambda 3$), therefore maximising their score.

The following λ values are commonly used in the Prisoner's Dilemma: $\lambda 1 = 3, \lambda 2 = 0, \lambda 3 = 5, \lambda 4 = 1$.

In the non-iterated game, the rational choice is to defect, while in the finitely repeated game, it is rational to defect on the last move and by induction to defect all the time. This game has been used throughout numerous research domains, including economics, computer science and the social sciences. More detailed discussions on the Prisoner's Dilemma and its various guises are widely available (Axelrod, 1984; Hoffmann, 2000; Delahaye et al., 2000; Kendall et al., 2006).

Evolutionary Algorithm

The experimental results presented in this paper involve agent environments simulated over successive generations. In each of these generations an evolutionary algorithm is applied to reflect the real world pressures on under performing entities and alternatively reward the best performing ones. In this section we will outline in detail the evolutionary algorithm used throughout our simulations.

In the domain of game theory, one of the most common evolutionary techniques involves replicator dynamics. These quite general evolutionary models replicate changes in agent's fitness through increasing or decreasing their representation in successive generations. Therefore in a population of n species, each of which adopts a strategy i, the population state can be represented as the following vector at time step t (Generation t):

$$x^{\,t} = (x^{\,t_0}, ..., x^{\,t_n}) \tag{3}$$

As a result, x_i^t represents the fraction of the population which can be considered belonging to a species i.

$$(x^{\,t_i} \geq 0, \sum_{i=0}^{n} x^{\,t_i} = 1) \tag{4}$$

The game payoffs represented in the payoff matrix are used to determine payoff to individual species throughout their lifetime. Payoff to a species i is viewed as an indicator of fitness and thereby a measure of its reproductive success (Smith, 1982).

$$js_ij^t = js_ij^{t-1} \times \frac{f(s_i)^{t-1}}{\sum_{j=0}^{n} f(s_j)^{t-1}} \tag{5}$$

The representation of a species i in generation t is its representation in generation $t-1$, by the fitness it achieved in generation $t-1$, as a proportion of the average population fitness in generation $t-1$. Hence, the growth rate of an individual species i is proportional to its fitness.

Uncertainty and Noise

This paper presents a number of noise experiments which investigate the effects of uncertainty on the agent population. In previous research when we first examined payoff preferences among agents, we observed significant levels of intermittent defections (Howley and O'Riordan, 2008). Because of this we undertook this more detailed investigation of these population dynamics which appeared to promote forgiveness among agent strategies. In addition this paper also examines this phenomenon through simulating noise in the game environment.

Existing research has simulated uncertainty through a series of methods involving noise (Bendor et al., 1991) and reduced population viscosity (Howley and O'Riordan, 2006b). In this paper we have used noise as a means of simulating uncertainty. This will serve to test the ability of agents to forgive opponents. Previous research has shown that more forgiving and generous strategies perform best in noisy environments (Bendor et al., 1991).

Simulator Design

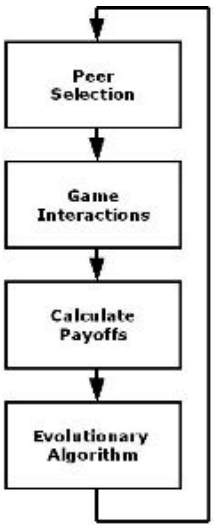


Figure 1: Game Cycle

In this section we outline our overall simulator design. We begin with an introduction to the game cycle (Figure. 1). We describe how we extended the Iterated Prisoners Dilemma (IPD) to allow agents express preferences for certain types of games. We also outline our strategy set.

Firstly, agents play their selected opponents. They enter IPD games using payoffs acceptable to their individual payoff preferences. The payoff preference of the offering agent us used and the opponent can then chose to interact or not. This decision is based on its own payoff preference. Once all games are played game payoffs are totaled and averaged. These are then used as measures of fitness for our evolutionary algorithm. This evolutionary algorithm uses replicator dynamics based on fitness to determine agent representation over successive generations. No crossover or mutation is applied.

Agents select their opponents probabilistically based on the proximity of their tags. This is done in a similar way to much previous research involving tag mediated interactions. Players with similar tags are far more likely to interact than other pairings where there may be some difference between players tags. This differs from common green beard dynamics, whereby by individuals identify each other through their beard colour and cooperate if their beards are a similar colour and defect if they are different (Dawkins, 1976). Our model limits peer interactions to individuals of a similar tag value, and makes no assumption about an individuals actions towards those of a specific tag value.

Strategy Set

In order to define a strategy set, we refer to existing research which uses three bit IPD strategies (Nowak and Sigmund, 1990). In our simulations each strategy genome includes four genes representing probabilities of cooperation in an initial move p_i, in response to a cooperation p_c and defection p_d. The final strategy gene p_t represents an individuals game payoff preference. Some strategies are be more inclined to prefer lower risk games while others will prefer higher risk games. This is similar to people who are often natural risk takers while others are more risk averse. As we will explain in the following section our game environment permits players to agree the 'temptation to defect' (TD) value in the Prisoner's Dilemma game. The resulting strategy genome looks like the following:

$$Genome = p_i, p_c, p_d, p_t \tag{6}$$

The Variable Payoff Prisoner's Dilemma

The extended IPD game remains similar to the original game described earlier. It remains a simple two player dilemma which is non-zero-sum, non-cooperative and played simultaneously. For this game to remain a Prisoner's Dilemma it must still remain within the constraints of the original game as mentioned earlier. This game differs in that the payoffs used in each game interaction are not always the same. The extended game uses the following adapted payoff matrix. In this game the $\lambda 1$, $\lambda 2$, $\lambda 4$ payoffs remain constant while in this extended game the value of TD is determined by the individual players involved in each game interaction.

Table 2: Adapted IPD Payoff Matrix

Players Choice	Cooperate	Defect
Cooperate	$(\lambda 1, \lambda 1)$	$(\lambda 2, TD)$
Defect	$(TD, \lambda 2)$	$(\lambda 4, \lambda 4)$

For this game to remain a valid IPD, then the value of TD must remain within the following range of values:

$$\lambda 1 < TD < 2 \times \lambda 1 \tag{7}$$

The IPD payoff values used throughout this research are as follows: $\lambda 1 = 5000$, $\lambda 2 = 0$, $\lambda 3 = TD$, $\lambda 4 = 1$. As stated above the value of TD must always remain within the following range: $\lambda 1 < TD < 2 \times \lambda 1$. These λ values provide an expressive range of possible TD values.

Our decision not to allow agents determine all game payoffs stems firstly from the need to maintain a valid Prisoner's Dilemma, and secondly that all interaction choices be based on a fair and equal footing. One can also argue that a Prisoner's Dilemma which allows the TD to change is identical to a bounded Prisoner's Dilemma where all payoffs are permitted to change but still remain bounded by an upper payoff limit. This is due to all the payoffs being interdependent and relative as specified by the PD constraints. Therefore by allowing the TD payoff to change, all the game payoffs change relative to each other.

Experimental Results

In this section, we present a series of simulations involving our multi-agent population. We present direct comparisons between a number of multi-agent environments when using fixed payoff games versus variable payoff games. Firstly we examine these differences under noiseless environmental conditions. Subsequently, we present this comparison using a noisy game environment, whereby agent actions are effected by a degree of noise which will demonstrate more clearly the emergence of tolerance in our variable payoff game simulations.

All the simulations outlined in this paper involve populations of 1000 agents. Each experiment is a aggregation of 50 experimental runs. Each game interaction lasts 20 iterations.

The first interaction model is a variable payoff model where agents agree a TD payoff depending on their respective p_t genes. As in a tag environment, players choose their peers based on their tag (p_t gene) similarity. In this model the p_t gene value reflects a players preferences for games of a certain value. A high p_t gene would result in a high TD

payoff game while a low p_t gene would result in a low TD payoff game. This results in players with similar p_t genes interacting in games with TD payoffs proportional to their p_t genes. To determine the probability of two individuals interacting we use a formula proposed in previous research on tag-mediated interactions (Riolo, 1997). The dissimilarity of two individuals (A and C) is defined as follows:

$$d_{A,C} = |Ap_t - Cp_t| \tag{8}$$

The second interaction model is almost exactly the same. This is a fixed payoff model which allows the players determine their peer interactions based on their p_t gene similarity. The one significant difference is that all games use the same fixed payoffs. $\lambda 1 = 5000, \lambda 2 = 0, \lambda 3 = 7500$ and $\lambda 4 = 1$.

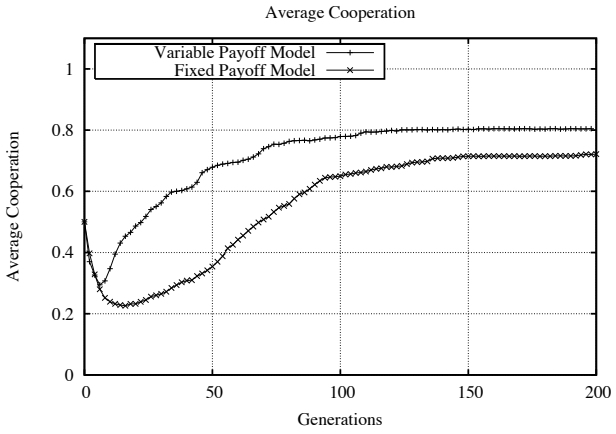


Figure 2: Average Cooperation.

We observe in the data shown in Fig. 2. the levels of cooperation attained using two agent models. These levels of cooperation were significantly higher in the variable payoff model. Furthermore these levels of heightened cooperation were also much faster to emerge. The table below presents a number of statistics reinforcing these observations. On average this model remained 0.10 greater than the static model. The data shown in Table 3 indicates the scale of the differences between the fixed and variable payoff environments. These differences were found to be statistically significant.

Model	μ	σ
Fixed	0.6102	0.1664
Variable	0.7103	0.1390

Table 3: Average Cooperation in Noiseless Environment

The following experiments show the average values for each strategy gene respectively. These values represent averages taken throughout the agent population at the start of each generation. From the results shown, we can ascertain the levels of reciprocity and tolerance present throughout the agent population. These can be identified from examining the p_c and p_d gene values respectively.

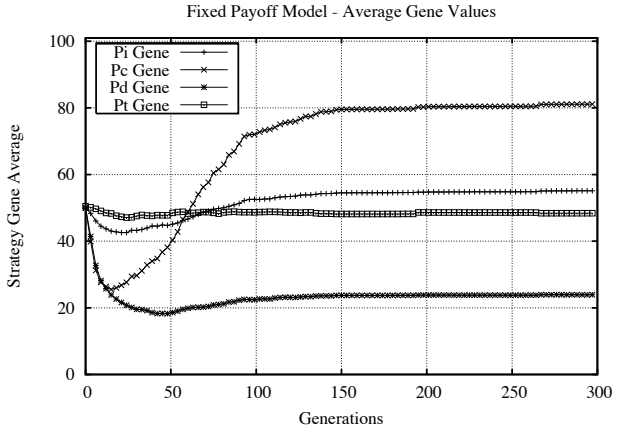


Figure 3: Fixed Payoff Game - Strategy Genes.

Fig. 3 shows average gene values recorded in the static payoff game environment. We observe how the p_t gene remains almost static. This gene experiences no evolutionary pressures as it serves simply as a tag for biasing interactions. The value of this gene is completely random from experiment to experiment. Therefore it's mean value across a large number of experiments always remains close to 0.5. The levels of cooperation in this model as shown in the first experiment (Fig. 2) can be attributed to the significant increase in the p_c gene from generation 20 onwards.

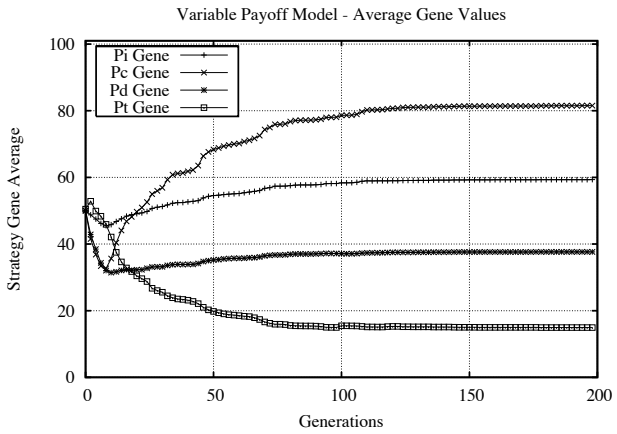


Figure 4: Variable Payoff Game - Strategy Genes.

The levels of cooperation identified in the variable payoff model in the initial experiment (Fig. 2.) are justified through the data shown in the in Fig. 4. This graph shows the p_d gene reaching levels that are significantly higher than in the static payoff model (Fig. 3.). As a result, agents are more likely to cooperate after a defection. This indicates a degree of tolerance or forgiveness which is far greater than in the

static payoff environment. Furthermore, the average p_c gene values indicate a similar likelihood of cooperation following an opponents cooperation. This characteristic emerges very rapidly in the initial generations of the variable payoff environment and is associated with a increased likelihood of mutual cooperation in an agent environment.

Summary The results shown here demonstrate the clear differences between static and variable payoff game environments. The heightened levels of cooperation identified in the variable payoff environment have been shown to emerge through specific genetic differences in that strategies that perform best in the respective game environments. These cooperation levels stem from the emergence of tolerance among the participant strategies. Defection in the static payoff model exerts no evolutionary pressure on the agent game preferences. Yet defection in the variable payoff model exerts pressure on agents to chose game interactions with lower TD game payoffs. This results in an agent population who are predominantly cooperative and also possess low p_t genes. Therefore any subsequent defections in this population incur few penalties and are tolerated throughout the agent population. These conclusions are confirmed through our analysis of the strategy genes in each game environment. The average p_t gene value initially increases as strategies choose high TD payoff games. They avail of these high TD payoffs by exploiting their peers, and this rapidly becomes more common throughout the population. Subsequently, these strategies start to mutually defect and begin to suffer. The p_t gene levels fall dramatically as cooperative groups emerge to dominate the population. Strategies who intermittently exploit to avail of certain TD payoffs thrive in this environment. These are the underlying reasons behind the increased tolerance and generosity throughout the variable payoff model.

Noisy Environments In the previous section we examined the similarities and differences between our static and variable payoff game environments. In order to more rigorously test our explanation of the differences between the two game environments, we will now examine their respective dynamics under noisy conditions. We represent noise as a probability that a move will be inverted from C to D or vice versa. The following experiments show the levels of cooperation recorded when alternative degrees of noise are simulated.

Fig. 5. shows the levels of cooperation involving simulations using fixed payoff games and alternative levels of environmental noise. From the simulations shown we can clearly see the effects of noise on levels of cooperation. As would be expected, 1% noise has a noticeable effect on cooperation while 5% has a much more dramatic effect throughout. These results show the extent to which strategies in the fixed payoff environment can cope with intermittent defections. High levels of tolerance would be very beneficial to individuals hoping to cope with intermittent defections.

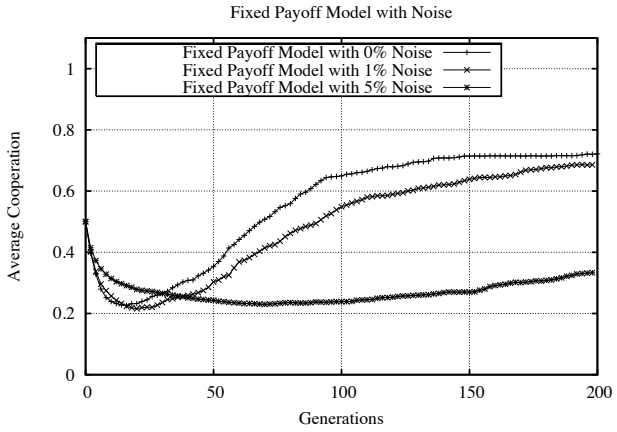


Figure 5: Average Cooperation

Model	Noise	μ	σ
Fixed	0%	0.6102	0.1664
Fixed	1%	0.4810	0.1661
Fixed	5%	0.2718	0.0390
Variable	0%	0.7103	0.1390
Variable	1%	0.6579	0.1396
Variable	5%	0.4924	0.1042

Table 4: Average Cooperation in Noisy Environments

Fig. 6. shows the effects of noise on levels of cooperation in a variable payoff environment. The main differences with the previous experiment in Fig. 5. are the levels of cooperation recorded for 5% noise. The strategies in the variable payoff environment appear to cope much better to these levels of noise. This is reinforces our earlier conclusions that these variable payoff environments promote tolerance throughout an agent population.

The final set of graphs show how the gene strategy values evolved within two game environments when 5% noise was introduced. The simulations shown in Fig. 7. represent the fixed payoff environment. These results show the same non convergence of the p_t gene, as its value carries no great significance in the fixed payoff model. The p_i and p_d genes fall in value while the p_c gene is the only gene which appears to recover in spite of the noise. The slow convergence of the p_c gene continues for about another 300 generations and reaches a level slightly below that identified in the noiseless experiment shown in Fig. 3. More significantly are the values of of the p_d gene which remain very low and indicate low levels of tolerance throughout the agent population. This contributes strongly to the levels of cooperation identified in Fig. 5. for this game environment using 5% noise. It is clear that any occurrences of intermittent defections as

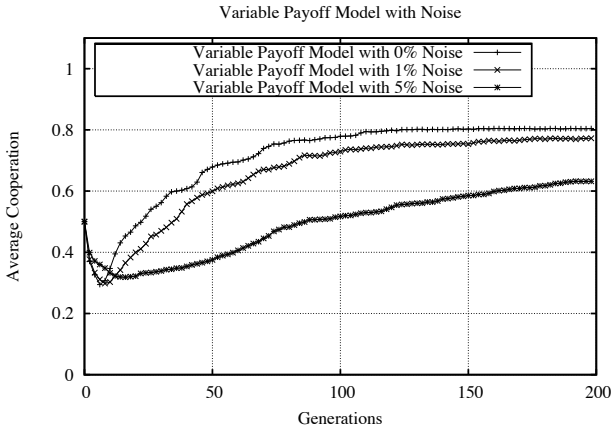


Figure 6: Average Cooperation

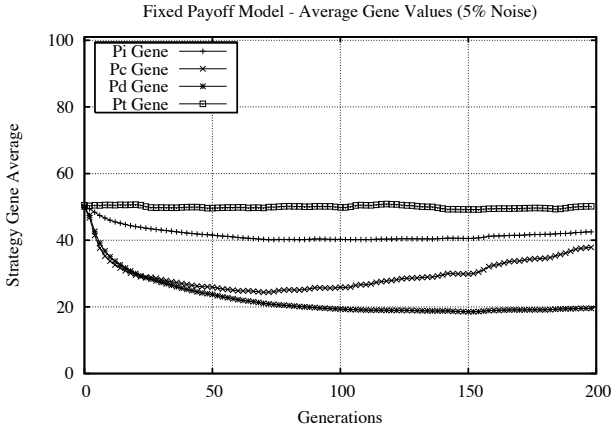


Figure 7: Average Cooperation

a result of noise will result in mutual defection between the participating individual agents.

Fig. 8. shows the gene values recorded in the variable payoff model with 5% noise. This experiment shows higher levels of tolerance as shown through the p_d gene values. Each of the strategy genes in this model continue to converge in a similar way to their noiseless counterpart presented in Fig. 4. but at a much slower rate. These indicate a much higher level of tolerance throughout the game environment and directly contribute to the heightened levels of cooperation identified previously. They explain the fundamental underlying dynamics which resulted to the significant results presented in Fig. 2.

Summary This section has examined a series of specific experimental results. These have shown the underlying differences between static payoff game interactions and the alternative agreed variable payoff model. These differences stem from the increased ability of strategies in the variable payoff model to forgive and tolerate intermittent defections.

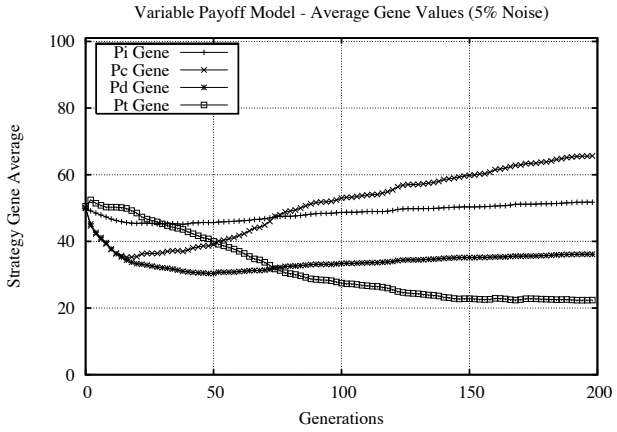


Figure 8: Average Cooperation

This is shown to be even more significant in a noisy environment where intermittent defections are more common. The statistical analysis shown in the tables indicate the significance of these differences. By demonstrating the each environment's ability to cope with intermittent defections, these noise experiments show the main factors that explain the differences first identified in Fig. 2. These noise experiments are not intended as an extensive examination of noise and its effects on IPD strategies. That would involve simulating many more levels and forms of noise.

Conclusions

This paper has presented two alternative game theoretic environments where agents play the Iterated Prisoner's Dilemma. Players bias their interactions through using a designated strategy gene. Through a series of experiments, we have shown that agent behavours can be fundamentally effected by the introduction of a variable payoff game. To date most research has been based on static payoff games and the resulting conclusions have been adopted and cited by many multi-agent researchers. We argue that variable payoff games provide a more realistic basis for real world agent interactions.

Initially we posed two fundamental research questions. Firstly, we queried the influence of variable payoffs on the agent environment. We have shown that this extension resulted in a significant increase in the numbers of strategies using higher value p_d genes. These resulted in greater levels of forgiveness throughout the agent population. We have also shown as previously (Howley and O'Riordan, 2008) that these agents favor lower value p_t genes. This indicates their preference for lower TD payoff games and reinforces the reasons why these strategies are more tolerant of defections. Reduced payoff rewards for defections would naturally make the agent more tolerant of such non cooperation. Initial exploitation for high TD games provides a significant advantage to strategies who only play games involving lower

risk games. These strategies then thrive and dominate the population.

Our second question queries which are the most successful strategies. From analysing the trends throughout all the experiments, it is clear that the strategies that are successful in fixed payoff environments are not successful in variable payoff environments and vice-versa. The most successful strategies in the fixed payoff environments are highly reciprocal and therefore not very tolerant. The most successful strategies in the variable payoff environment are highly tolerant and prefer low TD games. Once noise is introduced in the variable payoff environment, the strategies that are tolerant and encourage cooperation perform the best.

This paper has shown the fundamental differences between fixed and variable payoff environments from both a high level analysis and also a gene level examination. We have shown the intrinsic ability of variable game environments to encourage cooperation through tolerance. This leads to higher degrees of cooperation in both noiseless and noisy environments. These differences show the importance to future researchers, of differentiating between multi-agent environments where all agent interactions hold identical significance, and those which offer alternative rewards. This presents researchers with the possibility of encouraging tolerance throughout an agent population without making any assumptions about the agent population.

Acknowledgements

The primary author would like to acknowledge the support of the Irish Research Council for Science, Engineering and Technology for their support through the Embark Initiative.

References

Axelrod, R. (1984). *The Evolution of Cooperation*. Basic Books, New York.

Bendor, J., Kramer, R., and Stout, S. (1991). When in doubt cooperation in a noisy prisoner's dilemma. *Journal of Conflict Resolution*, 35:691–719.

Dawkins, R. (1976). *The Selfish Gene*. Oxford University Press, New York.

Delahaye, J. P., Mathieu, P., and Beaufils, B. (2000). *The Iterated Lift Dilemma: how to establish metacooperation with your opponent*. Springer-Verlag New York, Inc., Secaucus, NJ, USA.

Dellarocas, C. (2003). The digitization of word of mouth: Promise and challenges of online feedback mechanisms. *Manage. Sci.*, 49(10):1407–1424.

Fogel, D. B. (1993). Evolving behaviors in the iterated prisoner's dilemma. *Evol. Comput.*, 1(1):77–97.

Hamilton, W. (1963). The evolution of altruistic behaviour. *The American Naturalist*, 97:354–6.

Hoffmann, R. (2000). Twenty years on: The evolution of cooperation revisited. *The Journal of Artificial Societies and Social Simulation*, 3(2).

Holland, J. (1993). The effects of labels (tags) on social interactions. *Working Paper, Santa Fe Institute 93-10-064*.

Howley, E. and O'Riordan, C. (2006a). The effects and evolution of implicit trust in populations playing the iterated prisoner's dilemma. In *Proceedings of the 2006 Congress on Evolutionary Computation (IEEE CEC'06) held as part of the IEEE World Congress On Computational Intelligence (IEEE WCCI'06)*. IEEE Press.

Howley, E. and O'Riordan, C. (2006b). The effects of viscosity in choice and refusal ipd environments. *Artif. Intell. Rev.*, 26(1-2):103–114.

Howley, E. and O'Riordan, C. (2008). Agent interactions and implicit trust in ipd environments. In Tuyls, K., Nowe, A., Guessoum, Z., and Kudenko, D., editors, *Adaptive Agents and Multi-Agent Systems III*, volume 4865 of *Lecture Notes in Artificial Intelligence*, pages 87–101. Springer Verlag, Berlin.

Kendall, G., Yao, X., and Chong, S. Y., editors (2006). *The Iterated Prisoner's Dilemma: 20 Years On*. World Scientific.

Nowak, M. and May, R. (1993). The spatial dilemmas of evolution. *Int Journal of Bifurcation and Chaos*, 3:35–78.

Nowak, M. and Sigmund, K. (1990). The evolution of stochastic strategies in the prisoner's dilemma. *Acta Applicandae Mathematicae*, 20:247–265.

Riolo, R. (1997). The effects and evolution of tag-mediated selection of partners in populations playing the iterated prisoner's dilemma. In *ICGA*, pages 378–385.

Smith, M. (1982). *The Evolution and the Theory of Games*. Cambridge University Press, Cambridge.

Stanley, E. A., Ashlock, D., and Smucker, M. D. (1995). Iterated prisoner's dilemma with choice and refusal of partners: Evolutionary results. In *Proceedings of the Third European Conference on Advances in Artificial Life*, pages 490–502, London, UK. Springer-Verlag.

Taylor, C. and Nowak, M. (2006). Evolutionary game dynamics with non-uniform interaction rates. *Theor Popul Biol*, 69:243–252.

Analysis of a Dynamical Recurrent Neural Network Evolved for Two Qualitatively Different Tasks: Walking and Chemotaxis

Eduardo J. Izquierdo[1,2] and Thomas Buhrmann[2,3]

[1]Centre for Systems Biology, University of Birmingham, Edgbaston, Birmingham B15 2TT, UK
[2]Centre for Computational Neuroscience and Robotics, University of Sussex, Brighton BN1 9QH, UK
[3]Natural Motion Ltd, Oxford OX1 2ET, UK
eji21@sussex.ac.uk, tb30@sussex.ac.uk

Abstract

Living organisms perform a broad range of different behaviours during their lifetime. It is important that these be coordinated such as to perform the appropriate one at the right time. This paper extends previous work on evolving dynamical recurrent neural networks by synthesizing a single circuit that performs two qualitatively different behaviours: orientation to sensory stimuli and legged locomotion. We demonstrate that small fully interconnected networks can solve these two tasks without providing *a priori* structural modules, explicit neural learning mechanisms, or an external signal for when to switch between them. Dynamical systems analysis of the best-adapted circuit explains the agent's ability to switch between the two behaviours from the interactions of the circuit's neural dynamics, its body and environment.

Introduction

All organisms are equipped with a repertoire of distinct behaviours that allow for their survival and reproduction. The nematode worm *Caenorhabditis elegans*, for example, with 'only' 302 neurons shows a remarkable ability to perform a broad range of different behaviours (Hart, 2006; Rankin, 2004). Although our understanding of the neural basis for most of them is still at an early stage (de Bono and Maricq, 2005), it is known that overlap exists between some of the neural circuits responsible for these behaviours (Hobert, 2003). Also, it is well known that the morphology of living organisms is in constant change, both throughout evolution and during the lifetime of the organism. This work investigates how a single neural network that is not structurally divided into separate circuits can produce different behaviours in different bodies.

When modelling adaptive behaviours, assumptions have to be made with regard to the structure of the organism studied in order to simplify the modelling process or the analysis of the model's behaviour. One such assumption that has been made in the past is that functional modularity, the existence of several qualitatively different behaviours in the same organism or agent, should be mirrored by structural modularity in its neural controller. Complex systems are thus often divided into small parts that are synthesized in isolation. Such a divide-and-conquer approach can be very useful for engineering robots that need to perform multiple complex tasks, not least because it simplifies the understanding of how the robot works. But it is less useful in the context of developing the tools and language to understand biological organisms, as these may not necessarily have evolved to be easily decomposable.

First, we investigate whether a single neurocontroller can exhibit qualitatively different behaviours without imposing constraints on its structure. We use artificial evolution to synthesize a recurrent neural network that when coupled to two different simulated bodies, namely a one legged insect and a two-wheeled robot with a chemical sensor, has to perform legged locomotion in the former and chemotaxis in the latter case[1]. A successful agent has to detect which body it inhabits and generate the appropriate behaviour. It must do this in the absence of an external signal and without any online changes in the parameters of the controller. We aim to find the smallest network that can solve the task. Although the structure of the network is under evolution, we do not investigate whether the evolved networks exhibit a degree of structural or 'functional'[2] modularity.

Second, using the mathematical tools of dynamical systems theory, we explain how the circuit in interaction with its body and environment can generate distinct behaviours. We characterize the autonomous dynamics of the best-evolved circuit and how its dynamics vary with inputs. We then study how the observed behavioural patterns are generated through the closed-loop interaction of the neural dynamics with the body and environment, for the two different tasks. Finally, we show how the evolved agent makes use of context-dependent feedback to shape the different transients using the same dynamical landscape. This leads us to suggest a dynamical systems perspective on adaptive behaviour that goes beyond attractors.

[1]Both tasks have been studied in some depth in the Evolutionary Robotics literature. See Methods and Related work sections.

[2]Watson and Pollack (2005) argue that structural descriptions of the network are not sufficient to determine dependence or independence in the dynamics of different subsets of the network.

Methods

Walking task

The walking task employed follows very closely the simple one-legged body described and analysed in (Beer and Gallagher, 1992; Beer et al., 1999). Three variants of this model have been studied in (Beer, 1995), differing in whether sensory feedback is available constantly, only occasionally, or absent. Of the corresponding controllers, namely reflexive pattern generators (RPGs), central pattern generators (CPGs), and mixed pattern generators (MPGs), we focus on the first type only. The leg is controlled by three effectors: one specifies whether the foot has contact with the ground while the other two control clockwise and counter-clockwise torques for the leg's hinge-joint with the body (Figure 1A). The opposing torques model antagonistic muscles, commonly found in animal limbs. Sensory feedback is provided continuously by the leg's joint angle. At the beginning of each walking trial, the state of the leg (i.e. its angle with respect to the body) is initialised at random. The agent is then given 220 units of time to walk. The total distance covered during the trial measures performance.

Chemotaxis task

For the chemotaxis task we also follow a methodology similar to that employed in (Beer and Gallagher, 1992). A food patch, placed at arbitrary locations and orientations with respect to the agent, emits a chemical signal (s), whose intensity falls off as a function of the distance from the center of the patch (d): $s = e^{\lambda d}$, where λ is a constant, -0.0138. The agent can moves freely in an environment without walls and must find and remain in the vicinity of the food patch.

The agent has a circular body and possesses a sensor that can detect the intensity of the chemical signal at its location (Figure 1B). Additionally, it is equipped with two effectors located on opposite sides of its body. These effectors[3] can apply forces that move the body forward and rotate it. In the simplified physics of this environment, the velocity of movement is proportional to the force applied.

During a chemotaxis trial, a food patch is placed in a random direction from the agent, anywhere between 10 and 15 units of space apart. This is repeated after 100 units of time. Three food patches are shown in total. Performance is given by: $f_c = (d_i - \bar{d})/d_i$, where d_i and $\bar{d}$ are the initial and average Euclidean distance between the agent and the food patch, respectively.

Neural model

We use continuous-time recurrent neural networks as a model of the agent's internal dynamics. Each component

[3]Although the mechanics of the body correspond closer to a khepera-like robot, similar physics have been used in idealised models of the nematode worm's movement in (Ferree and Lockery, 1999). Instead of 'wheels', the effectors located on opposite sides model the ventral and dorsal neck muscles of the worm.

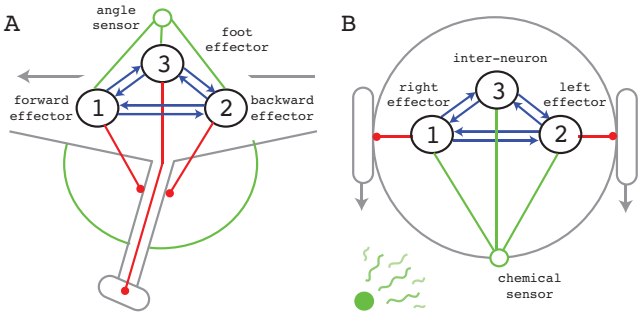


Figure 1: Task set-up. The same neural network circuit is used to control two different bodies: a one-legged insect-like walking agent (A) and a khepera-like chemotaxis agent (B). The circuits are fully inter-connected. The effector neurons control the antagonistic muscles and the foot for walking and the two effectors on the opposite sides of the body for chemotaxis. All of the neurons receive sensory perturbations: from the leg angle during walking, and from the proximity to the food during chemotaxis.

in the network is governed by the following state equation:

$$\tau_i \dot{y}_i = -y_i + \sum_{j=i}^{N} w_{ji}\sigma\left(y_j + \theta_j\right) + sw_i S(t) \qquad (1)$$

where y is the activation of each node; τ is its time constant; w_{ji} is the strength of the connection from the j^{th} to the i^{th} node; θ is a bias term; $\sigma(x) = 1/(1+e^{-x})$ is the standard logistic activation function; and N represents the number of nodes in the network. All nodes have access to the sensory perturbations via a set of connection weights: sw_i. The sensory input is normalised to run between 0 and 1 for both tasks. This prevents a solution that switches behaviour reactively as a response to different sensory input ranges. The network is fully connected (including self-connections) and no symmetry is imposed on its weight matrix. In simulation, node activations are calculated forward through time by straightforward time-slicing using Euler integration with a time-step of 0.1.

Evolutionary algorithm

The parameters of each circuit (i.e. biases, time-constants, inter-neuron and sensor-neuron weights for each node) are evolved using a version of the microbial genetic algorithm (Harvey, 2001). There are $N^2 + 3N$ parameters in total. These are encoded in a genotype as a vector of real numbers over the range [0, 1]. Offspring of microbial tournaments are generated as a mutation of the winner of the tournament (i.e. no recombination). The mutation is implemented as a random displacement on every gene drawn uniformly from a Gaussian distribution with mean 0 and variance 0.01. Each gene is forced to be in [0, 1]: when a mutation takes a gene out of this range it is reflected back. The

offspring replace the loser of the tournament. Genes are linearly mapped to network parameters in the range [-10, 10] for biases, inter-node and sensory weights and to the range [1, 20] for time constants. The size of the population used is 50 and we define a generation as the time it takes to generate the same number of new individuals. A minimal 1D wrap-around geography with demes of size 10 is used, such that only nearby individuals can compete in tournaments. Finally, because the fitness is noisy, agents are re-evaluated every time they participate in a tournament.

A successful circuit must maximize: (a) the distance walked when embodied in the insect-like body and (b) the time spent around the chemical-emitting food patch when in the khepera-like body. A fitness evaluation consists of 2 trials of the walking task and 15 trials of the chemotaxis task. At the start of each trial the state of the neural controller and the body is randomised. The performance on each task is averaged over all trials and normalised in the range [0, 1]. The fitness of an individual is calculated by multiplying the performance on both tasks.

Results

Evolutionary performance

Evolutionary searches with 3-, 4-, and 5-node circuits were performed. We examined the ability of populations to evolve for both tasks, and conducted control experiments in which either task was evolved on its own. For each condition, 20 evolutionary experiments with different initial random seeds were carried out.

First we compare networks of different size. Figure 2 shows the performance of the set of best circuits grouped according to size on the walking (2A) and chemotaxis task (2B). For each, two whisker plots show the performance of circuits on the task at hand. The grey whisker boxes correspond to populations evolved for only that task. White whisker boxes correspond to populations evolved on both tasks. Circuits of the same size perform better at generating the appropriate behaviour when evolved for only one task than those required to do both. This is true in all conditions and is expected. Also, the difference in performance is smaller for walking than for chemotaxis, which suggests it is the 'easier' task of the two. This is also as expected.

Nevertheless, sufficiently successful circuits that performed both tasks did evolve. However, is there a trade-off between walking performance and chemotaxis in the successful circuits? In other words, are some individuals good at walking but poor at chemotaxis and vice versa? In Figure 2C we show chemotaxis versus walking fitness for each of the best individuals from all evolutionary runs on the two tasks. No obvious tradeoffs are noticeable. Instead, most of the successful individuals at one task are also relatively successful at the other task.

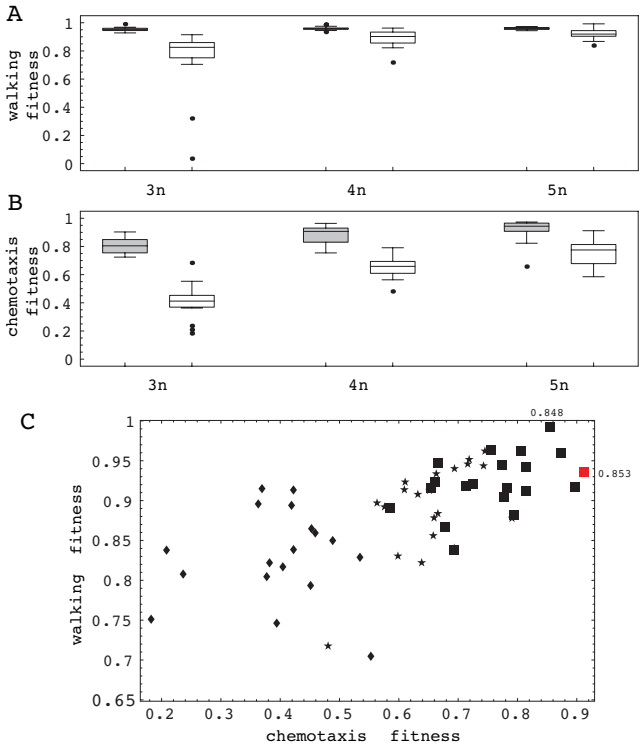


Figure 2: Evolutionary performance. Whisker plots (25% to 75% quantiles and outliers as points) comparing the fitness achieved by 3-, 4- and 5-node networks after having evolved for only one task (grey) and both tasks (white) in the case of walking [A] and chemotaxis [B]. [C] Chemotaxis versus walking fitness on the best 3- (diamonds), 4- (stars), and 5-node (squares) circuits evolved for both tasks.

Performance and behaviours

The best evolved agent achieved a walking fitness of 93.59% and a chemotaxis fitness of 91.18%. This individual corresponds to the red square from Figure 2C. In Figure 3 we show an example trial with this circuit performing both tasks. It is relevant to note that all neurons are active during both tasks.

Walking The optimal walking pattern for this one-legged model has been studied in (Beer et al., 1999). As can be seen from Figure 3B, the evolved pattern is almost perfectly aligned with the optimal pattern, at least geometrically. The different sections in this pattern correspond to particular stages of the walking cycle (labelled in grey): (1) foot up and swing, (2) foot down, (3) stance power, and (4) stance coast. This agrees with results in (Beer et al., 1999). Yet, we know the performance is only 93.59% of the optimal asymptotic velocity that the walking agent can achieve (0.627). The difference is in the timing. The best evolved circuit has 2 units of time delay between the mo-

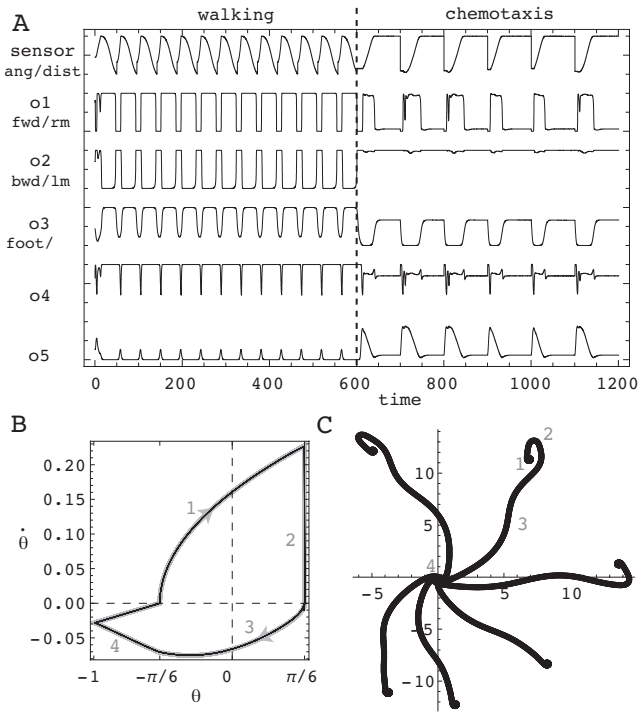


Figure 3: Example trial for best evolved circuit. [A] Sensory and neural pattern during walking and chemotaxis. Transition denoted by horizontal dashed line. [B] Walking behaviour in black. Angle (θ) versus angular velocity ($\dot{\theta}$). Optimal walking in thick grey. [C] Chemotaxis behaviour over six presentations of food.

ment it sets its foot down (phase 2) and the moment it starts to move the leg backwards (phase 3). This unnecessary foot 'rest' causes the degradation in performance, nevertheless maintaining the optimal geometry of the walking pattern. Thus, the best-evolved circuit completes the full cycle at a slightly slower rate than the optimal. If we turn the sensory feedback off, the circuit cannot walk. This is expected from the RPG.

Chemotaxis Even though it has often been used as an example in the artificial life literature, chemotaxis has not been studied in as much depth as the one-legged insect walker. Our agent has only one non-directional sensor. Hence the only way to detect the chemical gradient is by moving about. Organisms that are too small to sense the gradient along the length of their own bodies are in a similar situation. The only strategy available is to use subsequent sensory signals to estimate the chemical gradient in time rather than space.

We can identify four relatively different phases in the best-evolved agent during a chemotactic run: (1) circling search, (2) decreased turning in direction towards the gradient, (3) straight run, (4) circling around food patch. However, this is only a simplified observer-perspective heuristic and the phases are not always clear-cut. The full story is provided only by a geometrical account.

For the initial problem of finding the gradient the agent employs a heuristic that involves circling on the spot while the distance from the food is either constant or decreasing, and moving straight otherwise (i.e. while it is increasing). To confirm this hypothesis, we performed a series of experiments in which we allowed the agent to move while controlling the sensory information arbitrarily. We considered the initial transient behaviour under two conditions: when the sensor value is fixed or decreasing, and when it is increasing at a constant rate. During the former, the agent circles around a small region of about 2 units of space. During the latter, the agent reduces the turning behaviour as a function of the rate of increase of the sensor activity.

Interestingly, the chemotaxis behaviour of the best-evolved circuit employs a strategy similar to that observed in very simple organisms. In *E. coli*, for example, chemotaxis is achieved by modifying the frequency of 'tumbling' (Macnab and Koshland, 1972). In *C. elegans*, the turning behaviour is referred to as a 'pirouette' but the heuristic is similar (Pierce-Shimomura et al., 1999).

Switching behaviours We know the agent can perform well doing each task independently. In order to test whether it can also switch between them during its lifetime we change the circuit's body without resetting the state of the neurons and evaluate the circuit's performance. Although populations were not evolved to cope with this transition, most of the successful circuits managed to switch between tasks in both directions, including the best one analysed here. The example shown in Figure 3A is for a successful transition in one of the directions: from walking to chemotaxis. We will answer why this is possible in the last section of the results.

Dynamics of the decoupled circuit

As a first step towards understanding the evolved behaviours, we consider the dynamics of the circuit when decoupled from the environment. We do this by examining the asymptotic behaviour of the circuit after replacing the time-varying sensory input with a fixed parameter, thereby reducing it to an autonomous system.

Bifurcation diagram In Figure 4 we show the asymptotic behaviour of the circuit as a function of the possible sensory perturbations that it can receive. Solid black trajectories represent attractors, dashed black trajectories represent saddle-nodes and grey dots correspond to limit cycles. Three bifurcations can be observed and are shown in the figure as colored disks. From left to right, the first bifurcation is a saddle-node bifurcation (red disk), from which a fixed point ($a2$) and the saddle node (sn) arise. Fixed point $a1$ is a stable spiral point for $s < 0.38$. This spiral is weakened and a

stable limit cycle (lc) arises near the origin in what is likely to be a Hopf bifurcation (green disk). The size of the cycle first increases slowly and then comes crashing inwards until it reverts to a stable spiral point for $s > 0.77$.

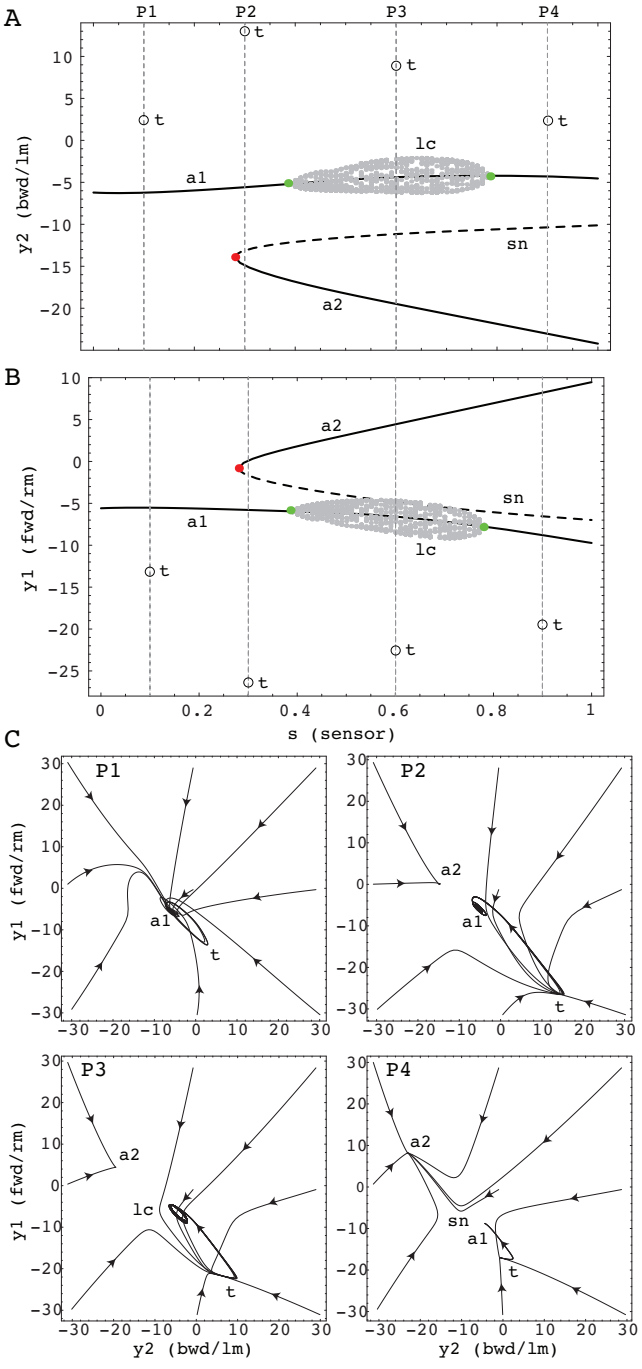


Figure 4: Bifurcation diagram and example phase portraits. [A and B] Two-dimensional slices of the 6D bifurcation space (5 neurons + sensor). [C] Two-dimensional slices of phase-portraits $P1$ through $P4$ for the two effector neurons, y_1 and y_2. See main text for an explanation of the labels.

Phase portraits The bifurcations divide the space of qualitatively different dynamics available to this circuit in four. The dashed vertical lines in Figures 4A and 4B represent the slices of parameter space that are studied in Figure 4C, which shows two-dimensional slices of phase-portraits $P1$ through $P4$. The portraits are shown only for neurons 4 and 5, which control forward and backward movement in both insect and khepera bodies. $P1$ corresponds to the family of phase portraits available when $s < 0.28$ before the first bifurcation occurs. As there is only attractor ($a1$) in the system all trajectories are drawn to it. The paths taken to get to it are not direct, but follow a spiral in its vicinity. Prior to this, however, a subset of the trajectories follow a much longer transient involving a loop near the region labelled t.

$P2$ corresponds to the family of phase portraits available between the first and second bifurcations ($0.28 < s < 0.38$). It comprises two stable fixed points $a1$ and $a2$, and a saddle-node (not shown). Trajectories starting in the top-left corner approach the newly created stable point, $a2$, whose basin of attraction (not shown) is smaller than $a1$'s. Hence, most trajectories approach $a1$. The transient towards it is, again, not direct. In fact, as we will see, this is always the case for this attractor. What varies is the spatial extent of the spiral loop. When looking through the perspective of neurons 4 and 5, any trajectory bound for $a1$ will first navigate towards t. In $P3$ the spiral attractor becomes a stable cycle. The transient remains similar.

In $P4$ the cycle disappears and gives way to a stable fixed point. Also, $a2$'s basin of attraction becomes larger, with certain initial configurations ending up in $a2$ that previously ended in $a1$. Also, the effect of the saddle-node (sn) becomes more obvious in this portrait. The transient loop (t) still exists, but it is relatively closer to $a1$.

Finally, approximations of the turning point of the transient loop are incorporated into our bifurcation diagram as disks labelled t in Figures 4A and 4B. As these are not real limit sets of the system, they do not show up in our bifurcation analysis. They will play, however, a fundamental role in the agent's autonomous behaviour. If the phase-portrait of the system is changing sufficiently fast (due to rapidly varying input), and if the neural state falls in the basin of attraction of $a1$, then we can predict that it will most likely be seen around t and never actually reach $a1$.

Brain-body-environment coupled dynamics

Let us now consider the behaviour of the agent when coupled to the environment and how it relates to the underlying dynamical landscape described in the previous section. Figure 5 depicts the trajectories of the controller when driven by the agent's sensor, which is itself influenced by the circuit's effectors and the corresponding changes to how the agent perceives the environment. Red lines correspond to the walking task, blue lines to chemotaxis. The trajectories are imposed over a simplified version of the circuit's

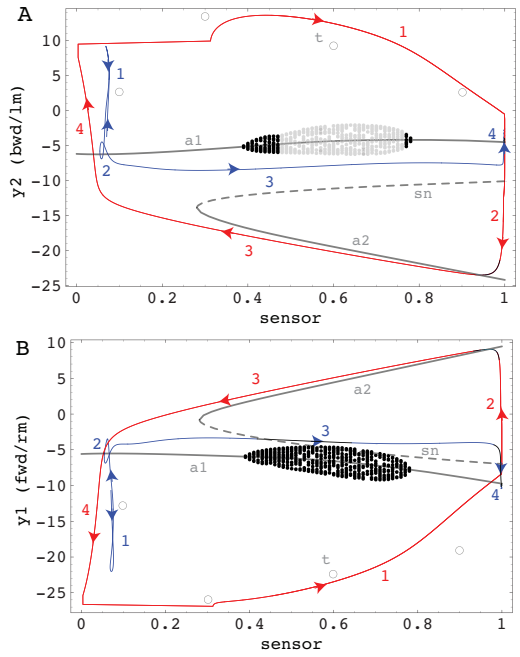


Figure 5: Brain-body-environment coupled dynamics for the two different tasks: walking (red) and chemotaxis (blue). Trajectories are imposed over the bifurcation diagram of the non-autonomous nervous system (grey). Two slices of the 6-dimensional space are shown. The same two slices shown for Figure 4A and 4B, respectively.

autonomous dynamics from Figures 4A and 4B, using the same projections. While the dynamics of the two tasks are significantly different, they share the same underlying dynamical landscape.

During the walking task, the dynamics of the circuit are constantly switching between approaching attractor $a1$ when swinging the leg forward, and approaching attractor $a2$ during the stance power and coast. However, while the system gets close enough to $a2$, it ends up relatively far from $a1$. In fact, the cycle that arises from the coupled system is observed to switch between $a2$ and t (the longer transient towards $a1$). This agrees with our prediction from the circuit's autonomous dynamics, which suggested that it is being driven at a relatively fast rate. We test this hypothesis in the next section. We also note that the cycles in y_2 and y_1 follow opposite directions, clockwise and counterclockwise, respectively. This reflects the antagonistic muscle cooperation necessary to produce the swinging of the leg.

The trajectory during chemotaxis is more subtle and is produced solely within the basin of attraction of $a1$. The circling search behaviour is produced by the longer transient towards t, in combination with the spiral shape in the vicinity of $a1$. However, the state of the neural controller doesn't really reach $a1$ until the agent gets close to the food patch, at which point the sensor gets maximally activated. As soon as the gradient towards the food is found, the sensory value increases and the phase-portrait shifts, leaving the state of the effector neurons in a region of space where the power of the opposing effectors are balanced, which corresponds to moving straight. Interestingly, the spiral attractor and indeed the limit cycle around $a1$ ensure that if the gradient ceases to increase, the agent will circle on the spot until it increases again. This agrees with our observation of the agent's chemotactic heuristic. Finally, once the agent reaches the top of the gradient, the dynamics come cycling in towards $a1$, which ultimately leaves the agent turning on the spot near the food patch.

Behaviour coordination: driven circuit

How does the neural controller perform the appropriate behaviour at the appropriate time? What our analysis shows is that it is *not* the neural controller itself that coordinates the change of behaviours. Instead, different patterns of feedback are created when the neural system is coupled to a different body, and it is these patterns that ultimately produce the distinct transient behaviours. It is worth emphasizing that brain, body and environment form a closed loop such that no single part is the sole cause for the difference in the dynamical patterns. The shape of the feedback is as much the result of neural output and body dynamics as the neural activity itself is the result of environmental feedback.

Is a particular feature of the feedback signal associated with the change of behaviours? From the previous section, we observed that the walking task is generated by the system's movement between two basins of attraction, in such a way that it never actually settled on any one of them. This suggests that fast switching between the two basins generates walking. During chemotaxis, on the other hand, the dynamics stay within the basin of $a1$ and movement is sufficiently slow to allow for it to draw close to the attractor, settling only when sufficiently close to the food patch.

In summary, the essential feature of the feedback is its time-scale. While the sensory feedback from the insect-like body is relatively fast, the sensory feedback from the khepera-like body is much slower. We test this hypothesis by driving the neural system with fast and slow sine waves, and compare the observed dynamics in internal space (Figure 6A) to the dynamics during walking and chemotaxis (Figures 6B and 6C). We find that, depending on the frequency, it will either: (i) jump from one attractor to the other, which is relevant to the walking behaviour, or (ii) stay on the central attractor, which is relevant to chemotaxis. Finally, this provides an explanation for why the evolved agent can switch between behaviours during its lifetime. The behaviours don't depend on where in neural space the state of the system is, but on the rate at which it is being driven by the feedback from its interactions with the environment, as a product of the mechanics of its body.

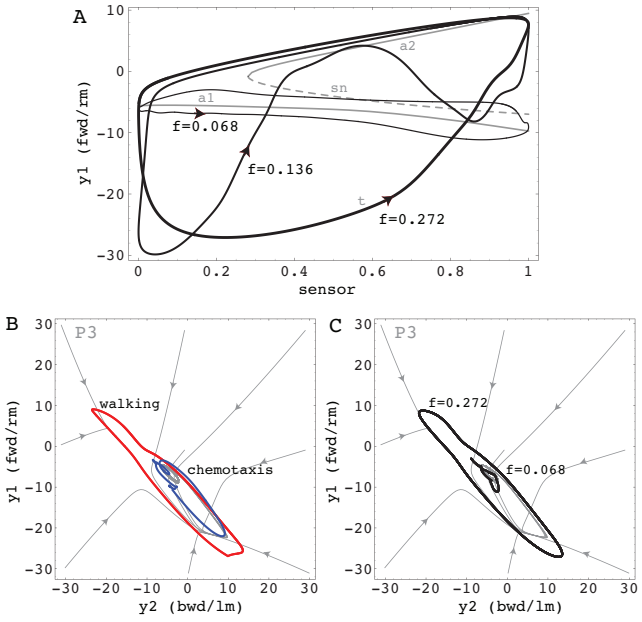


Figure 6: System driven by sinusoidal waves of different frequencies. [A] Trajectories of the neural state of the system are imposed over the circuit's bifurcation diagram (grey, limit cycles not shown). When the system is driven slowly ($f = 0.272$, thin black trajectory), the trajectory remains near the $a1$ attractor. As the system is driven faster ($f = 0.136$ and $f = 0.068$, thicker trajectories), the state of the system starts oscillating between attractors $a1$ and $a2$, but because of the longer transient towards $a1$, the oscillation is effectively between t and $a2$. [B] Two-dimensional slice through the space of neural activity during walking (red) and chemotaxis (blue) imposed over one of the phase-portraits ($P3$). [C] Neural activity for the system when driven by a fast ($f = 0.272$) and slow ($f = 0.068$) sinusoidal wave.

Related work

Synthesizing neural controllers to generate multiple qualitatively different behaviours is a challenge that has been posed by many. However, the focus has been on the role of modularity. Togelius (2004) showed how subsumption architecture models could be merged with an evolutionary robotics approach for a simulated robot on a learning task. Nolfi (1997) investigated modularity for evolution of a garbage collecting robot that had to cope with subtasks such as recognizing, picking up and disposing of desired objects. Although the networks had a hard-wired modular architecture, evolution was free to choose how these modules were used. Calabretta et al. (1999) addressed the same task, but used a system in which neural modules could evolve from a population of non-modular networks through gene duplication. Although both reported improved performance relative to monolithic networks, Ziemke et al. (1999) showed that in a more difficult version of the task, a monolithic recurrent network outperformed all modular architectures.

Many animals can rapidly change between different modes of locomotion. In (Ijspeert et al., 2007), the problem of designing the neural controller for switching between swimming and walking in a salamander-like robot is presented. In (Buckley et al., 2008), an agent is evolved to do phototaxis with the sensor in two different positions (front and back of the body) while constraining the dynamical system controller to use a single basin of attraction. In both papers, however, the two behaviours share a large range of qualities. Our work is different from theirs in that the two tasks (chemotaxis and walking) were chosen to be as different as possible, while sharing sensor and effectors.

Yamauchi and Beer (1994) evolved a simulated robot that had to learn which of two environments it was placed in and take an appropriate action such as to approach a desired position. They only succeeded after dividing the network into separate modules with explicitly assigned roles that were evolved separately. They then evolved a classifier network to determine which of the modules is to control the agent. Tuci et al. (2002) later successfully evolved a controller for a very similar task without dedicated modules. Finally, Beer and Gallagher (1992) evolved agents for chemotaxis and walking, but not for the same dynamical system controller.

Discussion

We have shown that small dynamical neural networks are able to implement qualitatively different behaviours as distinct transients on a single dynamical landscape. Specifically, we evolved an agent that could perform locomotion when coupled to a one-legged body and chemotaxis when controlling a khepera-like robot. We demonstrate this is possible without imposing structural modules on the controller, and without employing complicated fitness functions or evolutionary shaping protocols. Neither was it necessary to introduce parameter changes in the controllers or to provide a signal for when the swap of bodies and corresponding behaviour was to occur. The interactions of neural controller, body and environment alone are sufficient to create distinct transient dynamics appropriate for solving both tasks.

The divide-and-conquer approach championed in engineering would suggest that separate modules should be evolved to produce the two tasks independently. This however wouldn't necessarily simplify the problem, as the main challenge would then be to design a mechanism of coordination and a sophisticated sensing machinery to detect when to switch between the modules. More importantly however, while modular structures and synaptic-plasticity exist in living organisms, they were selected based on the adaptiveness of their behaviour and not on how apprehensible their internal mechanisms are. We therefore argue that understanding networks whose structure is not imposed from the top down will help us develop the tools to understand how multiple

behaviours are generated in living organisms.

Our dynamical systems account of the evolved agent indicates that it is misleading to associate a behaviour with an attractor or a basin of attraction in the decoupled internal dynamics of the controller. In fact, there can be many behaviours in the same basin of attraction, as shown in (Buckley et al., 2008), or single behaviours that require several different attractors, as in the RPG shown here and in (Beer, 1995). Furthermore, as this paper demonstrates, multiple behaviours may use an overlapping set of diverse attractors and their basins. This provides an example of the importance of understanding behaviours as a result of the interactions between brains, bodies and environments, where transients play an equal, if not more important, role than attractors.

A possible objection that could be raised about this work is that the behaviours presented here are either too simple or not sufficiently different from each other and that modular and hierarchical architectures and additional learning rules will be required for more 'complex' scenarios. We believe this is an important, but mostly open question.

Finally, an important feature of recurrent neural networks is that their history of activations allows them to respond to otherwise identical stimuli in a context-dependent fashion. In other words, a system with internal state, when embodied and situated, is not constrained to a single sensori-motor mapping (as was shown in our example). In von Uexkull (1957)'s terms, such systems could be said to "bring forth their own *Umwelt*". But while the act of interpreting sensory input contextually is usually attributed wholly to the agent, the example presented here shows that "meaningful" behaviour is the result of interactions in the brain-body-environment system as a whole.

Acknowledgements

We thank Peter Fine for feedback on an earlier draft of this paper. Mathematica tools developed by Randall Beer were used for parts of the dynamical analysis. This work was partly supported by the ESIGNET 6th Framework EU grant.

References

Beer, R., Chiel, H., and Gallagher, J. (1999). Evolution and analysis of model CPGs for walking: II. General principles and individual variability. *Journal of Computational Neuroscience*, 7:119–147.

Beer, R. D. (1995). A dynamical systems perspective on agent-environment interaction. *Artificial Intelligence*, 72:173–215.

Beer, R. D. and Gallagher, J. (1992). Evolving dynamical neural networks for adaptive behaviour. *Adaptive Behavior*, 1(1):91–122.

Buckley, C., Fine, P., Bullock, S., and Di Paolo, E. (2008). Monostable controllers for adaptive behaviour. In Asada, M., Tani, J., Hallam, J., and Meyer, J.-A., editors, *Proc. of the Tenth Int. Conf. on the Simulation of Adaptive Behavior*.

Calabretta, R., Nolfi, S., Parisi, D., and Wagner, G. (1999). Emergence of functional modularity in robots. In *Proc. of the Fifth Int. Conf. on Simulation of Adaptive Behavior*, pages 497–504. MIT Press.

de Bono, M. and Maricq, A. (2005). Neuronal substrates of complex behaviors in *C. elegans*. *Annual Review of Neuroscience*, 28:451–501.

Ferree, T. and Lockery, S. (1999). Computational rules for chemotaxis in the nematode *C. elegans*. *Journal of Computational Neuroscience*, 6:263–277.

Hart, A. (2006). Behavior. In Ambros, V., editor, *WormBook*.

Harvey, I. (2001). Artificial evolution: a continuing SAGA. In Gomi, T., editor, *Evolutionary Robotics: From Intelligent Robots to Artificial Life.* Springer-Verlag LNCS 2217.

Hobert, O. (2003). Behavioral plasticity in the *C. elegans*: Paradigms, circuits, genes. *Journal of Neurobiology*, 54:203–223.

Ijspeert, A. J., Crespi, A., Ryczko, D., and Cabelguen, J.-M. (2007). From Swimming to Walking with a Salamander Robot Driven by a Spinal Cord Model. *Science*, 315(5817):1416–1420.

Macnab, R. and Koshland, D. (1972). The gradient sensing mechanism in bacterial chemotaxis. *Proc. Natl Acad. Sci.*, 69(9):2509–2512.

Nolfi, S. (1997). Using emergent modularity to develop control systems for mobile robots. *Adaptive Behavior*, 5(3-4):343–363.

Pierce-Shimomura, J., Morse, T., and Lockery, S. (1999). The fundamental role of pirouettes in caenorhabditis elegans chemotaxis. *Journal of Neuroscience*, 19(21):9557–9569.

Rankin, C. (2004). Invertebrate learning: what can't a worm learn? *Current Biology*, 14:617–618.

Togelius, J. (2004). Evolution of a subsumption architecture neurocontroller. *Journal of Intelligent and Fuzzy Systems*, 15(1):15–20.

Tuci, E., Quinn, M., and Harvey, I. (2002). An evolutionary ecological approach to the study of learning behavior using a robot-based model. *Adaptive Behavior*, 10(3–4):201–221.

von Uexkull, J. (1957). A stroll through the worlds of animals and men: A picture book of invisible worlds. In Lashley, K. and Schiller, C., editors, *Instinctive behavior: the development of a modern concept*. International University Press.

Watson, R. and Pollack, J. (2005). Modular interdependency in complex dynamical systems. *Artificial Life*, 11(4):445–457.

Yamauchi, B. and Beer, R. (1994). Integrating reactive, sequential and learning behavior using dynamical neural networks. In Cliff, D., Husbands, P., Meyer, J., and Wilson, S., editors, *Proc. of the Third Int. Conf. on Simulation of Adaptive Behavior*, pages 382–391. MIT Press.

Ziemke, T., Carlsson, J., and Bodn, M. (1999). An experimental comparison of weight evolution in neural control architectures for a 'garbage-collecting' khepera robot. In Loffler, A., Mondada, F., and Ruckert, U., editors, *Proc. of the First Int. Khepera Workshop*, pages 31–40.

The Evolution of Evolvability in Gene Transcription Networks

Eduardo J. Izquierdo[1,2] and Chrisantha T. Fernando[3,2]

[1]Centre for Systems Biology, University of Birmingham, Edgbaston, Birmingham B15 2TT, UK
[2]Centre for Computational Neuroscience and Robotics, University of Sussex, Brighton BN1 9QH, UK
[3]Mathematical Biology, National Institute for Medical Research, Mill Hill, London NW7 1AA, UK
eji21@sussex.ac.uk, ctf20@sussex.ac.uk

Abstract

We present a case of a genotype-phenotype map, that when evolved in variable environments optimizes its genetic representation to structure phenotypic variability properties, allowing rapid adaptation to novel environments. How genetic representations evolved is a relatively neglected topic in evolutionary theory. Furthermore, the "black art" of genetic algorithms depends on the practitioner to choose a representation that captures problem structure. Nature has achieved remarkably efficient heuristic search mechanisms without top-down design. We propose that an important example of this, ubiquitous in biology is the structuring of the phenotypic variability properties of gene networks. By studying a simple model of gene networks in which topology is a function of interactions between transcription factor proteins and transcription factor binding sites (TFBS), we show that transcription factor binding matrices (TFBM) evolve to positively constrain phenotypic variability in response to transcription factor binding sequence mutations.

Introduction

Where there is redundancy in the genotype to phenotype map, there is neutrality. For a given phenotype, if the distribution of phenotypes accessible by one-mutant neighbours differs depending on the particular genotype that encodes the initial phenotype, then there is non-trivial neutrality (Toussaint, 2003), meaning that there is variation in the phenotypic exploration distribution. Toussaint has shown that selection can act on the effective fitness of exploration distributions (i.e. the quasispecies fitness), and claims that this is the mechanism for the evolution of evolvability (Wagner and Altenberg, 1996). Evolvability is the capacity to rapidly adapt to novel environments by natural selection.

Add to this that the environment of the offspring differs from the environment of the parent, due to mutation and external variations, e.g. bacteria may sometimes be in the gut, and at other times outside the host (Ciliberti et al., 2007). Co-evolution is the norm, rather than the exception, meaning that the phenotype required for 'optimum' fitness is always changing, i.e. the fitness landscape fluctuates. We show that a fluctuating fitness landscape selects for exploration distributions with greater evolvability.

What mechanisms are capable of "biasing the kind and amount of phenotypic variation produced in response to random mutation, such that more favourable and non-lethal kinds of variation are available on which natural selection can act" (Kirchner and Gerhart, 1998). We demonstrate that a ubiquitous developmental mechanism, the formation of gene networks based on TFBM-TFBS interactions, has the capacity to allow heredity of exploration distribution variants in gene network topology space, and that this is likely to be the case in real gene networks.

A notable limitation of evolutionary algorithms is that the variational machinery is not self-referentially encoded[1], whereas in embodied evolution (because of the *necessity* of a developmental decoding of the genotype to produce the phenotype) the genetic representation (Toussaint, 2003), or genotype-phenotype map (Wagner and Altenberg, 1996), can maintain variants in the exploration distribution that can be acted upon by positive selection. Reisinger and Miikkulainen (2007) claim to have developed an indirect encoding for a neural network capable of structuring the genotype-phenotype map, to speed up the evolution of Nothello players. Other compact and indirect encodings of neural networks are effective in some specific problem domains (Gruau et al., 1996) , (Hornby and Pollack, 2002). Understanding the principles of network evolvability in natural systems is an important goal.

Defining Evolvability and Robustness

Several definitions of evolvability and robustness exist in the literature. We define evolvability as an ordering of the rate of evolution, between individuals of equal fitness, when exposed to directed selection from a starting point, S, to an end point, F, in phenotype space, where S is not equal to F. One could say that an individual A is more evolvable than individual B (from S to F) if the best offspring of A is on average fitter than the best offspring of B (Turney, 1987). Another quantitative measure of the evolvability of a variational system is the probability that an offspring has fitness

[1]Self-referential encoding refers to a genotype-phenotype map that is capable of non-trivial neutrality.

greater than the parent (Barnett, 2003). Given a probability density function of offspring fitnesses from a single parent, the evolvability of that parent is the fraction of offspring fitter than itself (Smith et al., 2002; Altenberg, 1994a)[2]. Note that with these definitions we can only unbiasedly compare evolvability between individuals of the same fitness. Also, there is no such thing as evolvability when $S = F$, since there is no directed selection, and so one cannot measure a rate of evolution. On the other hand, in the case where $S = F$, robustness *is* defined. It is a measure of the capacity of phenotypes to remain unchanged, given stabilizing selection. Evolvability and robustness are measures of evolutionary behaviour. They are emergent properties of exploration distributions (McGregor and Fernando, 2005).

Evolvability Sustaining Mechanisms

Kirchner and Gerhart (1998) have described various mechanisms that increase the probability that an offspring, varying within certain bounds, will be viable. One important class of these is exploration and exploitation mechanisms. For example, the immune system adapts to evolutionarily novel antigens by implementing somatic selection. Microtubules control and manipulate cell organelles and chromosomes, independently of the number and location of these items, thus allowing variation in these items to be viable, with respect to mitosis for example. Pathfinding by axonal growth cones allows neural structures to evolve, and still be viable (Kirchner and Gerhart, 1998). These search mechanisms allow robustness as well as evolvability (Wagner and Altenberg, 1996).

Another mechanism that confers evolvability and robustness is weak interaction. Kirchner and Gerhart (1998) contrast the complex transcription regulation of eukaryotes with the simple regulation of prokaryotes. Eukaryotes have complex cis-regulatory regions whereas prokaryotes do not. Eukaryotes have enhancer-binding proteins with limited affinity and low sequence specificity for enhancer sites. Their binding affinities may also be contingent on other proteins (non-independent site affinities). This property is called *weak-linkage*. Several authors have considered the role of weak interactions. It has been hypothesised that *weak interactions* confer evolvability (Conrad, 1990), and robustness (Volkert and Conrad, 1998). For example, Kirchner and Gerhart (1998) claim that Calmodulin is a *versatile inhibitor*, meaning that with few mutational steps it can bind to a protein for which it is selected to bind, attributing this to its "low sequence requirements" for binding to targets, that result from its flexibility and stickiness.

The above mechanisms *extend* the viable range of the exploration distribution, rather than constraining its directionality as emphasized in (Arthur, 2004). The exploration distribution can also make sense of another class of mechanism prevalent in bacteria. These are the mechanisms that maintain genotypic diversity in the population, increasing the chance that at least some of the existing variants will be pre-adapted to the new environment. This is a kind of bet-hedging. Typically, *E. coli* isolated from populations in the wild contain a small proportion with a 100 fold increased mutation rate due to inactivation of an error correcting enzyme (Matic et al., 1997). This proportion is much greater than expected in the absence of selection (Tenaillon et al., 2001). The full gamut of genetic and epigenetic devices for structuring variation (in terms of rate, site and inducibility) is discussed in (Rando and Verstrepen, 2007). Remarkably, Kussell and Leibler (2005) have shown that where the cost of maintaining diversity is less than the cost of sensing the environment, stochastic switching is selected over more complex sensing and response mechanisms. This is often the case in bacteria. The exploration distribution is skewed by such processes.

There are clear examples of highly conserved core processes that have optimized exploration distributions. According to a model by Zhu and Freeland (2006) the genetic code is optimized to allow the rapid adaptive evolution of proteins. Using a simple model of a sequence-to-protein-structure map, they mutated the sequence thus altering the structure. They used a genetic algorithm to re-evolve the original structure and found that with the existing genetic code, the structure could re-evolve much faster than if a randomly chosen code was used. Protein stability has also been argued to be an adaptation for evolvability. Bloom et al. (2006) showed using a lattice protein model that "extra stability is neutral with respect to selection for protein function, but it can be crucial in allowing a protein to tolerate [destabilizing] mutations that confer beneficial phenotypes". That is, a protein with more stability was able to evolve to a new desired function faster.

Modularity of various forms appears to underlie many kinds of evolvability (Wagner and Altenberg, 1996; Force et al., 2005; Lipson et al., 2002), since adaptation to environment A can be carried out without interfering with adaptation to environment B. At the cellular level, there may exist exotic exploration distribution structuring mechanisms that remain mysterious. For example, the pattern of gene expression in the lifetime of a paramecium influences which genes are passed to its offspring in a very complicated way (Prescott and Rozenberg, 2002).

Cognition is the cherry on the cake of exploration distribution structuring systems. The effectiveness of lifetime variation generation mechanisms tends to increase over evolutionary time, with solutions being transmitted across generations in novel ways. Many aspects of cultural inheritance, permitted by human thought and language, are adaptations that allow rapid adaptation to novel environments. Cognitive

[2]There are many other definitions of evolvability with different emphasis, e.g. "evolvability is the ability of the genetic system to produce *and maintain* potentially adaptive genetic variants" (Hansen, 2006).

mechanisms for generalization such as associative learning and symbolic reasoning allow us to choose behaviours with remarkable directedness compared to random search.

The Evolution of Evolvability Sustaining Mechanisms

How did such mechanisms evolve? Toussaint (2003) describes that selection pressure could act on *effective fitness*. To produce robust encodings Toussaint selects explicitly for neutral variants, a process that is intended to mimic stabilizing selection. This corresponds to Kirchner and Gerhart (1998) who say that evolvability is a by-product of selection for robust development in the face of internal (mutational) and external (environmental) noise. Secondly, adaptations for evolvability may have been selected because they allowed better exploration of new environments by clades. Hitchhiking of evolvability conferring genes along with advantageous traits whose appearance they facilitate is one mechanism that could achieve this (Conrad, 1990). Recently, Earl and Deem (2004) have shown in a model of protein evolution that the rate of mutation and the "swapping" of protein modules increases in variable environments to confer greater evolvability. Another proposed mechanism is constructional selection where selection acts to filter new loci. Alleles at new loci that have low epistasis are favoured because they are less likely to be fatal, resulting in modular genotype-phenotype mappings Altenberg (1994b). Finally, Kashtan and Alon (2005) have shown that selection in variable environments with modular goals results in the establishment of an intermediate genotype state that can rapidly mutate to become optimal in either environment.

We demonstrate that adaptations for evolvability in gene networks arise due to individual level selection in variable environments. By using a genetic algorithm to evolve agents under fixed versus variable environments, we identify how exploration distributions are restructured by TFBM evolution, a process that is also expected in natural evolution.

The Construction of Gene Networks

Gene network topology emerges from the interaction of transcription factors (TFs) binding to "degenerate families" of transcription factor binding sites (TFBSs) that are of 5-25 nucleotides in length, situated on promotors (Moses et al., 2003). Degenerate refers to the fact that different transcription factor binding sites that bind the same TF protein may differ in 20-30 percent of bases (Collado-Vides et al., 1991). In *E. coli*, promotors are approximately 500 base pairs long and contain several TFBSs (Berg et al., 2004). A position-weight matrix (or transcription factor binding matrix, TFBM) represents the binding preferences of a TF. Empirically these can be inferred from genome sequences if independent evidence of TF binding exists (Stormo, 2000). The binding energy between a TF and the TFBS can be well approximated by the sum of independent contributions from positions in the binding site.

What are the modes of evolution of gene regulatory networks? Gelfend (2006) discusses the various approaches to this question. Many phenotypic differences between species are attributable to changes in gene expression patterns rather than changes in structural or metabolic proteins (Tirosh et al., 2008). How does evolution of the gene regulatory network take place? Babu et al. (2006) have shown that different species of bacteria have evolved new transcription factor proteins by duplication, divergence and sometimes subsequent loss of transcription factors. Wagner et al. (2007) have shown that transcription factor binding site abundance is under selection, and varies considerably between species.

Topological changes of real gene networks also occur on very short evolutionary timescales, especially in higher eukaryotes (Stone and Wray, 2001). In contrast to gain and loss of transcription factor proteins, these changes are caused by mutations in promotors that produce novel transcription factor binding sites.

At an even finer grain, Moses et al. (2003) have shown that within a transcription factor binding site (TFBS), some nucleotide positions show greater variation than other nucleotide positions, between related species and within the same genome. There is evidence that the more degenerate positions (i.e. those with lower information content) in the TFBM are those where the TF does not make much contact with the DNA, i.e. where the total stabilization energy in that column of the TFBM is low (Mirny and Gelfand, 2002).

We simulate gene network evolution to investigate under what conditions TFBMs evolve to improve evolvability.

Methods

Our gene network model considers N interacting units. Each unit consists of a transcription factor binding site (TFBS) that produces a transcription factor with a particular transcription factor binding matrix (TFBM). The interaction between this matrix and the TFBS sequence determines how a transcription factor will bind to the transcription factor binding site. All TFBSs have length K nucleotides. At each position in the TFBS , one of four bases can be present (A, T, G, or C). The TFBM is of size 4 by K. Each entry in the matrix contains a real number between 0 and 1. Each number represents the binding strength contribution of the nucleotide at that position in the TFBS to the total TF binding strength upon that TFBS. For simplicity, our TFBM implementation assumes that the contribution of different positions along the TFBS to the binding strength is independent.

The edges of the gene network graph are directed *from* the gene producing the TF *to* the gene possessing the TFBS. The strength of each edge is calculated as described above, by summing independent binding strength contributions from each position in the TFBS as specified in the TFBM. A low accumulated strength represents weak binding between that TFBM_i and TFBS_j. A larger binding strength represents

tighter binding. To specify a *desired* gene network topology, we define a certain binding strength as required for 'ideal' binding. A binding strength greater than this optimum is considered maladaptive. The biological justification for this is the implicit assumption that the TF should be sensitive to modifiers of binding. If it binds to strongly then the protein that it stimulates might as well have been constitutively expressed. More precisely, the fitness contribution of one edge of the gene network corresponds to the inverse of the Euclidean distance between the individual's topology and the desired topology, as given by:

$$f = \frac{1}{1 + \sqrt{\sum_i \sum_j \left((s_{ij}/\lambda) - t_{ij}\right)^2}} \quad (1)$$

where s_{ij} is the binding strength for connection TFBM$_i$ and TFBS$_j$; t_{ij} is 0 or 1 depending on whether a connection is desired to be present or absent (respectively); and λ is the ideal binding strength ($\lambda = 3$). If the Euclidean distance of a topology from an ideal topology X is less than its Euclidean distance from all other ideal topologies, then it is classified as topology type X. An ideal topology is one where the above fitness score is maximized. Note we do not consider the dynamics of the gene network in generating its topology, e.g. we assume TFs are always produced constitutively. Our selection function acts directly on the non-functional topology of the gene network.

Environments are modeled simply as desired gene network topologies. For example in environment A, individuals are optimal that have topology T_a. In a different environment, B, an altogether different topology might be required to survive, T_b.

The parameters of each gene network (i.e. TFBS and TFBM for each gene) are evolved using a microbial genetic algorithm (Harvey, 2001). There are NK discrete and $4NK$ real-valued parameters forming the genotype of each individual. The winner of a randomly chosen pairwise tournament replaces some of the loser's genes with its own. Each TFBM or TFBS of the loser is replaced by the corresponding genome parts of the winner with 0.9 probability. The resulting genome then undergoes mutation. Each TFBM or TFBS will mutate with $1/NK$ probability. On average, one of the full set of TFBSs changes to a base chosen at random. Mutation in the TFBM is implemented as a random displacement on every binding strength drawn uniformly from a Gaussian distribution with mean 0 and variance 0.01. Each strength in the binding matrix is forced to stay within the range 0 to 1. When a mutation takes it out of this range, it is reflected back. A generation is defined as P microbial tournaments, where P is the size of the population. All evolutionary experiments were conducted with populations of 100 individuals. We have not investigated how the findings depend on the choice of genetic algorithm. The major point to note is that TFBMs evolve much more slowly that TFBSs.

Several previously defined measures of robustness and evolvability are considered (Ciliberti et al., 2007; Zhu and Freeland, 2006). *Mutational robustness* is the fraction of one-mutant neighbours of the genome that are also viable. In this case, viability means that the gene network topology resembles the desired topology more closely than all other topologies (with the same number of genes). This is a measure of neutrality or redundancy of coding. *Topology connectivity* is the number of distinct 1-mutant topologies reachable from any one topology. High connectivity implies easy conversion between topologies.

The above properties can be calculated by studying the individual's *hierarchical metagraph* (see Figure 1). In this graph, a higher level node (large circle) represents a particular gene network topology (phenotype). Higher level nodes are connected if by one TFBS mutation, topology A can become topology B, and vice versa. Within a higher level node of a metagraph, there is another embedded graph whose nodes are the genome sequences of TFBSs that can sustain the particular gene network topology represented by the higher level node (i.e. sequence nodes). The higher level node can alternatively be thought of as a labeling of lower level sequence nodes. We define the *connectivity matrix* of a metagraph as the number of one-mutant neighbours connecting topology t_i to topology t_j, over all possible topology transitions.

It is clear that the hierarchical metagraph will depend critically on the evolved TFBMs. The TFBMs *shape* the phenotypic effect of a TFBS mutation. We define the *connectivity variance* simply as the variance over all values of the connectivity matrix. We use this measure as an index of the navigability of the metagraph. The measure synthesizes the system's mutational robustness and its topological connectivity. Our hypothesis is that evolvability in variable environments increases as the connectivity variance decreases, improving the navigability of gene network topology space, as promoter sequence space is explored. Expressed in another way, we propose that TFBM evolution structures the exploration distribution in phenotype space, by reducing connectivity variance.

Let us illustrate what a hierarchical metagraph is and how we can measure its navigability. We will use the simplest gene transcription network: a one-gene network (see Figure 1). There are only two possible topologies: self-binding (t_1) or non self-binding (t_2). Let's suppose that the length of the promoter region is 1. Therefore, there are only 4 possible sequences: A, G, C, or T. The self-binding would depend on the transcription factor binding strength. There are three possible scenarios that can arise. In the first scenario, all possible sequences generate topology t_1 and no sequences result in topology t_2. There are 12 (i.e. BNK) ways in which a sequence can mutate from sequence S_i to S_j where S_j is only one mutation away. In the first case, all of the mutations are neutral. There are 12 ways of going from t_1 to

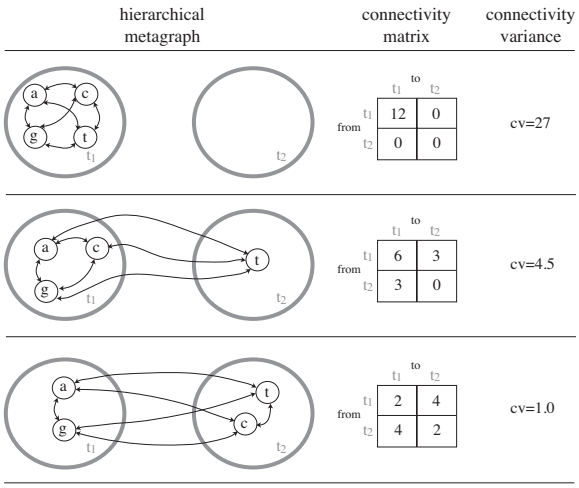


Figure 1: Hierarchical metagraph and the notions of its *connectivity matrix* and *connectivity variance* in the simplest possible one-gene regulatory network scenario.

t_1, and 0 ways of going from t_1 to t_2, t_2 to t_1, t_2 to t_2. The connectivity variance is the highest possible, in this case 27. In the second scenario, there is one sequence that produces t_2. The number of one-neighbour mutants shifts accordingly and the connectivity variance drops. In the final case, the connectivity variance is minimal.

From the example illustrated in Figure 1, it should be clear that the connectivity variance will be lower for more easily navigable metagraphs. Accordingly, easily navigable metagraphs will reduce the time required to adapt to any topology, including novel topologies. Connectivity variance is an index of evolvability in variable environments. We demonstrate that the properties of the metagraph, as measured by its connectivity variance, promote evolvability in this selection scenario.

Results

We first present an experiment where a population of 3-node gene networks are evolved in two different environments. The length of each TFBS, K, is 5. In one environment the target topology is a feedforward loop and in the other environment it is a feedback loop (see Figure 2A, the difference is colored red). The evolved sequences and TFBMs for the best individual are shown in Figures 2B and 2C, respectively. The result is a set of TFBMs such that, with a total of 3 site mutations (colored red) in each of the relevant TFBSs, the desired change in topology is achieved. Once the environmental transitions have been experienced several times, adaptation occurs without significant changes to the TFBMs.

Figure 2D shows that the time taken to adapt to a particular environment decreases over evolutionary time. Each environment was presented for 1000 generations. While it

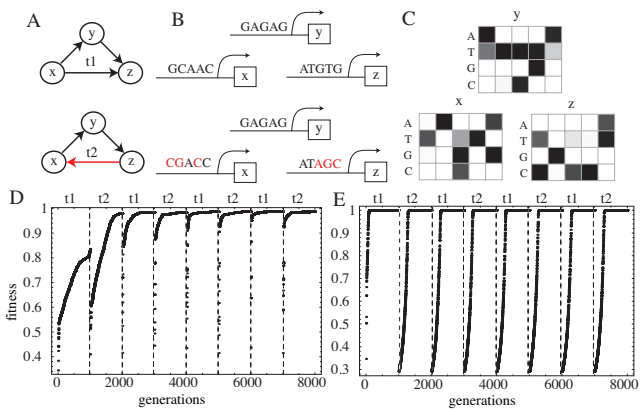


Figure 2: Example evolutionary run with variable environments for a 3-node GRN. [A] Two topologies evolved for: feedforward (t_1) and feedback loop (t_2). [B] Evolved promoter regions for each topology (TFBSs). [C] Evolved set of TFBMs. [D] Fitness versus Generations. Dashed vertical lines represent changes of environment. [E] Control experiment with direct encoding of connectivity shows no evolution of evolvability.

takes the population around 800 generations to adapt to environment B the first time it is encountered, this time drops to around 250 generations on the second occasion. From the third occasion onwards, the adaptation time to environment B is under 100 generations. The time to adaptation reaches steady-state after several environmental transitions. This was also observed for runs using different sized networks, lengths, and numbers of environments. For comparison, Figure 2E shows that when a direct encoding of the gene network is used, i.e. a binary connectivity matrix, there is no improvement in time to adaptation.

For a given size of gene network N, there is a minimum TFBS length K that is required to evolve perfect fitness in W distinct environments, see Figure 3 . This is shown in experiments conducted with 2- and 3-node gene networks with different numbers of variable environments (W) and different lengths of promoter sequences (K). Each data point in Figure 3 is an average over 100 evolutionary runs. For an N sized network, $2^{(N^2)}$ different topologies are possible. For any given experiment, the W different topologies the population would be evolved for were chosen at random from all possible topologies. All experiments ran for the same number of generations, 160000. The number of generations per transition is the same for all experiments, 1000. In Figure 3 we show the proportion of populations in which each agent fully adapts (top) and the time to adaptation (i.e. the time taken for the best individual in the population to reach 0.95 of optimal fitness) (bottom) as a function of W and K. Each point in the surface corresponds to the mean over only those populations where each individual adapted to all W environ-

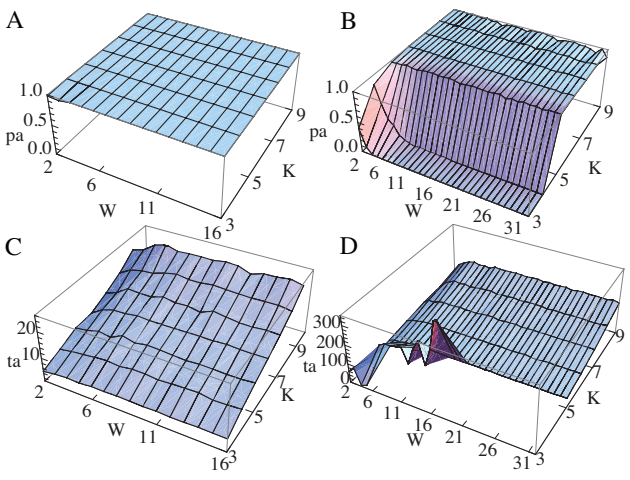


Figure 3: Top: The proportion of 100 independent populations that adapted to all W environments during the last round of environmental presentations as a function of the number of environments presented during evolution (W) and the length of the promoter sequence (K), for $N = 2$ [A] and $N = 3$ [B]. Bottom: The mean time to adaptation of the best individual in the population during the last round of presentations over all environments as a function of W and K, for $N = 2$ [C] and $N = 3$ [D].

ments. No points are shown in conditions where this was not the case.

Referring to Figure 3 C, for $N = 2$ gene networks, TFBS lengths of $K = 3$ are capable of evolving to adapt to all 16 possible topologies. With larger Ks, the adaptation time increases, presumably because the space of possible TFBS sequences also increases. Thus, finding the appropriate sequence that creates the desired topology becomes harder. Interestingly, while the search space becomes larger exponentially with K as given by 4^{NK}, the time to adaption increases only linearly.

Referring to Figure 3 D ,for 3-node GRNs there is a lower limit to K below which adaptation to all environments is not possible. For the minimum, $K = 3$, populations can adapt reliably, only to fixed environments $W = 1$. When tested in variable environments, gene networks with $K = 3$ fail to adapt to 0.95 optimal fitness before the environmental transition at 1000 generations. For $K = 4$ the situation is improved. Populations can adapt to up to 8 different environments. For higher W, the population again starts failing to adapt in time. For $K > 4$, the populations can adapt to varying environments for all W conditions tested.

Does evolution under variable environments increase evolvability? In order to address this question, we tested the ability for evolved individuals to rapidly adapt to *novel* envi-

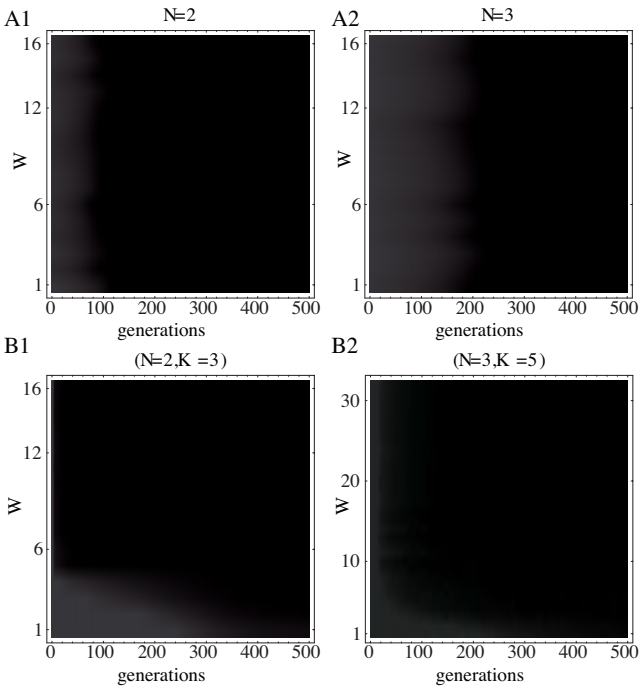


Figure 4: Evolvability. $Generations$ (x-axis) versus environmental variability W. Shades of gray represent the mean of the best individual's fitness over 100 independent evolutionary runs. Black represents poorly adapted populations. White represents well adapted ones. Four conditions are shown: smallest (top row) and largest (bottom row) Ks that evolved successfully under all W conditions tested, for networks of size 2 (left column) and 3 (right column).

ronments. We seeded a new population with genetic 'clones' from the TFBMs of the best evolved individual after 160000 generations. The TFBSs were chosen at random for each individual in the population. Taking into account the topologies the original population had previously evolved for, we set the new population to evolve to achieve a topology it had not previously been exposed to. Different populations were seeded with TFBMs evolved under several conditions. For each condition, the experiment was repeated 100 times.

In Figure 4 (bottom) we show the population's ability to evolve to novel environments under two conditions: $N = 2$, $K = 3$ and $N = 3$, $K = 5$. For all fixed environment conditions $W = 1$, the mean time to adaptation is much larger than for individuals evolved under variable environments $W > 1$. In fact, the ability to rapidly adapt to a novel environment improves with the number of environments that the population was previously evolved for. Finally, all of the conditions show a ceiling effect for sufficiently-varied environments, after which the time to adaptation ceases to improve. For comparison, Figure 4 (top) shows that a direct encoding does not have this property of improved evolvabil-

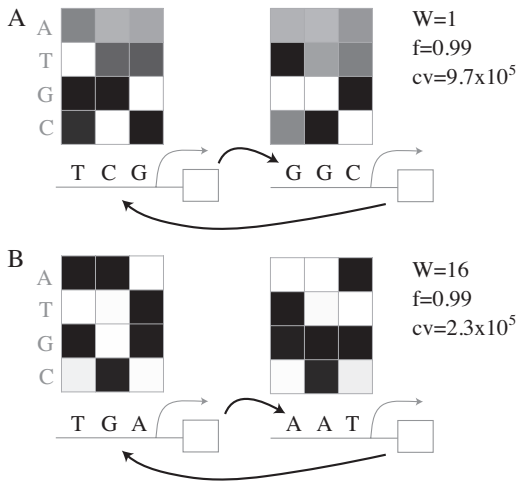


Figure 5: TFBMs evolved for the same topology under fixed [A] and variable [B] environments. Fitness for the feedback topology is equally good for both TFBMs (0.99). However, the connectivity variance for individual B is lower than the connectivity variance for individual A.

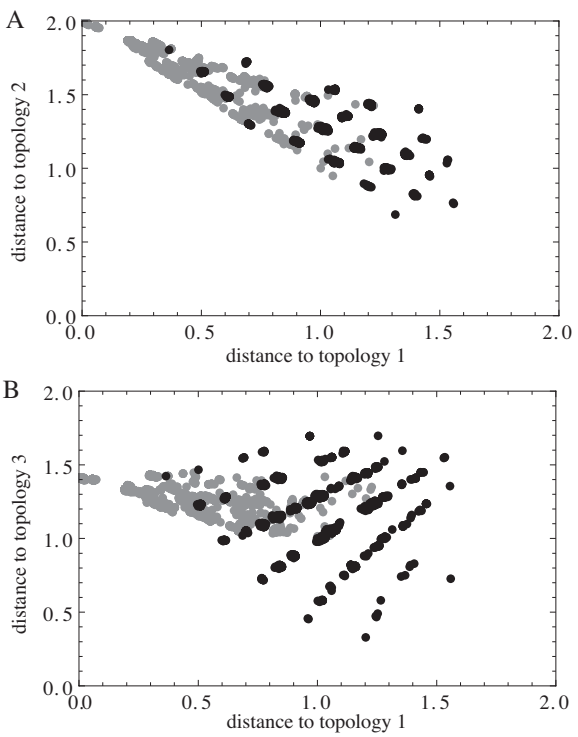


Figure 6: The exploration distribution for the best individual ($N = 2, K = 3$) evolved for a small ($W = 2$, gray) and large ($W = 15$, black) range of environments. The best individual evolved for topology one, t_1, is mutated 5000 times. On the x- and y-axes are shown the Euclidean distance of the resulting mutants to the topologies t_1, t_2 (top) and t_1, t_3 (bottom). Both individuals were evolved previously for t_1, t_2. However, t_3 (a fully connected topology) was evolved for in either case, i.e. it is novel.

ity to a novel environment, given a history of evolution with previously variable environments.

Greater variability of environments experienced during evolution improves evolvability according to the above measure. But why are TFBMs evolved for variable environments more evolvable to new environments than TFBMs optimised for a fixed environment? While two individual's TFBMs can be equally fit for environment T_x, the TFBMs from the individual that has evolved to change to different topologies generates a different exploration distribution to the TFBM evolved in stationary environments.

Figure 5 shows a case where two individuals (A and B) are equally fit for environment T_x, but with different TFBMs. A has evolved in fixed environments while B has evolved for variable environments. While both solutions are equally well adapted to produce the feedback loop topology (i.e. both have fitness of 0.99), the connectivity variance of the gene network evolved for variable environments (B) is less than a quarter of the connectivity variance of the gene network evolved for fixed environments (A). The implication is that the TFBMs evolved for individual B shape the TFBS fitness landscape such that all other topologies are more easily reachable from the present topology. This is not the case for TFBMs evolved for individual A in a static environment.

The exploration distribution can be directly visualized in Figure 6. Individuals evolved in high variability environments, $W = 15$, show a more diffuse exploration distribution to those evolved in low variability environments, $W = 2$. Individuals evolved for $W = 2$ show a tighter exploration distribution, passing from t_1 directly to t_2 without approaching close to other topologies.

Does the connectivity variance always decrease with the number of environments that the populations are evolved for? In order to answer this, the connectivity variance was calculated for all successfully evolved TFBMs for 2- and 3-node networks (see Figure 7). For the 2-node case, the study was exhaustive: taking into consideration all possible sequence configurations ($4^{(}N*K)$) and all possible topologies ($2^{(}N * N)$). For the 3-node case a sample of 1000 different sequence configurations was chosen at random.

The connectivity variance drops as a function of W, as hypothesised (see Figure 7). The connectivity variance results also explain the evolvability ceiling effect observed in Figure 4, where after a certain number of variable environments ($W > 5$) the ability to rapidly adapt to novel environments ceases to improve. The connectivity variance reaches its minimum also for $W > 5$.

Finally, we can compare the evolution of the connectivity variance for the two extreme conditions: fixed or variable

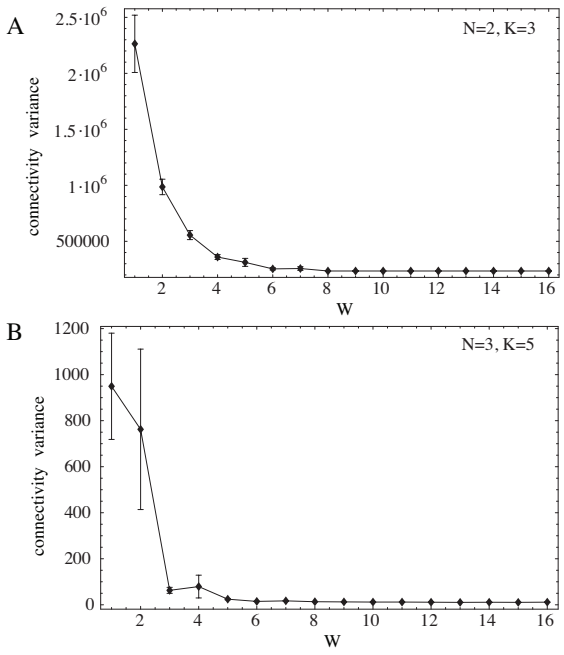


Figure 7: Connectivity variance as a function of variable environments (W). Two conditions are shown: [A] for $N = 2$ and $K = 3$, and [B] for $N = 3$ and $K = 5$. Each point represents the mean over 100 runs (bars represent the standard error).

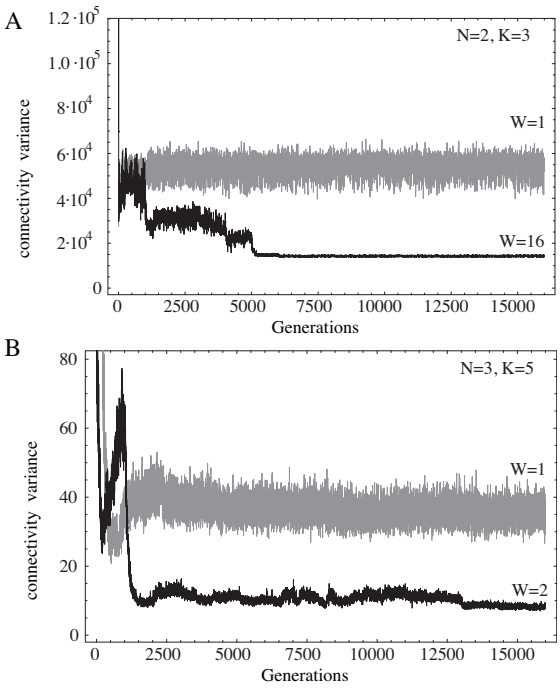


Figure 8: The evolution of evolvability. Connectivity variance of the best individual in the population over evolutionary time. Examples for fixed (gray) and variable (black) environments are shown for 2-node [A] and 3-node [B] GRNs.

environments. In Figure 8 we show the connectivity variance calculated from the TFBMs of the best individual in the population at every generation in four different conditions: fixed (gray) and variable (black) environments are shown for 2-node [A] and 3-node [B] GRNs. Under all conditions the populations evolved successfully (not shown). As can be appreciated in Figure 8, the connectivity variance of populations evolving in varying environments tends to evolve towards lower values (and thus higher evolvability), compared to the connectivity variance of the population evolving in fixed environments. This demonstrates the evolution of evolvability under variable environments.

Discussion

We gave an example of the evolution of evolvability in a system undergoing natural selection in variable environments. In our model, evolvability arose from the ability of the TFBM to evolve to shape the exploration distribution resulting from TFBS mutations. When populations were evolved in variable environments, the TFBMs allowed improved navigability in TFBS space as described by the connectivity variance measure.

The model lacks many features of real GRNs. In reality, promotor sequences are much longer than TFBSs, enhancer proteins modify TF binding, the expression of downstream TFs may be dependent on upstream TFs, and fitness depends on the dynamics of the GRN rather than on its topology alone. Introducing such features whilst maintaining TFBM-TFBS interactions is a challenge, and is likely to uncover further adaptations that could lead to unlimited heredity of exploration distribution variations.

Acknowledgements

This work was supported by the ESIGNET 6th Framework EU grant. Thanks to Richard Goldstein, Jon Rowe and Dov Stekel for helpful discussions.

References

Altenberg, L. (1994a). The evolution of evolvability in genetic programming. In Kinnear, K., editor, *Advances in genetic programming*, pages 47–74. Cambridge, MA: MIT Press.

Altenberg, L. (1994b). Evolving better representations through selective genome growth. *Proc. of the First IEEE Conf. on Evolutionary Computation*, 1:183–187.

Arthur, W. (2004). The effect of development on the direction of evolution:toward a twenty-first century consensus. *Evolution and Development*, 6:282–288.

Babu, M. M., Teichmann, S. A., and Aravind, L. (2006). Evolutionary dynamics of prokaryotic transcriptional regulatory networks. *J Mol Biol*, 358(2):614–33.

Barnett, L. (2003). *Evolutionary Search on Fitness Landscapes with Neutral Networks*. PhD thesis, School of Cognitive and Computer Science, University of Sussex, Falmer, Brighton.

Berg, J., Willmann, S., and Lässig, M. (2004). Adaptive evolution of transcription factor binding sites. *BMC Evol Biol*, 4(1):42.

Bloom, J., Labthavikul, Sy, T., Otey, C., and Arnold, F. (2006). Protein stability promotes evolvability. *Proc. Natl. Acad. Sci. USA*, 103(15):5869–5874.

Ciliberti, S., Martin, O., and Andreas, W. (2007). Robustness can evolve gradually in complex regulatory gene networks with varying topology. *PLoS Computational Biology*, 3(2):0164–0173.

Collado-Vides, J., Magasanik, B., and Gralla, J. (1991). Control site location and transcriptional regulation in escherichia coli. *Microbiol Rev.*, 55(5):371–394.

Conrad, M. (1990). The geometry of evolution. *BioSystems*, 24(61-81).

Earl, D. and Deem, M. (2004). Evolvability is a selectable trait. *Proc. Natl. Acad. Sci. USA*, 101(32):11531–11536.

Force, A., Cresko, W., Pickett, F., Proulx, S., Amemiya, C., and Lynch, M. (2005). The origin of subfunctions and modular gene regulation. *Genetics*, 170:433–446.

Gelfend, M. (2006). Evolution of transcriptional regulatory networks in microbial genomes. *Current Opinion in Structural Biology*, 16:420–429.

Gruau, F., Whitley, D., and Pyeatt, L. (1996). A comparison between cellular encoding and direct encoding for genetic neural networks. In Koza, J. R., Goldberg, D. E., Fogel, D. B., and Riolo, R. L., editors, *Genetic Programming 1996: Proc. of the First Annual Conference*, pages 81–89, Stanford University, CA, USA. MIT Press.

Hansen, T. (2006). The evolution of genetic architecture. *Annu. Rev. Ecol. Evol. Syst.*, 37:123–157.

Harvey, I. (2001). Artificial evolution: a continuing SAGA. In Gomi, T., editor, *Evolutionary Robotics: From Intelligent Robots to Artificial Life.* Springer-Verlag LNCS 2217.

Hornby, G. and Pollack, J. (2002). Creating high-level components with a generative representation for body-brain evolution. *Artificial Life*, 8:223–246.

Kashtan, N. and Alon, U. (2005). Spontaneous evolution of modularity and network motifs. *Proc Natl Acad Sci USA*, 102(39):13773–8.

Kirchner, M. and Gerhart, J. (1998). Evolvability. *Proc. Natl. Acad. Sci. USA*, 95:8420–8427.

Kussell, E. and Leibler, S. (2005). Phenotypic diversity, population growth, and information in fluctuating environments. *Science*, 309(5743):2075–8.

Lipson, H., Pollack, J., and Suh, N. (2002). On the origin of modular variation. *Evolution*, 56:1549–1556.

Matic, I., Radman, M., Taddei, F., Picard, B., Boit, C., Bingen, E., Denamur, E., and Elion, J. (1997). Highly variable mutation rates in commensal and pathogenic e.coli. *Science*, 277:1833–1834.

McGregor, S. and Fernando, C. (2005). Levels of description: A novel approach to dynamical hierarchies. *Journal of Artificial Life*, 11:459–472.

Mirny, L. and Gelfand, M. (2002). Structural analysis of conserved base pairs in protein-dna complexes. *Nucleic Acids Res*, 30(1704-1711).

Moses, A. M., Chiang, D. Y., Kellis, M., Lander, E. S., and Eisen, M. B. (2003). Position specific variation in the rate of evolution in transcription factor binding sites. *BMC Evol Biol*, 3:19.

Prescott, D. and Rozenberg, G. (2002). How ciliates manipulate their own dna - a splendid example of natural computing. *Natural Computing*, 1:165–183.

Rando, O. and Verstrepen, K. (2007). Timescales of genetic and epigenetic inheritance. *Cell*, 128:655–668.

Reisinger, J. and Miikkulainen, R. (2007). Acquiring evolvability through adaptive representations. In *Proc. of Genetic and Evolutionary Computation Conference*, pages 1045–1052.

Smith, T., Husbands, P., Layzell, P., and O'Shea, M. (2002). Fitness landscapes and evolvability. *Evolutionary Computation*, 1(10):1–34.

Stone, J. and Wray, G. (2001). Rapid evolution of cis-regulatory sequences via local point mutations. *Mol. Biol. Evol.*, 18(9):1764–1770.

Stormo, G. (2000). Dna binding sites: representation and discovery. *Bioinformatics*, 16(1):16–23.

Tenaillon, O., Taddei, F., Radman, M., and Matic, I. (2001). Second-order selection in bacterial evolution: selection acting on mutation and recombination rates in the course of adaptation. *Res. Microbiol*, 152:11–16.

Tirosh, I., Weinberger, A., Bezalel, D., Kaganovich, M., and Barkai, N. (2008). On the relation between promoter divergence and gene expression evolution. *Molecular Systems Biology*, 4(159):1–11.

Toussaint, M. (2003). *The evolution of genetic representations and modular adaptation*. PhD thesis, Institute for Neuroinformatics, Ruhr University of Bochum, ND 04, 44780 Bochum—Germany.

Turney, P. (1987). Increasing evolvability considered as a large-scale trend in evolution. In Wu, A., editor, *Proc. of 1999 Genetic and Evolutionary Computation Conference Workshop Program*, pages 43–46.

Volkert, L. and Conrad, M. (1998). The role of weak interactions in biological systems: the dual dynamics model. *J. Theor. Biol.*, 193:287–306.

Wagner, G. and Altenberg, L. (1996). Complex adaptations and evolution of evolvability. *Evolution*, 50:329–47.

Wagner, G., Otto, W., Lynch, V., and Stadler, P. (2007). A stochastic model for the evolution of transcription factor binding site abundance. *J Theor Biol*, 247(3):544–553.

Zhu, W. and Freeland, S. (2006). The standard genetic code enhances adaptive evolution of proteins. *Journal of Theoretical Biology*, 239:63–70.

Energy, Entropy and Work in Computational Ecosystems: A Thermodynamic Account

Mariusz Jacyno and Seth Bullock

School of Electronics and Computer Science
University of Southampton
Southampton, UK
mj04r@ecs.soton.ac.uk

Abstract

Recently, computer scientists have begun to build computational ecosystems in which multiple autonomous agents interact locally to achieve globally efficient organised behaviour. Here we present a thermodynamic interpretation of these systems. We highlight the difference between the regular use of terms such as energy and work, and their use within a thermodynamic framework. We explore the way in which this perspective might influence the design and management of such systems.

Introduction

Modern IT systems are increasingly complex, in some cases resembling computational ecologies or ecosystems comprising myriads of interacting elements, each with their own thread of control and autonomy (Huberman and Hogg, 1993; Bullock and Cliff, 2004). As the scale, dynamism and interconnectedness of such systems increases, effective control via some central executive becomes nontrivial and eventually infeasible. Consequently, there is increasing interest in drawing inspiration from the homeostatic properties of some analogous large-scale, adaptive, decentralised natural systems, and extracting design principles for artificial computational ecologies (Parunak and Brueckner, 2004; Zambonelli and Parunak, 2004). Underpinning this paradigm are concepts and theories of self-organisation that derive from the study of physical systems in far-from-equilibrium conditions (Nicolis and Prigogine, 1977; Kay, 1984).

However, the man-made nature of computational ecosystems gives them a particular teleological status that differs somewhat from the physical (and biological) systems to which thermodynamic accounts of self-organisation are typically applied. Moreover, computational systems are instantiated as physical systems, giving rise to a problem of identifying the level of description at which to apply the self-organisation/thermodynamic interpretation. Partly as a consequence, a consistent thermodynamic account of their behaviour is not straightforward. As engineers aiming to build artificial systems exhibiting adaptive and organised behaviour, we must be careful in applying the concepts and tenets of self-organisation. There may be some value in the informal use of technical language (e.g., attractor, basin of attraction) to re-describe problems, etc., but there are risks of confusion when technical terms (that may also have lay meanings) come to be used as mere façons de parler.

The purpose of this paper is thus to analyse self-organisation in natural systems as governed by the physical laws of thermodynamics and, based on this, to clairfy and make explicit an analogous interpretation of the functioning of artificial self-organising computational ecosystems.

Thermodynamics in natural systems

One powerful strength of a thermodynamic account of self-organisation is its potential to apply across physical, chemical, biological, social, and socio-technological domains. However, it is most clearly and straightforwardly articulated in the absence of the beliefs, desires, and functions that are proper parts of the 'higher' systems. Here we first present the framework in the context of physical and then biological systems before demonstrating its application in the context of a particular class of socio-technological system.

Thermodynamics of self-organisation

Studies investigating the thermodynamics of self-organisation in far-from-equilibrium systems can be found in (Nicolis and Prigogine, 1977; Swenson, 1997; Kauffman, 2000). Irrespective of whether the investigated system is described in terms of 'dissipative structures' (Nicolis and Prigogine, 1977), autonomous agents (Kauffman, 2000) or an autocatakinetic system (Swenson, 1997), self-organisation is interpreted as a process of organised energy flow from which work can be extracted and employed by the system for its structure maintenance (Kay, 1984; Wicken, 1989; Swenson and Turvey, 1991). Central to understanding this process are the following concepts derived from thermodynamics: displacement from equilibrium, energy transfer, gradient dissipation, constraint formation and work.

Displacement from equilibrium

According to classical thermodynamics, the behaviour of physical systems can be explained as transformations of energy between the system and its surroundings. Hence, when both are allowed to interact, what is exchanged between them is energy (Kay, 1984). Energy, here, has a general meaning, defining the capacity of the system to perform work, and may be added to the system by increasing its temperature, pressure or a chemical potential.

Considering the energy of the system and its environment, we can measure the relative difference between both, often defined as a potential or gradient. If the gradient is equal to zero, meaning that both the system and its environment have the same energy (e.g., temperature, or pressure) we consider them to be at equilibrium. In this state, the system is indistinguishable from its environment and has no capacity to perform work. Any deviation from equilibrium implies that free energy is stored, and that there may be the potential to release this energy through useful work. The extent to which a system is displaced from equilibrium is reflected in the gradient (difference) between the state variables defining its energy state (e.g., temperature) and that of its environment.

Energy transfer

To displace a system from equilibrium requires that it be supplied with energy (be it thermal, mechanical or chemical), distinguishing it from its surroundings. According to the first law of thermodynamics, energy transfer can proceed in two different ways: through heat (Q) and work (W). This is captured in the formula summarising the first law:

$$dU = dQ + dW,$$

where dU is the infinitesimal increase in internal energy of the system, dQ is the infinitesimal amount of heat added to the system and dW is the infinitesimal amount of work done on the system. Although heating up a system and performing work on it will each increase its energy, each differs in the manner in which energy is being distributed in the system and thus whether the system moves away from equilibrium.

This difference is reflected through entropy (S) which can be interpreted as a measure of the uncertainty about how energy is distributed in the system (Jaynes, 1965, 1979). Adding heat (Q) to the system increases our overall uncertainty about the energy content of the system and causes proportional increase in entropy. This is manifested through the following relation:

$$dS = dQ/T,$$

where S is the entropy, dQ is the infinitesimal amount of heat added to the system and T is the absolute temperature of the system. For this reason, it represents the amount of energy that we lose information about when it is transferred and that we are thus unable to extract. When, on the other hand, work is done on the system (W) our knowledge about the energy content of the system increases, thus we are better able to distinguish between the system and its environment. In this case, work done on the system does not affect internal system entropy and thus represents the only way to move a system further from equilibrium (Kay, 1984).

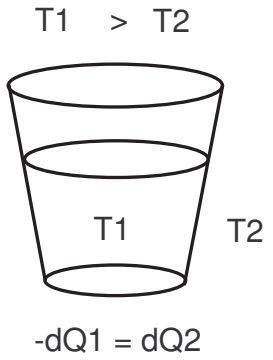


Figure 1: A glass of liquid at temperature $T1$ is placed in a room at temperature $T2$, where $T1 > T2$. The disequilibrium produces a field potential that spontaneously drives a flow of energy in the form of heat, $-dQ1$, from the glass to the room so as to drain the potential until it is minimized (the entropy is maximized). At this point thermodynamic equilibrium is reached and all flows stop. The expression $-dQ1 = dQ2$ refers to conservation of energy in that the flow of heat from the glass equals the flow of heat into the room.

Gradient dissipation

The second law of thermodynamics states that if two systems are allowed to interact and exchange energy, that is if the constraints imposed between them are removed, then the systems will evolve to equilibrium, a new state in which we cannot differentiate between the systems. A statistical consequence of this physical law is that entropy will increase.

The active nature of the second law is intuitively easy to grasp and empirically easy to demonstrate. Figure 1 shows a glass of hot liquid placed in a room at a cooler temperature. The difference in temperatures in the glass-room system constitutes a potential and induces a flow of energy in the form of heat. This 'drain' on the potential flows from the glass (source) to the room (sink) until the potential is minimized (the entropy is maximized) and the liquid and the room are at the same temperature. At this point, all flows and thus all entropy production stops and the system is at thermodynamic equilibrium. The same principle applies to any system where any form of energy is out of equilibrium with its surroundings (e.g., whether mechanical, chemical, electrical or energy in the form of heat).

The second law alone does not tell which of the available

energy transfer paths the system will select in order to move back to equilibrium. The explanation to this can be provided in a classic experiment on self-organisation first devised by Henri Bénard in 1900 (Swenson and Turvey, 1991). A viscous fluid is held between a uniform heat source below and the cooler temperature of the air above. That is, there is a potential difference between fluid and air with a field force of a magnitude, F, determined by the difference between the two temperatures. When F is below a critical threshold heat flows from the source (fluid) to the sink (air) in the form of disordered collisions between the constituent molecules, and entropy is produced. If F exceeds the critical threshold Bénard 'cells' emerge spontaneously, each cell consisting of hundreds of millions of molecules moving collectively together in the form of rotating vertical convection columns. In this organised mode, the transfer of energy through the system and its dissipation to its surroundings is much more efficient than through unorganised collisions (Schneider and Kay, 1995). Such behaviour does not violate the second law. As long as a self-organising system produces entropy (minimises potentials) at a rate that is sufficient to compensate for its own ordering (persistence away from equilibrium) then the balance demanded by the equation of the second law is not violated (Kay, 1984; Swenson and Turvey, 1991).

Work

So far we have discussed displacement from equilibrium, constraint on energy transfer and gradient dissipation as distinct concepts describing the active nature of physical laws. But how can they be employed to control energy movement within systems, such that useful work could be extracted from their functioning (Jaynes, 1988)? Consider a system consisting of two connected tanks of equal volume but with different numbers of gas molecules. This difference defines a gradient between both tanks. As soon as a conduit between them is opened, gas whooshes through it, equalising the number of molecules in the tanks and erasing the gradient between them. Gas can rush through even if it has to turn a turbine along the way, thereby doing mechanical work. The energy to do that work came from the thermal energy of the environment, but the conversion from thermal to mechanical energy was paid for by the increase of disorder as the system equilibrated. Now, if we repeat the first process again by first closing the conduit and transferring energy from one tank to the other, we can repeat the same process of work extraction and gradient dissipation. Although simplified, this principle of work extraction constitutes a thermodynamic work cycle, which underpins the supply of most of the world's electric power and almost all motor vehicles.

Information

Within statistical mechanics, the entropy of a system at equilibrium can be recast in terms of the variety of microscopic states available to the system:

$$S \equiv k \ln \Omega \,,$$

where Ω is the number of states in which the system can be found when at equilibrium, and k is the Boltzmann constant, $1.38x10^{-16} J/K$. Consequently, entropy has been interpreted as a measure of macro-level disorder, formalised as Shannon entropy (Shannon, 1948) defined as:

$$S = -\sum p_i \log p_i,$$

where i ranges over the possible states of the system and p_i is the probability of finding the system in state i.

As such, it is possible to reinterpret the thermodynamic work cycle in information theoretic terms (Jaynes, 1988; Nelson, 2004). We have seen that the difference between doing work on a system and merely heating it up is the difference between how informed we are about the organisation of the system's energy. The potential gradient that must be established within a system before useful work can be extracted from it is thus also an informational property. Given that we are interested in computational systems that consume electricity and also process information, there is scope for the equivalences between information, energy and entropy to be useful, but also confusing.

Thermodynamics beyond physics

The application of thermodynamics is not limited only to physical systems (Jaynes, 1988). Ever since Alfred Lotka (1922) began writing about energy flows as the basis for natural selection, there has been a thermodynamic paradigm in evolutionary theory. Lotka observed that selection will favor those organisms that, in pulling resources into their own service, also increase the energy throughputs of their ecosystems (Wicken, 1989). What all organisms have in common is that they operate and evolve at some remove from thermodynamic equilibrium. By doing so they maintain the integrity of their organisational structures by irreversibly degrading free energy through informed kinetic pathways acquired through evolution. From this perspective, succession can be considered as the process by which an ecosystem moves away from thermodynamic equilibrium with its environment (Kay, 1984). By developing this account, the principles of variation and natural selection can be given a sound thermodynamic basis. The principle of variation derives from two sources: the entropic drive to generate configurational randomness and the quantum indeterminacy about where that randomness will occur. Natural selection follows from competition among alternative patterns of energy utilisation (Wicken, 1988).

One consequence of this perspective is an increasing appreciation that organisms can be viewed as more sophisticated 'engines' than the physical systems described so far

(Swenson, 1997). According to Kauffman (2000), for instance, life or its physical manifestation can be described in terms of an autonomous agent. This agent is a collectively autocatalytic system performing one or more thermodynamic work cycles that: (1) measures useful displacements from equilibrium from which work can be extracted; (2) discovers devices that couple to those energy sources such that work can be extracted; and (3) applies work to develop and maintain the constraints that enable the further extraction of work.

What are computational ecologies?

Figure 2 illustrates the architecture of a modern IT system. The infrastructure is an open system of interacting elements whose organisation is free to change and grow organically through the removal and introduction of components. Depending on the level of interpretation, these elements may be thought of either as physical servers or the software components hosted by them. However, these two levels of description encourage different thermodynamic accounts, and care must be taken in translating between them. While we may ultimately be interested in the physical time and energy required in order to achieve computational tasks, it has often been convenient to recast this problem in terms of efficient information processing with no explicit mention of energy, heat, etc.[1] However, while the 'motive force' driving the physical system stems from physical energy, the equivalent potential or gradient at the software level must be understood in terms of informational differences. Understanding how computation in such systems can be managed through self-organising mechanisms requires us to disentangle the physical system and software system levels.

For our purposes, the physical level of description can be stated rather straightforwardly. Autonomic computing systems are made up of a large-scale network of interconnected clusters of machines, each offering computational or storage resources. These functions are dependent on the constant supply of energy that is being fed to these machines in order to maintain their on-line functioning. The outcome of this energy consumption in the physical world is heat, generated in proportion to the intensity of computation. In an efficient system, this will in turn be related to system throughput, defined in terms of the number of computational tasks achieved per unit time.

By contrast, the software level of description, which will be our primary concern in the paper, is a little more complicated. We assume that the computational power offered by a system's physical servers constitutes a limited capacity raw resource, and that the efficient distribution of this resource to meet the needs of a set of users makes it natural to decentralise the control over their management to software elements (Kephart and Chess, 2003). This may be realised by applying a decentralised multi-agent architecture (Sycara, 1998) comprising a population of agents possessing their own thread of control and autonomy but perhaps lacking access to some central repository of system information. Figure 3 depicts the roles of software agents in managing system resources for such a system. The process is initiated by allocation requests from infrastructure users, U, illustrated by dashed arrows crossing the system boundary. Requests arrive in parallel and are intercepted by "user" or "consumer" software agents responsible for resource allocation (depicted as open circles). This incoming information 'agitates' the allocation process in resource consumers, inducing them to discover and select amongst available resource providers (represented as solid circles) until one agrees to execute the requested job (solid connecting line). In addition to executing the tasks for which they are currently configured, resource providers may also adapt to locally perceived demand by reconfiguring to offer the most demanded kinds of service.

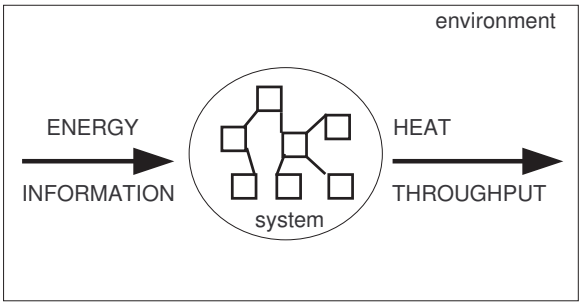


Figure 2: General architecture of the modern IT system.

It is important to note that the physics of the system imposes constraints on the software level. Allocation and reconfiguration decisions are only necessary as a consequence of the assumption that each resource provider is physically constrained such that it may only serve a limited number of consumers at the same time, and, furthermore, may only offer a limited set of services at any one time. Allocation and reconfiguration decisions must be efficient only as a consequence of the assumption that each interaction between agents, and each reconfiguration event incur associated physical costs in terms of time or power consumption.

The co-adaptation of resource providers and resource consumers takes place under conditions in which the demand for particular resource types may vary unpredictably. This requires providers to reconfigure their provision and consumers to track these reconfigurations. Consequently, the stability of the whole 'ecosystem' is dependent on the establishment of the information flow pathways that enable localised system elements to efficiently adapt and adjust their behaviours to the current system state. As the processing and

[1] However, heat management in both large-scale and micro-scale systems is a growing concern (Skadron et al., 2003; Sharma et al., 2005).

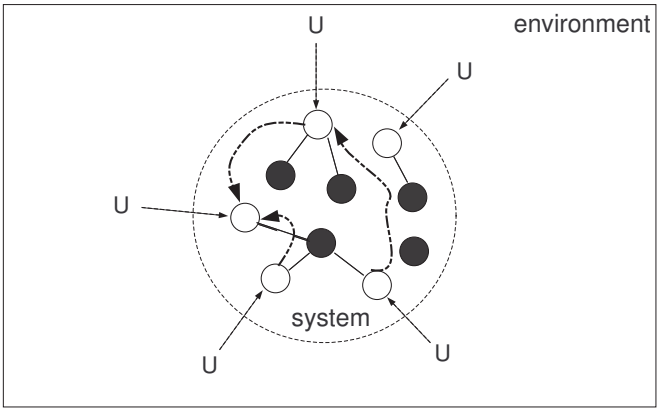


Figure 3: Resource allocation process conducted on the IT system software level.

propagation of information (represented by dotted arrows) is fully decentralised across the population of autonomic elements, understanding how this process may self-organise resource management is non-trivial. Nevertheless, there exists a range of studies focusing on information flows in different decentralised architectures demonstrating that effective decentralised control can be achieved if localised elements organise their information exchange (Packard, 1988; Guerin and Kunkle, 2004; Brueckner and Parunak, 2003). In the next section we will present a thermodynamical interpretation of this kind of self-organisation.

Energy, entropy and work in computational ecosystems

From the considerations outlined above, we might expect to find that the continued efficiency of self-organising computational ecosystems depend on them tending to establish and maintain information flows that bring about informational gradients that constrain agent behavioural choice. Under these constraints, agent behaviour results in efficient resource consumption, doing work for the user, but also re-establishes the constraints that enable the further extraction of work maintaining the system far from equilibrium. In what follows we will elaborate on this picture, focusing on the role of local information exchange.

Equilibrium

Each agent within a computational ecosystem may be characterised by its behavioural *repertoire*, the set of actions that are currently available to it. During each decision-cycle, an agent is required to select one action from the set of available ones and, by executing it, act upon its environment. The behaviour of an agent will exhibit the highest Shannon entropy when selection of any action is equally probable during each decision cycle, and the agent behaves randomly. Since the entropy of the whole population can be measured as the average over individual agent entropies, a multi-agent system can be said to be at equilibrium when all agent decisions are made at random.

Work

Whereas the establishment of an energy gradient is a precursor for useful work in a physical system, here it is useful to consider an *information gradient*. Recall that it is not the mere injection of energy that allows a physical system to perform work, but the *organisation* of this energy, which displaces the system from equilibrium. The same can be said of the distribution of information within a computational system. When one agent is informed such that it can be distinguished from the rest of the system, there is the potential for it to act in a manner that is constrained by its information, perhaps performing useful work. However, in a self-organising system, an agent's actions are liable to propagate information to other agents ensuring that informational disparities tend to be extinguished as they are exploited. Both energy and information flows are the result of local interactions between system components (molecules in a physical system; agents in a computational system) and, under the right conditions, both are 'motive forces' for achieving spontaneous system organisation.

Notice that information flow within a system constrains agent behaviour only if it creates a gradient between that agent and its surroundings. For instance, incoming information must perform work on agents rather than merely raising system temperature. While a constant supply of organised information is required to drive a system far from equilibrium, notice also that if the subsequent propagation of information between agents within the system is also organised, then agents can do work on (organise, inform) one another as they perform useful work for us, extinguishing their own potential gradient in the process. Recall the Bénard cells described earlier. There, molecules of a fluid, across which an energetic gradient is imposed, spontaneously organise each other such that they convey heat more efficiently than would be achieved by a random organisation of molecules.

Entropy

We have seen that information gradients may allow agents to make useful decisions, but that, for a system in flux, any collectively arising gradient informing agents about an available resource will eventually become 'dissipated'. That is, a flow of information about this resource will attract agents to consume it, extinguishing the original gradient, releasing constraints on agent behaviour and increasing system entropy. A computational ecosystem in which information propagates amongst agents is thus one in which there is a tendency for the system to equilibrate to an inefficient, essentially random state. However, it is precisely this tendency for information to propagate that can give rise to the possibility of efficient, persistent, self-organised behaviour.

Without information flows (of the right kind), agents cannot inform one another, organising or constraining each other's behaviour in a manner that is capable of achieving efficient work.

Case studies

Here, we provide three examples of decentralised system architectures (Parunak and Brueckner, 2001; Gambhir et al., 2004; Jacyno and Bullock, 2008), the functioning of which can be interpreted from the thermodynamical viewpoint outlined above. In each case, the local decision-making of individual system elements is achieved through the creation and destruction of gradients, work done on the system and by the system is manifested through the imposition and gradual release of behavioural constraints, and there is an important role for information flow.

Entropy in a two-agent system

A thermodynamic account of self-organisation within a multi-agent system is presented by Parunak and Brueckner (2001). The authors consider a simple coordination problem between two agents who desire to be together, one a mobile walker, the other in a fixed location. Both agents are embedded within a spatial environment with neither knowing the location of the other. The coordination problem for the walker is to locate the other agent and move towards it. An intelligent observer capable to seeing the state of both agents could send instructions to direct the movement of the walker. However, in this model Parunak and Brueckner investigate stigmergic coordination inspired by organisation in insect colonies. For this purpose, the stationary agent deposits pheromone molecules at its location. Initially, the walker is unable to sense any molecules and performs unguided movements. However, once pheromone molecules diffuse through the environment and are detected by the walker, it follows the gradient formed by them, thus reaching the target. We can understand how self-organised system behaviour emerges from the random processes of pheromone molecule diffusion on two levels: a macro-level at which coordinated behaviour of the walker agent arises; and a micro-level represented by a random motion of pheromone molecules that diffuse through the environment. An analysis of system organisation at both levels based on Shannon's entropy reveals that an increase in the micro-level entropy (as pheromone molecules diffuse to occupy an increasing number of locations) is accompanied with a decrease in entropy at the macro-level (as the movement of the walker is increasingly informed by the pheromone gradient).

This simple example illustrates not only how 'intelligent' behaviour emerges from a simple, entropy increasing processes, but also that the resulting self-organisation does not defy the second law of thermodynamics since the price paid for the entropy reduction at the macro system level is the increase in entropy generated by the random process that produces and maintains the gradient.

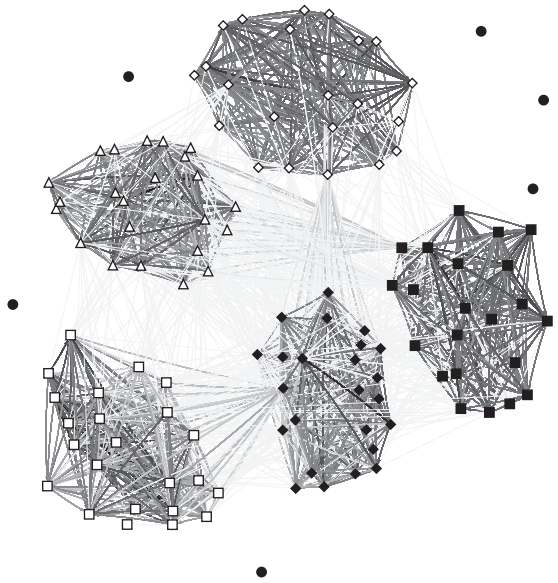


Figure 4: Communities of agents formed as a population of agents self-organise to reliably match consumption and provision of different resource types.

A full population model

A continuation of Parunak and Brueckner's work is presented by Gambhir et al. (2004). Here, the authors apply a computational model of an ant foraging system to demonstrate how complex organisation of interacting agents can be explained in terms of ideas from equilibrium and non-equilibrium thermodynamics. Their analysis of this classic example of self-organisation distinguishes three distinct modes of system behaviour: structure formation, structure maintenance and structure decay. During structure formation, some members of a population of agents diffusing over the environment discover a food source and establish a pheromone distribution instructing other agents to organise their activities into a foraging trail. By maintaining this structure, the population achieves reliable transport of food to the nest. Once the food source becomes depleted, the structure beings to decay and the agents return to their initial disorganised state.

To interpret how the system is displaced from equilibrium and how work is extracted from these conditions, the authors evoke ideas of unconstrained and constrained transfers of energy that are responsible for thermodynamical organisation and work extraction. Within a computational system, unconstrained flow of heat is considered as a diffu-

sive, entropy producing process of agents performing random walks. By contrast, constrained transfer of energy, in the form of interactions with an organised pheromone distribution, is interpreted as work done on agents, constraining their behavioural degrees of freedom (i.e., agent movements are directed to climb the pheromone intensity gradient, as in the case of the walker agent discussed above). The insights drawn from this model are similar to those arrived at by Parunak and Brueckner. An initial increase in entropy, during which agents explore state space, enables the formation of organisation, imposing constraints on agent behaviour through interaction with the pheromone field. To measure construction and destruction of constraints in this self-organising system, Shannon's entropy is applied. The measure of *useful work* done by the system is represented by the number of pieces of food taken from the food-source to nest over a run.

So far we have considered a population of agents that move from thermodynamic equilibrium to a constrained state and then back to equilibrium. However, in order to characterise a computational ecosystem that organises itself such that it remains far from equilibrium, in a dynamic, poised state where constraints are continually formed and released in a reflexive, self-perpetuating manner, we need to go a step further.

A self-organising computational ecosystem

Here, the system is an example of a computational ecosystem consisting of a population of consumer and provider agents responsible for reliable and efficient management of the resources offered by the system. Consumer agents belong to distinct groups, each characterised by the type of resource they are interested in allocating. Providers, on the other hand, are capable of offering any type of resource in general, but at any one time they are configured to offer only one type. As the interaction between agents and reconfiguration of offered resources has an associated cost (time), the efficiency of the system depends on discovering an organisation of agents that maximises the system's allocative throughput, at the same time avoiding unnecessary reconfiguration of providers or resource competition on the consumer side. This is achieved when consumers and providers self-organise into communities within which providers reliably offer a certain type of resource and consumers are biased towards selection of available providers (Jacyno and Bullock, 2008). Example communities are depicted in Figure 4.

Initially, the population of agents is uninformed and behaves randomly: consumers choose providers at random, and propagate information to one another at random. Organisation of the ecosystem into stable communities of agents is achieved through the formation and maintenance of information gradients between agents. These gradients are established through "gossiping", e.g., the local exchange of information about providers by individual consumers. As a consequence of sensing some gradient, an agent's initially unbiased selection of resources becomes constrained (work is done on the agent by the gradient). Agent behaviour is constrained in two ways. First, just as the agents in the previous case studies were able to exploit a pheromone gradient to discover food, here, consumer agents are constrained such that they tend to choose suitable providers. In doing so, they consume resource, and as a side-effect tend to dissipate the gradient that they were informed by. Second, the same gradient constrains agents such that they now tend to propagate information to a non-random sub-set of agents. By organising the information flows that propagate gossip such that agents form communities with shared interests, the system can maintain itself in a far from equilibrium organisation that allows useful work to be undertaken efficiently.

A complete analysis along these lines would clearly be more involved than in the previous examples, since here structure formation, maintenance and decay are ongoing processes that are capable of maintaining global system stability far from equilibrium. In particular, we have had to identify the manner in which the system organises the propagation of information in addition to merely establishing and releasing a constrain in order to achieve a piece of work. Here, we have attempted to lay some groundwork for further analysis of such systems by articulating the way in which thermodynamic ideas can offer a framework that focuses engineers on critical aspects of the system design.

Discussion and Conclusions

The aim of this paper is not to provide a ready-to-apply solution to the control of decentralised IT systems, but to point to and organise important work that has already been done in other research areas focusing on self-organisation and the homeostatic properties of natural systems. If we aim to engineer self-organising IT systems, we must understand the underlying thermodynamic principles of natural self-organisation, and, in particular, how to apply these principles in the context of open IT systems.

We have described how information disparity drives self-organisation in a population of software agents and that random behaviour is an integral part of the maintenance of information flows that allow such a population to organise effectively. This contrasts starkly with the (sometimes implicit) assumption present in the multi-agent system community that software agents share complete knowledge of the system, and make decisions as a result of joint deliberation, or at the behest of a central executive charged with deducing optimal behaviour. This approach is analogous to relying on a kind of maxwell demon to control a computational ecosystem. The demon knows the position and state of every element in the system and is able to impose/remove constraints that allow the system to do useful work. However, thermodynamic considerations imply that, even if such a demon

could be implemented, it would be extremely costly.

The interpretation provided here should not be considered exclusive. While thermodynamics and self-organisation have been the object of extensive research, there are still open questions with respect to the application of these ideas to systems that are far from equilibrium but capable of maintaining steady state (Kay, 1984). In such cases, considerations of thermodynamical systems at, close to, or moving towards their equilibrium state are insufficient, making far-from-equilibrium thermodynamics an open and active area of study with direct implications for engineering open computational ecosystems.

References

Brueckner, S. and Parunak, H. V. D. (2003). Self-organizing MANET management. In Marzo, G. D., Karageorgos, A., Rana, O. F., and Zambonelli, F., editors, *Engineering Self-Organising Systems*, pages 1–16. Springer.

Bullock, S. and Cliff, D. (2004). Complexity and emergent behaviour in ICT systems. Technical Report HP-2004-187, Hewlett-Packard Labs.

Gambhir, M., Guerin, S., Kauffman, S., and Kunkle, D. (2004). Steps toward a possible theory of organization. In A. Minai, Y. B.-Y., editor, *Proceedings of International Conference on Complex Systems*, page 9. W. H. Freeman.

Guerin, S. and Kunkle, D. (2004). Emergence of constraint in self-organizing systems. *Journal of Nonlinear Dynamics, Psychology, and Life Sciences*, 8:2:131–146.

Huberman, B. A. and Hogg, T. (1993). In Nadel, L. and Stein, D., editors, *Lectures in Complex Systems*, chapter The Emergence of Computational Ecologies, pages 185–205. Addison-Wesley.

Jacyno, M. and Bullock, S. (2008). Energy, entropy and work in computational ecosystems: A thermodynamic account. In *In Proceedings of the 11th Conference on Artificial Life (ALIFE 2008)*.

Jaynes, E. T. (1965). Gibbs vs Boltzmann entropies. *American Journal of Physics*, 33(5):620–630.

Jaynes, E. T. (1979). Where do we stand on maximum entropy? In Levine, R. D. and Tribus, M., editors, *The Maximum Entropy Formalism*, pages 15–127. MIT Press.

Jaynes, E. T. (1988). The evolution of Carnot's principle. In Erickson, G. J. and Smith, C. R., editors, *Maximum-Entropy and Bayesian Methods in Science and Engineering*, pages 267–284. Kluwer.

Kauffman, S. (2000). *Investigations*. Oxford University Press.

Kay, J. J. (1984). *Self-Organization In Living Systems*. PhD thesis, Deparment of Systems Design Engineering, University of Waterloo.

Kephart, J. O. and Chess, D. M. (2003). The vision of autonomic computing. *IEE Computer*, 36(1):41–50.

Lotka, A. J. (1922a). Contribution to the energetics of evolution. *PNAS USA*, 8:145–151.

Lotka, A. J. (1922b). Natural selection as a physical principle. *PNAS USA*, 8:151–154.

Nelson, P. (2004). *Biological Physics: Energy, Information, Life*. W. H. Freeman.

Nicolis, G. and Prigogine, I. (1977). *Self-organization in Non-equilibrium Systems: From Dissipative Structures to Order Through Fluctuations*. J. Wiley & Sons.

Packard, N. H. (1988). Adaptation toward the edge of chaos. In J.A. Kelso, A. M. and Shlesinger, M., editors, *Dynamic patterns in complex systems*, pages 293–301. World Scientific.

Parunak, H. V. D. and Brueckner, S. (2001). Entropy and self-organization in multi-agent systems. In J. Müller, E. Andre, S. S. C. F., editor, *Proceedings of the fifth international conference on Autonomous agents*, pages 124–130. ACM Press.

Parunak, H. V. D. and Brueckner, S. A. (2004). Engineering swarming systems. In Bergenti, F., Gleizes, M.-P., and Zambonelli, F., editors, *Methodologies and Software Engineering for Agent Systems*, pages 341–376. Kluwer.

Schneider, E. D. and Kay, J. J. (1995). Order from disorder: The thermodynamics of complexity in biology. In Murphy, M. P. and O'Neill, L., editors, *What Is Life: The Next Fifty Years. Reflections on the Future of Biology*, pages 161–172. Cambridge University Press.

Shannon, C. E. (1948). A mathematical theory of communication. *Bell System Technical Journal*, 27:379–423 and 623–656.

Sharma, R. K., Bash, C. E., Patel, C. D., Friedrich, R. J., and Chase, J. S. (2005). Balance of power: Dynamic thermal management of internet data centers. *IEEE Internet Computing*, 9(1):42–49.

Skadron, K., Stan, M. R., Huang, W., Velusamy, S., Sankaranarayanan, K., and Tarjan, D. (2003). Temperature-aware microarchitecture. In *Proceedings of the 30th International Symposium on Computer Architecture*, pages 2–13. IEEE Computer Society.

Swenson, R. (1997). Autocatakinetics, evolution, and the law of maximum entropy production: A principled foundation towards the study of human ecology. *Advances in Human Ecology*, 6:1–47.

Swenson, R. and Turvey, M. (1991). Thermodynamic reasons for perception-action cycles. *Ecological Psychology*, 3(4):317–348.

Sycara, K. (1998). Multi-agent systems. *AI Magazine*, 10(2):79–93.

Wicken, J. S. (1988). Evolution, thermodynamics, and information: Extending the darwinian program. *The Quarterly Review of Biology*, 63(1):84–85.

Wicken, J. S. (1989). Evolution and thermodynamics: The new paradigm. *Systems Research and Behavioral Science*, 6(3):181–186.

Zambonelli, F. and Parunak, H. V. D. (2004). Towards a paradigm change in computer science and software engineering: A synthesis. *The Knowledge Engineering Review*, 18(4):329–342.

Evolving Asynchronous Cellular Automata for Density Classification

Francis Jeanson

University of Sussex, Brighton, UK
f.jeanson@sussex.ac.uk

Abstract

This paper presents the comparative results of applying the same genetic algorithm (GA) for the evolution of both synchronous and randomly updated asynchronous cellular automata (CA) for the computationally emergent task of density classification. The present results indicate not only that these asynchronous CA evolve more quickly and consistently than their synchronous counterparts, but also that the best performing asynchronous CA find equally good solutions on average to the density classification task in fewer computational steps than synchronous CA.

Introduction

For the past 50 years cellular automata (CA) have established themselves as popular platforms to investigate complex phenomena. Their attractiveness stems in part from their ability to expose highly complex or even chaotic behaviour from an initially simple spatial configuration and set of update rules. An important insight that CA may provide is that in contrast to real world systems their dynamical laws are not bound by the classical laws of physics. Instead the laws that dictate the behaviour of a system are fully defined in terms of a state update policy which we may call Φ. Φ is defined by: the neighbourhood r of cells whose states causally impact the state of other cells, the rules that dictate *how* these neighbouring cell states impact other cells, and the global selection policy which specifies the set of cells that are to be updated by those rules. Traditionally work in this field has mostly been preoccupied in finding new sets of rules that give rise to interesting emergent phenomena. Indeed the general behaviour of a CA will exhibit a large diversity of dynamics with respect to rule updates. However, it is important not to omit the role of the neighbourhood and selection policy. The goal of this paper is to focus particularly on the latter by exposing the impact of the selection policy on evolved rules for cellular automata.

The most popular selection policy employed in CA research is synchrony. Here, all cells are selected and updated to their next state at each time step. Synchrony in cellular update implies that an automaton may potentially exploit the entire cell space to perform interesting global computations from local cellular interactions. This is possible because of an advantageous set of update rules found by the genetic algorithm (GA). A good set of rules that does in fact exploit this space efficiently at every time step is rare and ultimately difficult to find even via genetic search. In contrast an independent random updating selection policy where a single cell is updated at each time step does not allow an automaton to exploit space, since only a small region defined by the neighbourhood r of a single cell can be looked-up at any given time. This however may favour a genetic selection of rules that allow a CA to exploit time over space.

Asynchronous Computation

In contrast to synchronous cell state update where all cells of the automaton are updated in unison, asynchronous cellular automata (ACA) employ a selection policy whereby a single or a subset of cells are updated at a single time step. Independent random updating where a single cell is picked with uniform probability and updated over a single step is standard, although a number of alternatives have been explored (Schonfisch and de Roos 1999)[1]. In the past few decades ACA have been more carefully considered. Asynchronous update has been argued to be a more realistic approach in models of biologically inspired complex systems (Dellaert and Beer 1994; Harvey and Bossomaier 1997; Lee et al. 2007). Dellaert and Beer initially hypothesized that one of the main drawbacks from asynchrony comes from the difficulty due to indeterminacy in analyzing their behaviour. Furthermore this indeterminacy also seems to suggest that no general state attractor can be reached by asynchronous cellular automata. Harvey and Bossomaier challenge these worries and show that for random boolean networks (RBNs) asynchronous update may lead to a point attractor with a probability of $1/2^N$; where N is the number of nodes in the network (Harvey and Bossomaier 1997). They also show that loose attractors may be reached for the same type of update mechanism. Furthermore they intro-

[1]They distinguish step-driven from time-driven asynchronous updating. For instance a number of cells could be picked given a certain probability, in a particular sequence, or at a particular time.

duce practical methods for the analysis of indeterministic updating via probabilistic reasoning by careful inspection of node connectivity in RBNs. Interestingly experimental results by Harvey and Bossomaier indicate that the random node update with replacement method arrived more quickly at a point attractor in state space than did random update without replacement and synchronous update in RBNs. Lee, Adachi and Peper (2007) also propose asynchronous updating in two-dimensional cellular automata as a reliable, and biologically sound mechanism for self-replication. To them is appears that natural systems must have acquired tolerance to indeterministic interactions at the cellular level and that novel strategies have developed allowing the system to profit from this indeterminism. Kanada (1997) also emphasizes the significance of modeling ACA for real world applications and biological simulation. According to him synchronous emergent computation causes what he calls 'phantoms' which can be characterized as fragile system states. Slight disturbances from environmental noise will however prevent such systems from existing. His work on one-dimensional ACA shows that random or noisy interactions that naturally occur in these systems may play a positive role for our modeling and understanding of real world systems behaviour.

The current topic of research introduces a novel application of ACA by applying them to a computationally emergent task: density classification. Although, a priori, asynchronous automata have been considered to bear significance principally for the modeling of biologically inspired systems, the originality of their behaviour and ability to handle noise should give rise to interesting behaviour in a purely computational task. In fact, exploring the potential of asynchronous cellular updating in the well defined density classification task should eliminate any conceptual obscurities and inspire the potentially widespread significance of asynchrony in locally coordinated phenomena. For instance, we may think of the implications for the understanding of termite colony stigmergy (Grassé 1959, Beckers et al. 1994), or even for understanding synergistic effects of neuronal activity in large neural groups (Edelman 1993, Kelso 1995).

Synchronous Density Classification

Mitchell et. al (Mitchell, Hraber and Crutchfield 1993; Mitchell, Crutchfield and Das 1996) have thoroughly investigated the potential of genetic evolution for computational emergence in the density classification task. This task requires that a given one-dimensional cellular automaton determines the initial density that is most present in the initial cellular state within a number of update steps. By the end of these update iterations all the cells of the CA should be in the state identical to the state originally dominating the density of the CA. For instance given a 10 cell, two-state (0 or 1) CA, the 10 cells after K update steps should all be set to 1 if the original density of the CA contained more 1's than 0's - or they should all be set to 0 otherwise. Mitchell describes this problem as a task that the CA needs to accomplish by making use of local information for global coordination. In dynamical terms, a set of update rules of a CA's policy Φ must be found so that given any initial cellular state distribution the CA will follow a path in state space to a point attractor for that initial configuration. Mitchell et al.'s experiments have all been conducted using synchronous cellular update. By using a genetic algorithm to evolve the update rules of synchronous cellular automata (SCA) they have been able to obtain a diverse number of rules that are amongst the best known rules to date for the density classification task given any initial configuration density. Mitchell et al.'s ϕ_d rule has about 95% success rate in comparison the best known rule: the GKL rule with 97.8% (Gaks, Kurdyumov and Levin 1978). A few years later Land and Belew (1995) noted that they obtained from genetic evolution rules performing as well as GKL. In the same paper however they prove that no two-state CA can perform the density classification task perfectly.

In the following section I present results obtained by partially replicating the evolutionary mechanism employed by Mitchell et al. Because the purpose of the experiment was not aimed at discovering better rules I conducted 30 runs on each experiment instead of the 100 that Mitchell et al. examined in order to save computational time for the evolution of asynchronous rules. Exactly the same genetic algorithm was used to evolve rules in both the synchronous and the asynchronous scenario. In contrast to the synchronous scenario, asynchronous updating is performed by independent random selection at a single time step of a single cell which is then updated according to the rule table for that CA.

Evolving Update Rules

In their initial experiments Mitchell, Hraber and Crutchfield (1993) evolve one-dimensional cellular automata with lattice size $N = 149$. This ensures that the initial configuration always contains either a majority of 0's or a majority of 1's. The chromosomes evolved were binary strings encoding the rule outputs for a given CA. These update rules consider a neighbourhood radius $r = 3$ as can be seen in figure 1; hence the total number of cells lookup for the update of a particular cell is $2r + 1 = 7$: the three cells to the left, the three cells to the right and the current cell itself[2]. Hence given that each cell has either binary state 0 or 1, the total number of possible update rules is $2^7 = 128$. Thus a chromosome is represented as a binary strings of length 128. This implies that the search space in which the genetic algorithm must find good solutions to the density classification task is of size 2^{128} - too large for any brute force search. A single run of their GA consisted of evolving a set of 100 update rules over 100 generations. At each generation 100 new

[2]Rule lookup wrap arround the CA for beyond boundary conditions.

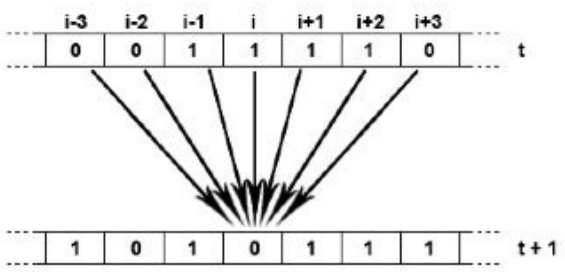


Figure 1: Illustration of cell state transition from step t to step $t + 1$ according to an arbitrary update rule. With r = 3, three neighbours to the left and to the right of the current cell i are looked-up to determine cell i's subsequent state.

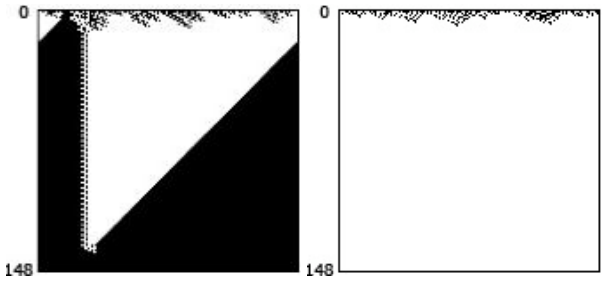


Figure 2: Sample iteration of a successful 1D-SCA in density classification. On the left the rule solving for a greater density of 0s; and the right the same rule for a greater density of 1s.

initial configurations with a biased distribution were created. This biased distribution ensures that there is a uniform distribution over the initial 100 densities. According to them the biased distribution allowed for the GA to find increasingly better solutions to the problem much more easily than if all the initial configurations had a probability distribution of 1/2, which rendered the task almost intractable. Each of the 100 rules are tested on each of these initial configurations over a fixed number $K = 149$ update steps. After the 100 rules have been tested the top 20 rules were selected from the population and copied to the next generation. Out of these 20 elite rules two were picked with replacement at random. Single point crossover was performed between these two rules after which mutation at exactly two loci was applied for each new offspring. In order to pick the best rules for the subsequent generation each rule was given a score corresponding to the number of correct density classifications it accomplished over the 100 initial configurations. The GA in the present experiment on synchronous updating was implemented exactly as described above. Mitchell et al. however collected their results after 300 runs. Because my aim was to compare the dynamics of synchronous updating with respect to asynchronous updating I opted for a more economical set of 30 runs per cellular configuration, in a total of 6 configurations. Initial test runs showed however that rule populations would often get stuck in very low minima. Yet because each configuration was meant to be evaluated over 30 runs only, I decided to add more noise to the mutation rate by allowing for a random mutation to occur at any loci with double probability, i.e. 1/64 for each 128 rule outputs.

The GA used for finding rules in asynchronous updating is identical to the algorithm for synchronous updating. As opposed to the SCA where all cells are updated according to the state of their neighbouring cells at the same time, the ACA implemented here selects a single cell with uniform random probability with replacement. Hence a cell has a probability of $1/N$ of being selected for update at every step while the remaining cells stay unchanged.

Finding Good Solutions Reliably

After running this evolutionary algorithm on 30 runs for SCA, I obtained by the final generations a majority of rules that would correctly classify the density of a given initial configuration about 50% of the time. These results are similar to those obtained by Mitchell et al. This would suggest that most runs found rules that could classify for almost any initial density whether the initial configuration contained more 1s than 0s or vice versa, but not both. All runs typically start with a set of poor performing rules. Out of the 30 runs, 1 run failed to evolved any rule that would correctly classify over 5% of the time; yet 7 runs succeeded in finding rules that classified with a success rate of 97% or higher. Although at first glance this seems to suggest that rules better than GKL were found, it is important not to forget that these rules where only tested on a set of 100 biased initial configurations. Further testing of these rules on larger set of initial configurations would provide a more accurate idea of a rule's actual performance. This is beyond the scope of the present paper however. Figure 2 illustrates the cellular progression of one of the best rules in synchronous updating. From this figure we notice that the evolved rules find a solution quickly for a greater initial density of 1s but takes much longer when the initial configuration contains more 0s. This was true for all the highest performing rules evolved for this SCA, and suggests that although a large number of SCA rules are capable of correctly classifying higher densities of 1s or 0s, a special strategy must evolve to successfully classify the opposite state density - we may call this 'density preference'. It is interesting to note that these rules highly resemble a specific type of rule found by Mitchell et al. which they call ϕ_a. This rule performs what they call block expansion to solve the task (Mitchell 1998).

Hence it isn't trivial for the GA to find good solutions to the density classification task in SCA. It seems that a satisfactory point attractor is only reached for half of the initial conditions on a large majority of the runs. In contrast however asynchronous updating gave interesting results in other dynamical systems. Harvey and Bossomaier for instance,

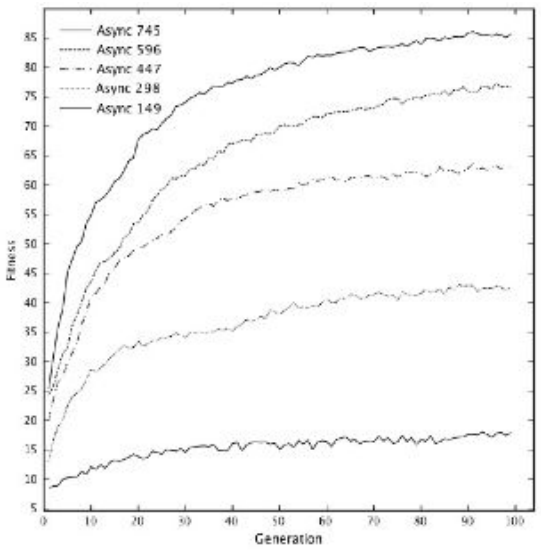


Figure 3: Average fitness of ACA for $K = 149, K = 298, K = 447, K = 596$, and $K = 745$ over 100 generations in 30 runs.

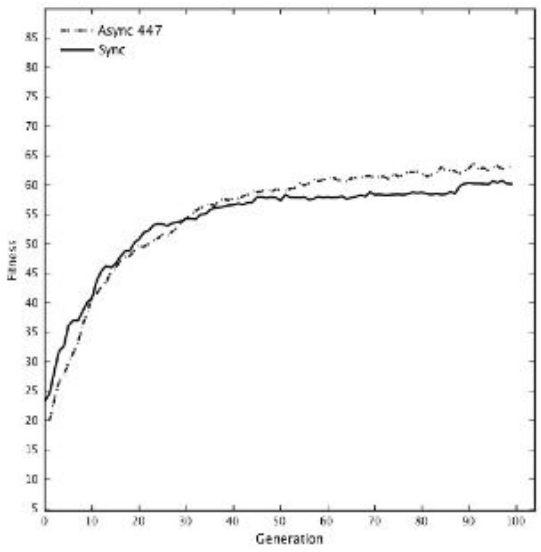


Figure 4: Average fitness of SCA, in comparison to ACA with $K = 447$ over 100 generations in 30 runs.

noticed that asynchronous RBNs tend to arrive a point attractors more quickly than synchronous updating. It should be interesting then to investigate this phenomenon in ACA for the density classification task.

Because ACA require that only a single cell be updated at every step, they require N times less computation over a given number of steps K. It was predicted then that a greater number of update steps would be required to obtain well performing rules in the density classification task with asynchrony. For this reason I conducted 30 runs on 5 different ACA configurations. The first configuration held the K number of update steps to 149. I'll refer to this scenario as Async 149. For the second configuration I decided to double the number of K steps in an ACA's computation: Async 298, and run the GA 30 times. Following the same procedure I ran the evolutionary algorithm on Async 447, Async 596, and Async 745, each extensions of Async 149 by factors of 3, 4, and 5 respectively.

As expected Async 149 did not find any high performing rules for the density classification task. The GA did manage to find a number of rules for Async 149 that correctly classified the CA up to about 22% of the time. These more easily solvable cases stem from conditions where initial configurations had highly biased densities. As factors in the number of update steps K increased, one notices an almost linear increase in performance at first. Evolving Async 298 gave rise to some rules reaching a success rate nearing 45%. Async 447 provided consistent rules surpassing the 50% mark, with a number of rules reaching rates of success of 68%, with an average of about 63%. However this progressive linear increase halted with $K = 596$. Evolving Async 596 indeed rarely gave rules with results any better than 80%, although 75% was consistently reached by at least half the rules after 100 generations. The increase to $K = 745$ confirmed this sudden decrease in rate of improvement, with top rules averaging the 86% mark. This diminishing rate of improvement suggests that there exists a non linear increase in the complexity for finding good rules that solve initial configurations with highly even state densities. Figure 3 illustrates the progression in fitness for each type of scenario over 100 generations.

In comparison the average success of rules discovered for SCA reaches roughly 60% (Figure 4) which happens to perform *worse* than Async 447 after 100 generations. However it is important not to forget that although the average success is relatively low, evolving SCA did give rise to the highly performing rules discussed above with rates nearing 100% success. This indicate then that SCA is prone to a much higher deviation than its ACA counterparts. We notice form Figure 5 that the standard deviation of evolved rules for asynchronous update increases as the number of steps K increases. This can be explained by the increasing specialization of a set of rules in the population. In other words, better performing rule sets have greater opportunity to 'prove themselves' when given more time to accomplish the task. We also notice from Figure 5. that the standard deviation of each ACA begins by increasing over the first generation quarter, but then all deviations progressively diminish.

The decrease in standard deviation in fitness means that rules that are more fit are found more consistently and provide increasingly similar performance. Hence evolving ACA gives rise to an increasingly reliable set of rules for the

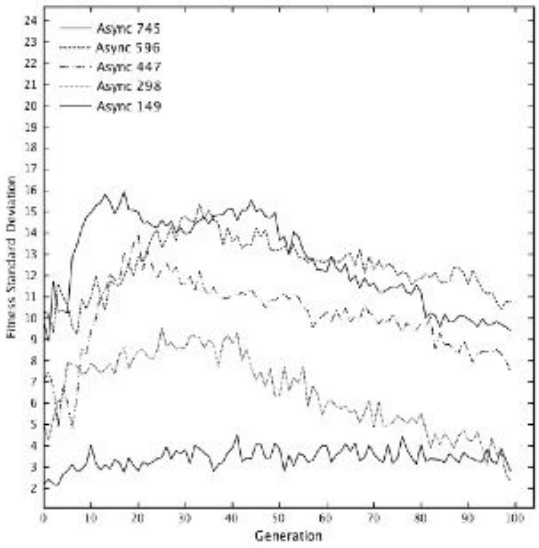


Figure 5: Standard deviation of rule fitness for Async 149, Async 298, Async 447, Async 596 and Async 745, over 100 generations.

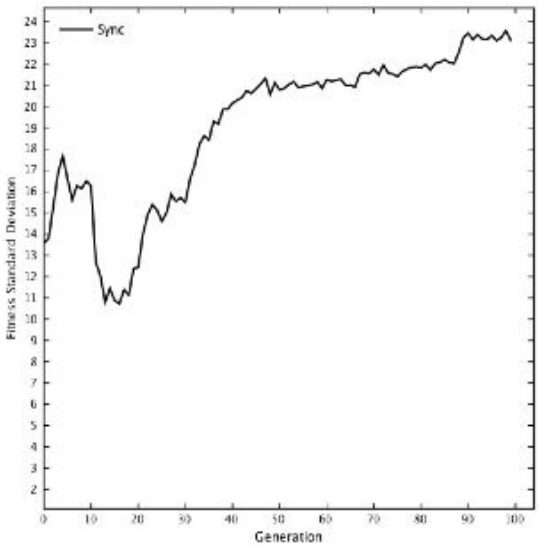


Figure 6: Standard deviation of rule fitness for SCA over 100 generations.

given task. A few hypotheses may begin to explain these results. Perhaps the GA evolves at every run a set of rules that are increasingly similar genetically as generations go by. This would inevitably cause a narrowing pool of rules that the GA always succeeds in finding. This hypothesis however doesn't fit the results which show that there is an initial increase of the standard deviation. A hypothesis that I consider more probable is that the GA manages to narrow down the pool of successful rules only after generating first a highly diverse pool of rules in which some perform very well and other quite poorly, after which a sort of population neutrality appears then to form by mutation and crossover which allows for this increasing and reliable specialization of the rule population after the first generation quarter.

In contrast the evolved SCA are increasingly more volatile as generations progress. Figure 6 not only shows a much higher deviation for SCA than for ACA but how this deviation increases after each generation. As mentionned earlier this high deviation is explained by the majority of rules which get stuck in local optima by achieving the correct classification of initial configurations for one of two states only, i.e. density preference. The few cases that find on the one hand high performing rules or on the other hand very poor rules will inevitably cause this strong deviation. In comparison ACA do not seem to suffer from this density preference (Figure 7). From this data then it is clear that evolving ACA provides good rules much more reliably than does the evolution of SCA for this task.

Finding Good Solutions Rapidly

Synchronous update implies that all cells are updated in unison, however as seen in Figure 4 SCA rules that perform well at the density classification task are sparse. Yet for ACA well performing rules are much easier to find for the GA. Figure 4 even shows how Async 447 has rules performing better than SCA on average. However Async 447 performs single cell update at every step. Thus after 447 update steps it has performed exactly $1 * K = 447$ cell updates. In comparison SCA perform $N * K = 149^2$ cell updates. At first glance this appears to mean that Async 447 found on average better solutions than SCA with about 50 times less computation. This isn't quite exact however. I mentioned earlier that evolved SCA rules will on average solve half of the initial configurations within half a dozen steps due to density preference. Solving the other half of initial configurations though will typically take at least 100 update steps. By averaging for both possible initial configurations is it reasonable to assume that on average roughly 55 steps are required to solve the density classification task when $N = 149$ with the best evolved rules. This means that most rules for SCA require about $55 * 149$ cell updates which still represents 18 times more computation then that required by Async 447. Overall this suggests that ACA can in fact perform faster than SCA at this task given that less perfect target rules are

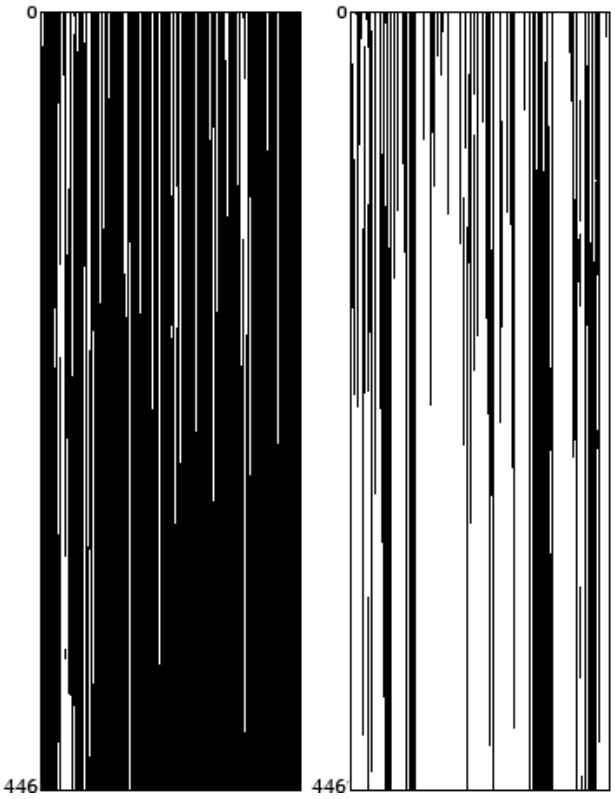


Figure 7: Sample iteration of a 1D-ACA in density classification. On the left the rule solving for a greater density of 0s; and the right the same rule for a greater density of 1s. Although perfect decisions are not made for these more difficult initial configurations, it is easy to see how the left ACA progressively eliminates cells in state 1. And vice versa for the ACA to the right.

required by a system or user exploiting these dynamics.

Conclusion

From the results collected here asynchronous cellular update in one-dimension automata may exhibit important computational qualities. Essentially these results first show that a genetic algorithm can find high performing update rules for ACA more reliably than for SCA. I've shown that rules for independent random updating of cell states to classify initial densities are found more consistently than when the updating is synchronous with similar success rates. This was particularly made evident by a different standard deviation between both update methods: Whereas SCA show consistent overall increasing deviation as rule generations go by, ACA show initial increase during the first quarter followed by significant decrease in standard deviation of fitness. Because the standard deviation of ACA isn't simply lower than that of SCA but actually decreases suggests that a quite radically different phenomenon is taking place when finding rules for asynchronous updating via genetic algorithms. Interestingly the evolved rules for ACA do not appear to suffer from density preference as do SCA. It is suspected that this phenomena is intimately tied to the reliability of ACA for finding good solutions. It appears then that as a tradeoff from exploiting space to find high performing rules, synchronous update renders the computation of a CA highly prone to instability. In contrast asynchronous update policies seem to allow for more robust activity by exploiting time, while providing sufficiently good performance.

Furthermore the results suggest that ACA can decide with much less computation the density of an initial configuration than SCA if this density is relatively biased. On average ACA performed about 18 times faster (computing time) than SCA with similar results. This concurs with the idea that although ACA may take more time (update steps) to arrive at a point attractor they require much less overall computation (computing time) than do SCA. This agrees with precedent results found by Harvey and Bossomaier (1997) on dynamics of asynchronous random boolean networks. Further work should be conducted to examine more precisely the threshold in update steps at which ACA find high performing rules for density classification. Also, a better understanding of how ACA exploit the state space should be developed.

The choice of density classification as a nontrivial task for the global arrangement of cell states from local interactions is proposed here as a simple yet well defined problem for exploring the potential of asynchronous updating in complex dynamics. Because CA behaviours are fundamentally dictated by their update policy Φ, it is reasonable and perhaps useful to regard Φ as the underlying 'physics' of these systems. The spatially distributed nature of cells and their update over time motivates the use of CA for real world models of global dynamics from local interactions. Hence the results obtained herein could potentially contribute to the better understanding of complex dynamics in natural phenomena. Arguably, dynamical properties of asynchronous cell selection may give insight into temporally dissociated interactions such as in chemical reactions, neural group activity, population dynamics etc. The two observed advantages of asynchronous random cell updating in the present experiments (reliability and rapidity) have quite distinct implications. Although both aspects may be practically exploited for engineering prospects, the reliability characteristic of asynchrony in the context of natural phenomena relates purely to the 'availability' of the underlying physics (update rules) which give rise to the behaviour of interest. Here, results imply that under conditions of asynchronous random cell selection these rules are more readily available for density classification from genetic search. Although specific to this task, such flexibility could speculatively be shared by other natural phenomena as mentioned above. The second aspect - which shows that density classification is obtained

more rapidly in ACA than SCA on average - may predict, however, that convergence towards stable attractors in natural temporally dissociated phenomena is likely to occur with higher frequency. This, of course, is contingent upon the fact that the dynamics of the present task can be extrapolated to other real world phenomena.

References

Beckers, R., Holland, O.E., Deneubourg, J.L, (1994). From local actions to global tasks: stigmergy and collective robotics. In P. Maes and R. Brooks, editors, *Artificial Life IV*, pages 181-189. MIT Press, Cambridge, MA.

Dellaert, F., Beer, R. (1994). Towards an evolvable model of development for autonomous agent synthesis. In P. Maes and R. Brooks, editors, *Artificial Life IV*, pages 246-257. MIT Press, Cambridge, MA.

Edelman, G. (1993). Neural darwinism: Selection and reentrant signaling in higher brain function. *Neuron*, 10:115-125.

Gaks, P., Kudryumov, G.L., Levin, L.A. (1978). One-dimentional uniform arrays that wash out finite islands. *Probl. Peredachi Inform.*, 14:92-98.

Grassé, P.P. (1959). La reconstruction du nid et les coordinations inter-individuelles chez *Bellicositermes natalensis* et *Cubitermes* sp. La theorie de la stigmergie: Essai d'interpretation des termites constructeurs. *Ins. Soc.*, 6:41-83.

Harvey, I., Bossomaier, T. (1997). Time out of joint: Attractors in asynchronous random boolean networks. *ECAL97*.

Kanada, Y. (1997). Asynchronous 1D cellular automata and the effects of fluctuation and randomness. Extended paper from the Artificial Life IV Poster.

Kelso, J.A.S. (1995). *Dymanic Patterns*, pages 37-43. MIT Press, Cambridge, MA.

Mitchell, M. (1998). *An Introduction to Genetic Algorithms*, pages 44-55. MIT Press, Cambridge, MA.

Mitchell, M., Hraber, P.T., Crutchfield, J.P. (1993). Revisiting the edge of chaos: Evolving cellular automata to perform computations. *Complex Systems* 7:89-130.

Mitchell M., Crutchfield, J.P., Das, R. (1996). Evolving cellular automata with genetic algorithms: a review of recent work. In Proceedings of the First International Conference on Evolutionary Computation and Its Applications *EvCA'96*. Moscow, Russia: Russian Academy of Sciences.

Land, M., Belew, R.K. (1995). No perfect two-state cellular automata for density classification exists. *Phys. Rev. Lett.* 74: 5148-5150.

Lee, J., Adachi, S., Peper, F. (2007). Reliable self-replicating machines in asynchronous cellular automata. *Artificial Life*, 13:397-413

Schonfisch, B., de Roos, A. (1999). Synchronous and asynchronous updating in cellular automata. *BioSystems*, 51:123-143.

Effects of Signalling on the Evolution of Gene Regulatory Networks

Dafyd J. Jenkins and Dov J. Stekel

Centre for Systems Biology, School of Biosciences, University of Birmingham, U.K., B15 2TT
djj134@bham.ac.uk d.j.stekel@bham.ac.uk

Abstract

We investigate whether observed transcription network structures and network motifs are a byproduct of the mechanisms by which DNA strands evolve, or if they are fundamental to the function of the network. We explore this with an evolutionary model with stochastic Boolean network simulation. Structurally distinct regulation strategies are observed in populations evolved with and without internal energy signalling. However, food signalling is not used in either population in the case when the food supply itself is constant. Parallels between the evolved networks and CRP-cAMP regulation in *Escherichia coli* and the endosymbiont *Buchnera aphidcola* are presented and discussed. Comparing the evolved networks with neutrally evolved populations indicates that networks evolve to lose most regulatory activity, due to loss of binding sites and transcription factor activity, including losing global regulation mechanisms.

Introduction

Transcription regulation and cell-signalling networks have been studied extensively; recently high-throughput 'omics' technologies have provided a wealth of data, including genome sequences, gene-expression, metabolism and protein-protein interaction profiles. Systems biology attempts to use these data to develop models for reconstructing and analysing transcription, signalling and other networks. Local analysis determines the function of over-abundant network motifs (Milo et al. (2002)), such as the 'feed-forward loop' (FFL), which can function as a low-pass filter (Mangan and Alon (2003)). Network motifs are a valuable tool for analysing transcription regulation networks, but do not always indicate dynamical behaviour: the bi-fan motif exhibits wide ranging and non-characteristic behaviour when modelled using biologically plausible parameters (Ingram et al. (2006)) and the process of DNA replication can also cause the over-abundance of motifs (van Noort et al. (2004)). Global analysis, such as node degree, of entire transcriptional networks has indicated an approximately scale-free out-degree distribution and an exponential in-degree distribution in both prokaryotic and eukaryotic organisms (Albert (2005)).

The use of energy signals in biological regulatory networks is well studied. The transcriptional regulator complex CRP-cAMP is one of *Escherichia coli*'s global regulators, known to regulate several hundred genes as listed in the EcoCyc database (Karp et al. (2007)). The large number of positive interactions by CRP-cAMP in biosynthesis pathways indicates that energy signals are used for growth by cells (Zheng et al. (2004), Hardiman et al. (2007)). A subunit of the CRP-cAMP complex, cAMP, is a signalling molecule derived from ATP; ATP concentration indicates 'energy' within the cell. When the concentrations of CRP and cAMP reach sufficient levels, the activated transcription factor complex forms. Whilst CRP-cAMP is a dual-regulator (activation and repression), 142 of the 173 known and predicted interactions in the EcoCyc database are identified as activating interactions.

Organisms without energy signalling are also prevalent in nature. *Buchnera aphidcola* is a bacterium related to *E. coli*, having a common ancestor diverging 250 million years ago (Moran and Mira (2001), Shigenobu et al. (2000)). *B. aphidcola* has a different lifestyle to *E. coli*; it has evolved an endosymbiotic relation with aphids, while *E. coli* exists as a free-living bacterium. *B. aphidcola* cells live in an environment of sufficient food, which is simpler than many other bacterial environments. *B. aphidcola* strains have lost most of their genome and regulatory network, retaining around 600 genes, representing a subset of *E. coli* genomes (Shigenobu et al. (2000), Wilcox et al. (2003)). This lack of regulation allows the over production of several amino acids, which are excreted and subsequently used by the aphid. The lack of an 'energy signal' observed in *B. aphidcola* is due to the absence of *crp* and *cycA*, the genes responsible for the CRP-cAMP transcription factor (Shigenobu et al. (2000)).

Many computational models exist for evolving transcription network structure, such as the Artificial Genome (Quayle and Bullock (2006)) and Artificial Regulatory Network (Kuo et al. (2006)), which are capable of evolving very realistic structure. However, the behaviours evolved are often arbitrary and non-realistic, such as matching a specific pattern of expression. Models also typically omit en-

ergy usage, which is a fundamental requirement for transcription regulation. Stochasticity, whilst has been shown to have substantial effects on gene expression in biological cells (Elowitz et al. (2002)), is also often omitted from models of gene regulation networks.

We investigate the effects of dynamics in the evolution of transcription network structure using models with and without energy signalling. We introduce a model that evolves networks using realistic evolutionary operators and is simulated with simple inputs and output to determine fitness. Our model introduces regulation type for binding sites, new evolutionary operators, signalling mechanisms as inputs and biosynthesis as output. We simulate the networks using a stochastic Boolean network paradigm, representing simplified transcriptional network dynamics. The results of these evolutions are presented and analysed, and relevance to biological systems is discussed. Graph theoretic approaches are used to compare the directed evolution and networks that have evolved neutrally over the same time period, highlighting the effects of the directed evolution.

The exploratory results presented in this paper highlight the potential insights into evolutionary behaviour that can be obtained using simple, yet biologically realistic models.

Method

The model has two distinct components: 1) network generation and static structure and 2) network simulation, dynamics and evolution.

Network Generation

To generate the gene regulatory network, we use the model introduced by van Noort et al. (2004) and extended by Cordero and Hogeweg (2006). This model produces a network with realistic connectivity and structure of specific protein-DNA binding interactions when evolved without a fitness function, "neutral evolution". A genome initially consists of N regulatory genes, where each gene has a regulatory region with between 0 and I binding sites, bs, and a protein, p. Each binding site and protein has a specific shape, S, represented by an integer drawn from a discrete circular space $\{0, 1, 2, ..., S_{max} - 1\}$ (with $S_{max} - 1$ adjacent to 0). The binding strength, B_{ij}, between two shapes, S_i and S_j is defined as:

$$B_{ij} = \begin{cases} 1/(D_{ij} + 1) & if\ D_{ij} \leq D_{max} \\ 0 & otherwise \end{cases}$$

where D_{ij} is the shortest integer distance between the shape of the protein, S_i, and the binding site, S_j. A binding distance, D_{max}, is defined as the maximum distance between two shapes that will interact. A matrix, M, is created where M_{ij} is the strength of binding B between protein i and binding site j. From this matrix, the network connectivity can be visualised and analysed. The binding strength, B, between a protein and binding site is used during network simulation.

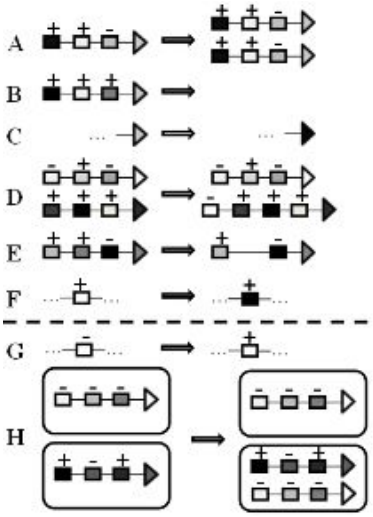


Figure 1: Evolutionary operators. Rectangles represent binding sites (+ activating; - repressing), triangles represent gene/protein product. Shape is represented by greyscale colour. Original operators defined in Cordero and Hogeweg (2006) are shown in parts A - F. A) gene duplication, B) gene loss, C) protein mutation, D) binding site duplication, E) binding site loss and F) binding site mutation. The two evolutionary operators introduced in this work, G) binding site regulation 'flip' and H) horizontal gene transfer. Operators A - G apply to each gene or regulatory region within a genome, whereas H only applies to the whole genome level.

In addition, our model introduces two types of regulation for each binding site: positive, bs^+; and negative, bs^-. Thus, as in real gene regulatory networks, binding sites can either increase the rate of a gene's transcription (positive regulation) or decrease transcription (negative regulation). Figure 2 shows an example network and interactions.

Specialised Genes In addition to the regulatory genes in the original models, we introduce three new types of genes:
Energy signal genes: these genes have a protein product, but no regulatory region. The expression status is based on the amount of energy within the cell. Energy in the model abstractly represents the ATP, amino acids and other molecules a biological cell requires to grow, transcribe mRNA molecules and translate them into protein molecules and other processes.
Food signal genes: these genes represent the food available to the cell and are used as the input into the model when it is simulated. they have a protein product, but no regulatory region. The energy level of the model increases whenever a *food signal gene* is activated. Each *food signal gene* has an energy value associated with it, which is the amount of energy added to the model when the gene is activated.
Biomass pathway genes: these genes have a regulatory re-

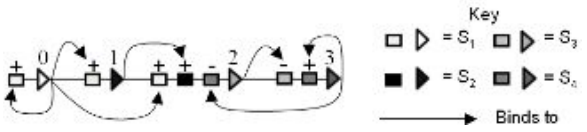


Figure 2: An example 4-gene network showing protein-DNA interactions. Genes 0,1 and 2 form a type-1 coherent feed-forward loop (FFL). Additionally, gene 0 has an activating self-regulating connection. A fourth gene in the circuit acts as an AND gate in the FFL, by negatively regulating gene 2. If gene 3 is transcribed, it negatively regulates the FFL, and causes the FFL to be an AND gate. If gene 3 is not present, then then the FFL will be OR gate.

gion, and generate *biomass* when expressed. They are used as the output of the model when it is simulated, represent cell growth, and have both an energy consumption (amount of energy used when gene is activated) and biomass production (amount of biomass added when activated) value associated with them.

Neutral Evolution and Evolutionary Operators

Once the network has been initialised, it is neutrally evolved for a given number of steps by randomly selecting a gene from the genome and applying a mutation operator. Cordero and Hogeweg define six mutation operators which operate at either the gene or binding site level: 1) gene duplication: the entire gene (protein product and regulatory region) is copied and added to the genome, producing an exact replica of the original gene, 2) gene loss: the entire gene is removed from the genome, 3) protein mutation: the protein shape is changed, 4) binding site duplication: a binding site from another gene is randomly copied into the regulatory region, 5) binding site loss: the binding site is removed from the regulatory region, 6) binding site mutation: the binding site shape is changed.

Shape mutation (protein and binding site) in the Cordero and Hogeweg model consists of either incrementing or decrementing the shape, S, by 1 with equal probability. We use a more realistic mutation operator allowing the shape to make larger jumps around the shape space, using the integer part of a normal random variable with $\mu = 0$ and $\sigma = log_{10}S_{max}$.

We define two new evolutionary operators: 7) binding site regulation 'flip': the binding site 'flips' its regulation type from positive to negative or vice versa, 8) horizontal gene transfer (HGT): a portion of another genome is horizontally transferred and copied into the genome (corresponding to DNA-uptake or plasmid transfer). This operator is applied at the genome level only.

Due to the specific function of the *energy*, *food* and *biomass* genes, not all evolutionary operators are applied to them. The evolutionary operators applied to *energy signal genes* and *food signal genes* are 1) gene duplication (however, the duplicated gene loses the specialised functionality as the original) and 3) protein mutation. The evolutionary operators applied to *biomass pathway genes* are 1) gene duplication (functionality not duplicated), 3) protein mutation, 4) binding site duplication, 5) binding site loss, 6) binding site mutation and 7) binding site regulation 'flip'.
All mutation rates (and other model parameters) are given in Table 1.

Network Simulation and Dynamics

In order to further investigate the structure of the networks evolved using realistic evolutionary operators, we introduce a simulation system for examining the dynamics of the networks. We use a Boolean network model (Kauffman (1969)) to simulate the dynamics of the network over a number of discrete time steps. Stochasticity is added to the simulation with random, basal levels of transcription. At each time step a number of steps takes place in order:

1. Energy signal gene status (ON if energy threshold is exceeded, OFF otherwise) and food signal gene status (ON if food available this time step, OFF otherwise) is determined.
2. Determine protein-DNA interactions for all ON genes.
3. Determine gene activation status.
4. Update energy and biomass levels.
5. All bound binding sites unbind (all binding sites are OFF).
6. Check model has energy remaining - if the energy level is ≤ 0 then the model 'dies' due to lack of energy, and simulation terminates.

where ON = 1 and OFF = 0. All genes are OFF initially.

Protein-DNA Interaction Protein-DNA interactions are determined by the following logic equation:

$$binding_status_{ij} = (B_{ij} \times gene_status_i) > R$$

where B_{ij} is the binding strength between protein i and binding site j, $gene_status_i$ is the activated status of gene i (is the gene transcribing/translating) and R is a random number between 0 (inclusive) and 1 (exclusive).
The resultant matrix indicates binding site occupancy.

Gene Activation Status Gene activation status is determined by:

$$gene_status_i = \begin{cases} 1 & if\ f(x) > 1 \\ & or\ f(x) = 0\ \&\ K_{basal} > R \\ 0 & otherwise \end{cases}$$

$$f(x) = \sum_{a=1}^{A} G_i bs_a^+ - \sum_{b=1}^{B} G_i bs_b^-$$

Parameter	Value	Note
S_{max}	128	
D_{max}	3	
Starting genome size	$32,256$	
Max. starting binding sites/gene	3	
Initial mutations	2000	
Gene duplication	1×10^{-3}	[1]
Gene loss	1×10^{-3}	[1]
Protein mutation	5×10^{-3}	[1]
Binding site duplication	8×10^{-3}	[1]
Binding site loss	8×10^{-3}	[1]
Binding site mutation	8×10^{-4}	[1]
Binding site 'flip'	8×10^{-4}	
Horizontal gene transfer	5×10^{-5}	
Max. genes horizontally transferred	10	
Basal transcription rate, K_{basal}	1×10^{-2}	
Binding threshold, T_{bind}	0.5	
Population size	1000	
Generations	100	
Simulation time steps	1000	
Starting energy	500	
Energy signal gene threshold	250	
Food gene energy generated	5	
Biomass gene energy consumed	50	
Biomass gene biomass produced	50	
Biomass genes in genome	2	

Table 1: Model and evolution parameters

where A is the number of occupied positive binding sites of gene G_i, B is the number of occupied negative binding sites of gene G_i and R is a random number between 0 (inclusive) and 1 (exclusive). Binding site occupation is determined by the $binding_status$ matrix.

Molecular Production Costs Transcription and translation are not free processes: energy is used whenever they take place. When an activated gene's protein binds to a binding site, the energy value of the model is decreased by 1, representing the cost of transcribing and translating the transcription factor.

Biomass production also requires energy. Whenever a *biomass gene* is activated, the energy level decreases by the gene's 'energy consumption' value, and the biomass level increases by the gene's 'biomass production' value.

Deterministic Simulation The simulation can be turned into a deterministic Boolean network, by replacing the DNA-protein interaction step (2) with a binding threshold:

$$binding_status'_{ij} = (B'_{ij} \times gene_status_i)$$

$$B'_{ij} = \begin{cases} 1 & if\ B_{ij} \geq T_{bind} \\ 0 & otherwise \end{cases}$$

[1] Values taken from Cordero and Hogeweg (2006)

Basal transcription, K_{basal}, is also set to 0, meaning that a gene must be bound by an activator to transcribe.

Evolution Framework

The evolution framework used in the model is a standard genetic algorithm, with a fixed population size, and a purely elitist strategy that emulates the spatial constraints on a bacterial population, in, for example, a chemostat, where the fittest cells are ones that replicate fastest. A daughter cell is generated at each generation representing a simplified bacterial asexual replication. Due to the nature of DNA replication both the daughter parent cells are subjected to possible mutation. During replication, each gene in the genome can be affected by one of the evolutionary operators (#1-7). HGT (#8) is applied after genome replication and mutation. If HGT takes place, a donor genome from the population is selected at random, and a randomly selected number of genes are copied from the donor genome.

Fitness of an individual model is based solely on the level of biomass production after the defined number of time steps. If the simulation terminates due to lack of energy, the model has died and has a fitness of -1. In the neutrally evolved populations, the fitness function is a random number between 0 and 1, implying no selection pressure.

Model lineages are defined as a group of models with a common ancestor and are determined after evolution.

Results and Discussion

Model and Environment Regimes

In a simple environment, where the model has a constant supply of food, we evolved four types of models: 1) Energy signal gene present in a small genome, 2) Energy signal gene present in a large genome, 3) Energy signal gene not present in a small genome, 4) Energy signal gene not present in a large genome.

With an energy signal gene and a small genome, a final population evolves with a very simple regulatory network (Table 2). The main component of this network is a strong positive regulation of one of the *biomass genes* from the *energy signal gene*, but also has some residual connectivity between regulatory genes. However, no regulation (positive or negative) due to the input food genes was evolved. This is to be expected, as the environment remains constant, and so provides no useful information to be exploited. This regulation network is a simple, but effective system; whenever the model has sufficient energy, the energy signal is present, and it strongly activates the biosynthesis pathway gene; when the energy drops below this level activation of the biosynthesis pathway ceases. Only one of the biomass genes is activated, so whilst the system may not be maximally efficient at generating biomass, the model is far less likely to overexpress genes, in particular the energy-expensive biomass genes, and so is far more likely to survive to the end of the

simulation. This network also allows a far more robust regulation of biosynthesis, as the energy signal gene is not affected by noise. The use of an energy signal for activating growth parallels many organisms such as *E. coli*, with its use of CRP-cAMP.

Regulator type		Population			
		E	**N**	**NI**	**R**
No ES	Activator	31.85	43.08	46.48	45.92
	Repressor	17.27	43.58	46.13	46.05
	Dual	1.73	3.02	2.37	1.77
ES	Activator	6.17	43.45	47.39	47.41
	Repressor	1.66	51.64	46.94	47.70
	Dual	0.05	3.48	2.32	1.75

Table 2: Mean distribution of connection type in different populations of both energy signal and no energy signal. E is evolved, N is neutral, I is initial and R random population

With no energy signal gene and a small genome, a very different regulation network is evolved. In this population, the most successful models again consisted of no regulation due to the input food genes, and so no input stimuli at all were available (as the energy threshold gene is regulated by the model itself, it can be classed as an input). Thus the models rely solely on stochasticity for transcription and translation of random genes. The model did however evolve some positive regulation from a small number of standard genes to the biomass genes (Table 2); this increases the probability that the biomass genes will be activated at a given time step, and so the efficiency of generating biomass. Whilst this network is not very efficient at generating biomass, or robust to noise due to the reliance on stochasticity, it is well adapted for survival. The lack of energy signalling used in the evolution of this population of models shares several parallels with the lack of signalling in *B. aphidcola* cells, and a similar, simple regulatory network in observed in both. The exploitation of stochastic gene expression seems to be a robust sub-optimal solution for survival without environmental information. This solution may provide a mechanism for survival in early gene regulatory networks, until more precise signalling networks evolve, or could itself be the basis for a signalling network.

With a much larger genome, with or without an energy signal, we observe very different results. Network connectivity is necessarily high because of the number of genes and small shape space. Models are unable to survive because they very quickly over-express many genes and use up all energy. Even under more energetically favourable conditions (energy from food = 40; starting energy = 4000) the models are still unable to survive. This indicates the importance of repressors within biological networks to tightly regulate the processes of transcription and translation, as are not 'free' (they require energy sources e.g. ATP). Other com-

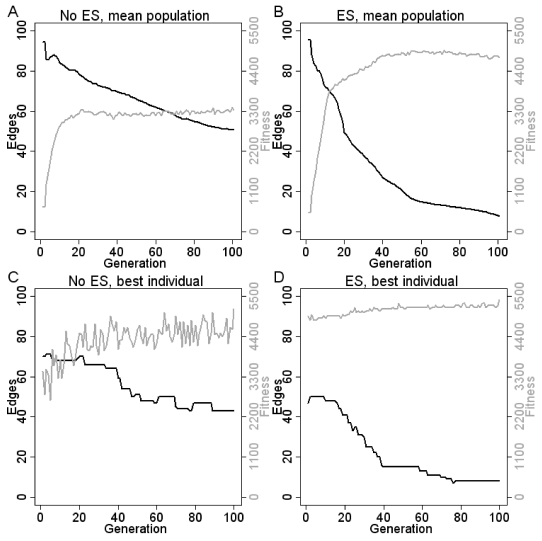


Figure 3: Evolutionary history of mean population fitness and number of regulatory connections in a small genome. A) mean fitness and number of regulatory connections in the population with no energy signal. B) mean fitness and number of regulatory connections in the population with an energy signal. C) best individual in the final population without an energy signal. D) best individual in the final population with an energy signal. The decrease in network connectivity and increased fitness can be seen in all plots.

putational models have obtained the evolution of repressor systems, even in constant environments (Jenkins and Stekel (2008)). The regulatory network of *E. coli* displays a preference for negative regulation by transcription factors in many different systems (Karp et al. (2007)). This may indicate a further use of negative regulation as an adaption for efficiency, as well as enabling large scale switching of regulatory systems, fast responses and maintaining homeostasis. Indeed, strong negative self-regulation has been shown to decrease the amount of mRNA needed to express a protein at a set level, thus reducing the use of energy expensive processes (Stekel and Jenkins (2008)). One possible explanation for the lack of large global repressors evolving in the current implementation of the model is the energy cost of maintaining sufficient numbers of repressor proteins. Protein stability is fixed to one timestep, so proteins must be produced each timestep, using up large amounts of energy. In biological systems, protein stabilities ranging from minutes to many hours are observed (Nath and Koch (1970)). The stability of a protein is often associated with function: signalling proteins are typically short-lived; metabolic proteins are often more stable. Modifying the model to allow proteins to evolve their stability may allow the evolution of global regulators. In addition, real biological molecules have a large shape space, due to the very high dimensionality of protein shape. Increasing the shape space in the model

could help alleviate the high network connectivity.

Effects of Stochasticity Removing stochasticity dramatically alters the networks evolved. Whilst a similar regulation mechanism is observed in populations with an energy signal, the number of connections does not rapidly decrease. Network connectivity remains high, with the exception of input genes. This occurs as the regulatory genes will only be transcribed if activated by another gene, leading to large parts of the network which are highly intra-connected with no external inputs. There is no pressure to reduce this connectivity, provided no input genes connect into the large highly connected parts. The high connectivity may appear to be a complex solution, however, the increased connectivity may merely mask the underlying core functionality of the model.

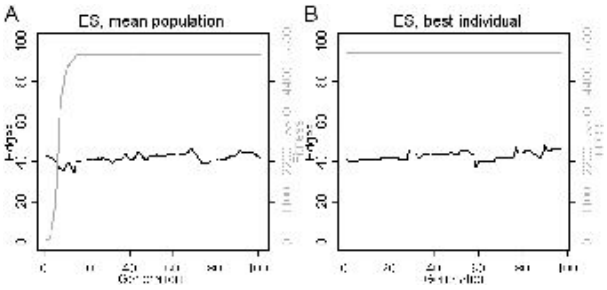


Figure 4: Evolutionary history of deterministically simulated small genome population with energy signal. A) mean fitness and number of regulatory connections in the population. B) best individual in the final generation of the population. Network connectivity remains high in both plots, unlike in the stochastic simulation populations.

Populations without an energy signal were unable to produce any surviving models, indicating that the exploited stochasticity of the original solution is essential.

Neutral Evolution and Comparison

To compare the effects of directed evolution, we evolved populations under the same four conditions but used a random fitness function to simulate 'neutral' evolution. We examined the model networks at three points during the evolution: 1) after random model initialisation (R), 2) after initial period of evolutions, creating a 'realistic' network (NI) and 3) after a given number of generations (N).

Several network properties are extracted from the networks: binding site distribution, binding site regulation type ratio, gene 'out' degree (number of genes the transcription factor interacts with), gene 'in' degree (number of transcription factors which regulate the gene) and number and type of self-regulating connections.

Binding Sites A general trend for loss of binding sites can be see in Figure 5. In the directed evolution populations, a larger number of genes in both populations have no binding sites, and have a much smaller distribution of maximum binding sites per gene. This shows how the model has evolved to optimise its regulatory network, by reducing it. There was no significant bias to regulation type in each population (Table 3), however, a clear trend for activating connections in the evolved populations is shown in Table 2. This may be linked to the lack of the evolution of global repressors as discussed above. Without a global regulatory mechanism, the model is unable to effectively regulate the expression of the genes, and so the alternative solution is to reduce the probability of transcription factor activity by losing binding sites. Whilst this solution does not prevent transcription, it does reduce it. In fact this mechanism is exploited in the populations without an energy signal.

	Binding site type	Population			
		E	N	NI	R
NoES	Activator	17.06	24.37	25.84	25.42
	Repressor	18.55	25.27	25.73	25.51
ES	Activator	12.38	24.70	25.76	25.54
	Repressor	13.69	26.56	25.67	25.64

Table 3: Mean distribution of binding site regulation type in different populations of both energy signal and no energy signal. E is evolved, N is neutral, I is initial and R is random population

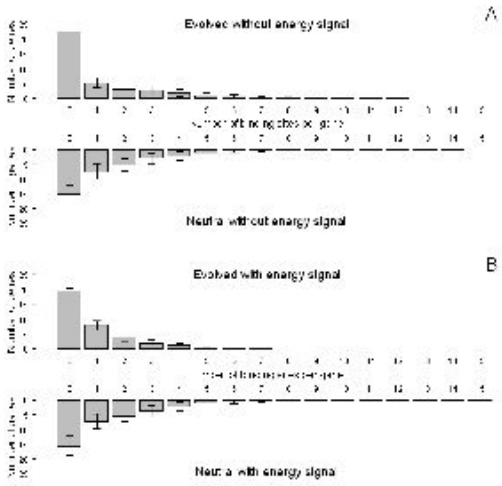


Figure 5: Mean number of binding site for each gene in each model in small genome populations. Error bars are 1 s.d. of the population. A) evolved and neutral populations without energy signal, B) evolved and neutral populations with energy signal. The loss of binding sites in the non-neutral populations can be seem in both panels; the evolved populations have a larger number of genes without any binding sites, and have a lower maximum number of binding sites per gene.

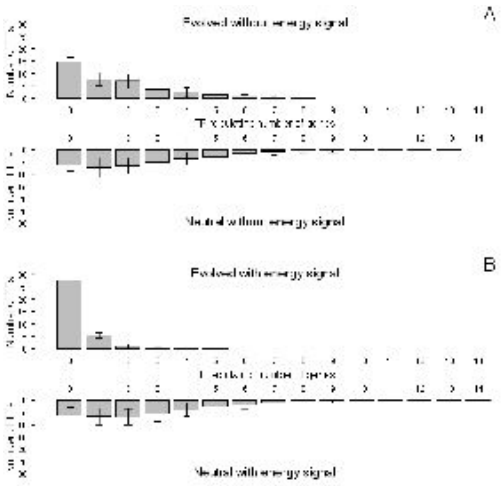


Figure 6: Mean gene 'out' degrees for each gene in each model in small genome population. Error bars are 1 s.d. of the population. A) evolved and neutral populations without energy signal, B) evolved and neutral populations with energy signal. The loss of connectivity in the evolved populations is indicated by a larger number of genes which do not act as transcription factors and a reduced number of 'global' transcription factors.

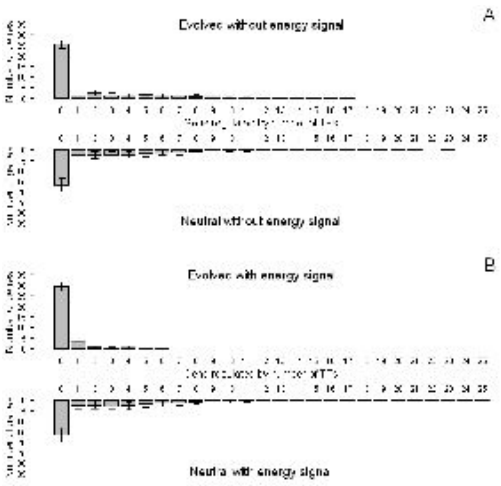


Figure 7: Mean gene 'in' degrees for each gene in each model in small genome populations. Error bars are 1 s.d. of the population. A) evolved and neutral populations without energy signal, B) evolved and neutral populations with energy signal. The loss of connectivity in the evolved populations is indicated by a larger number of genes without any transcription factor interaction and a smaller distribution of interactions.

Transcription Factor Activity The loss of large amounts of regulation can be seen in the interactions between transcription factors (TFs) and genes. This is indicated by the 'out' degree for each transcription factor (Figure 6), and the 'in' degree for each gene (Figure 7). We observe an increase in the number of proteins that do not act as TFs and the number of genes which are not regulated by any TFs. The maximum number of genes regulated by a TF is also significantly reduced in the evolved populations, in particular the population with an energy signal. The maximum number of TF's regulating a gene is also significantly reduced in the evolved populations.

The number of self-regulating genes were separated into: activating only, repressing only, and dual regulation. Again, a clear trend can be observed from the directed populations from Table 4. The two evolved populations have lost nearly all of their self-activating connections, and a large proportion of their self-repressing connections. Whilst more activating connections in total are conserved (Table 2), a larger number of negatively self-regulating connections are conserved, indicating the importance of negative self-regulation in transcription networks.

These results indicate the loss of interaction within the network, and highlight that complex regulatory networks are unnecessary to survive within a stable environment. The preference for losing self-activating connections and preserving more self-repressing connections shows that the network attempts to optimise its energy usage by preventing further transcription of unrequired genes.

Further Discussion The results obtained from the evolutions described above have shown two very different, but realistic regulation mechanisms have been selected and evolved. When no energy signal gene is present in the genome, the population has evolved to exploit the stochasticity within the transcription and translation processes. Whilst the biomass genes are seemingly not activated by the food inputs, they have evolved a large number of activating connections from many other genes. This strategy allows the model to exploit the stochastic gene expression, potentially tuning the number of activating connections to ensure that enough genes will randomly activate the biosynthesis pathways, and ensuring that these pathways are not overexpressed.

The other regulatory mechanism evolved, whilst being less complex, is one that is observed in many biological regulatory systems. The energy signal is used as input for the biosynthesis pathways, and regulation of other genes is much more tightly controlled through loss of connectivity.

These rather surprising results highlight the complexity of regulation networks even in the most simple of environments. They also show the ingenious mechanisms which natural selection, and the evolutionary operators it uses, have discovered and optimised in both the model networks presented here and the real biological systems.

Regulator type		Population			
		E	N	NI	R
No ES	Activator	0.0165	1.0535	1.319	1.3295
	Repressor	0.4295	1.1645	1.3125	1.3060
	Dual	0	0.0785	0.0600	0.0435
ES	Activator	0.0005	1.0070	1.3125	1.3205
	Repressor	0.2830	1.4440	1.3130	1.3395
	Dual	0	0.1060	0.0575	0.0520

Table 4: Mean number of activating, repressing and dual-regulating self-regulating connections per model within the energy signal and no energy signal populations. The loss of connectivity can be seen in the evolved populations. The evolved populations show a significantly smaller number of activating and repressing and no dual interactions compared with the neutral and random populations. E is evolved, N is neutral, I is initial and R is random population

Summary and Conclusions

This paper expanded an existing model for genome evolution and added a simulation method, developed from Boolean network models. Models are evolved in populations with and without energy signalling genes, and the evolved models are compared with models evolved neutrally, and random models.

Results from the evolutions indicate a decrease in the number of regulatory connections within the networks, and a preference towards negative regulatory interactions. A number of parallels are drawn between the evolved models and biological systems, including: regulation by the global regulator CRP-cAMP in *E. coli*; a regulation mechanism similar to the endosymbiont *B. aphidcola*; the use of negative regulation as a mechanism for efficiency; and the need for differing protein stabilities dependent on function.

Acknowledgments

We thank Gavin Thomas for information on *B. aphidcola*. DJJ is funded by BBSRC Studentship BBS/S/S/2005/12006. Simulations were on High Performance Compute Cluster funded by BBSRC grant BB/D524624/1.

References

Albert, R. (2005). Scale-free networks in cell biology. *Journal of Cell Science*, 118:4947–4957.

Cordero, O. X. and Hogeweg, P. (2006). Feed-forward loop circuits as a side effect of genome evolution. *Molecular Biology and Evolution*, 23(10):1931–1936.

Elowitz, M. B., Levine, A. J., Siggia, E. D., and Swain, P. S. (2002). Stochastic gene expression in a single cell. *Science*, 297(5584):1183–1186.

Hardiman, T., Lemuth, K., Keller, M. A., Reuss, M., and Siemann-Herzberg, M. (2007). Topology of the global regulatory network of carbon limitation in *Escherichia coli*. *Journal of Biotechnology*, 132:359–374.

Ingram, P. J., Stumpf, M. P. H., and Stark, J. (2006). Network motifs: structure does not determine function. *BMC Genomics*, 7(108).

Jenkins, D. J. and Stekel, D. J. (2008). A new model for investigating the evolution of transcription control networks. *Artificial Life*. In press.

Karp, P. D., Keseler, I. M., Shearer, A., Latendresse, M., Krummenacker, M., Paley, S. M., Paulsen, I., Collado-Vides, J., Gama-Castro, S., Peralta-Gil, M., Santos-Zavaleta, A., naloza Spínola, M. I. P., Bonavides-Martinez, C., and Ingraham, J. (2007). Multidimensional annotation of the *Escherichia coli* K-12 genome. *Nucleic Acids Research*, 37:7577–7590.

Kauffman, S. A. (1969). Metabolic stability and epigenesis in randomly constructed genetic nets. *Journal of Theoretical Biology*, 22(3):437–467.

Kuo, P. D., Banzhaf, W., and Leier, A. (2006). Network topology and the evolution of dynamics in an artificial genetic regulatory network model created by whole genome duplication and divergence. *Biosystems*, 85(3):177–200.

Mangan, S. and Alon, U. (2003). Structure and function of the feed-forward loop network motif. *Proceedings of the National Academy of Sciences*, 100(21):11980–11985.

Milo, R., Shen-Orr, S., Itzkovikz, S., Kashtan, N., Chklovskii, D., and Alon, U. (2002). Network motifs: Simple building blocks of complex networks. *Science*, 298:824–827.

Moran, N. A. and Mira, A. (2001). The process of genome shrinkage in the obligate symbiont *Buchnera aphidicola*. *Genome Biology*, 2(12).

Nath, K. and Koch, A. L. (1970). Protein degradation in *Escherichia coli*: 1. measurement of rapidly and slowly decaying components. *The Journal of Biological Chemistry*, 245(11):2889–2900.

Quayle, A. P. and Bullock, S. (2006). Modelling the evolution of genetic regulatory networks. *Journal of Theoretical Biology*, 238(4):737–753.

Shigenobu, S., Watanabe, H., Hattori, M., Sakaki, Y., and Ishikawa, H. (2000). Genome sequence of the endocellular bacterial symbiont of aphids *Buchnera* sp. aps. *Nature*, 407:81–86.

Stekel, D. J. and Jenkins, D. J. (2008). Strong negative self regulation of prokaryotic transcription factors increases the intrinsic noise of protein expression. *BMC Systems Biology*, 2(6).

van Noort, V., Snel, B., and Huynen, M. A. (2004). The yeast coexpression network has a small-world, scale-free architecture and can be explained by a simple model. *EMBO reports*, 5(3):280–284.

Wilcox, J. L., Dunbar, H. E., Wolfinger, R. D., and Moran, N. A. (2003). Consequences of reductive evolution for gene expression in an obligate endosymbiont. *Molecular Microbiology*, 48(6):1491–1500.

Zheng, D., Constantinidou, C., Hobman, J. L., and Minchin, S. D. (2004). Identification of the CRP regulon using *in vitro* and *in vivo* transcriptional profiling. *Nucleic Acids Research*, 32(19):5874–5893.

Evo-devo *in silico*: a Model of a Gene Network Regulating Multicellular Development in 3D Space with Artificial Physics

Michał Joachimczak, Borys Wróbel
Computational Biology Group, Department of Genetics and Marine Biotechnology
Institute of Oceanology, Polish Academy of Sciences
Powstańców Warszawy 55, 81-820 Sopot, Poland
mjoach@gmail.com

Abstract

We present a model of multicellular development controlled by a gene network in which the connectivity is determined by the proximity of sequences in N-dimensional space. Thus the sequences of individual genes can be visualised as points in space which approach or move away from one another as the genomes evolve. The genotype-phenotype (morphology) mapping in our model is indirect, relies on artificial physics, and allows cell adhesion and free movement in 3D space. Cell differentiation is allowed by positional information provided by factors that diffuse in this space, and the differential gene expression in each cell determines the cell fate (such us division, death, growth and movement). We apply a genetic algorithm to find genotypes that can direct morphogenesis of non-trivial asymmetrical shapes. We then investigate the mechanism of such developmental process and the features of gene regulatory network that direct the embryogenesis.

Introduction

Generation of two-dimensional (2D) patterns, such as the French flag (Wolpert, 1969, see also Steiner et al., 2006), is much simpler then the problem of 3D morphogenesis. Recent progress in this field was facilitated by the introduction of gene regulatory networks (GRNs) as an indirect regulatory mechanism in 3D artificial embryogenesis that relies in part on simulated physics (e.g. Eggenberger-Hotz, 2003b,a, 2004, see also Bongard and Pfeifer, 2003), in contrast to the initial more abstract approaches, such as generative encodings (see e.g. Prusinkiewicz and Lindenmayer, 1996).

In biological systems, the structure of GRNs is encoded indirectly in nucleic acid-based genomes. Coding sequences are accompanied (usually, preceded) by regulatory areas (cis-regulators, promoters) which regulate the level of gene expression. The coding sequences code for functional products: catalysers of biochemical reactions (enzymes, ribozymes), proteins that have structural/mechanical roles, and finally, regulatory products that bind other products or bind to regulatory areas in the genome to control the production of other products (gene expression). The gene products are the nodes in the biological GRNs while the edges are defined by regulatory interactions. The amino acid sequence of a protein product (or a nucleotide sequence of an RNA molecule) defines its 3D structure in a way that is still far from being fully understood (this is the so-called folding problem). Interactions between three-dimensional molecules (between proteins, proteins and nucleic acids, and between RNA molecules, etc.) are even more difficult to model.

Several approaches to encode the structure of the artificial GRNs in the genome using an abstraction of this "lock and key" mechanism of molecular recognition have been proposed. For example, product-promoter affinity can be determined in an all-or-none manner by a direct match between numbers assembled from the digits in the genomic sequence (Quayle and Bullock, 2006) or coded directly in the genome, possibly with real-number rounding (Bongard and Pfeifer, 2003). Jakobi (1995) used a different approach, with promoter affinity (a discretised value from 0 to 1) determined by the match between triplets of "chemicals": characters in a regulatory protein sequence (from a 64-letter alphabet) and in the genome (from a four-letter alphabet). The triplets are found indirectly in a metaphor of genome scanning by the RNA polymerase, folding of the regulatory protein, and protein-protein interaction between them. The method proposed by Eggenberger-Hotz (2003a) is much simpler and relies on direct proximity of real numbers encoded in the genome. Bit-by-bit comparison of 32-bit integers is another method of similar complexity (Banzhaf, 2003; Kuo et al., 2004). Bentley (2003) proposed a much more indirect approach based on encoding the coordinates for subsets of Mandelbrot set and matching their similarity.

We extend the approach that uses the proximity of real numbers (a 1D approach) by introducing a model of GRN in which product-promoter affinity depends on the Euclidean distance between points in N-dimensional gene sequence space. As the genomes evolve, these points approach or move away from one another.

The Model

Outline

In the model of embryogenesis proposed here, multicellular development starts from a single cell. Each cell of an

individual has the same linear genome: a list of genetic elements. Each element is characterized N+1 real values (N coordinates in the gene sequence space and a gene modifier, see below) and an integer (element type), with "gene" (a metaphor of a coding sequence) and "promoter" (an abstraction of a regulatory region) as the main types.

One of the key interests in our research is the gradual increase in complexity of regulatory networks and inherent mathematical properties of the corresponding graphs. We thus allow the genomes of arbitrary size and regulatory units that have no upper limits on the number of nodes they directly interact with. Furthermore, we provide multiple layers of possible interactions in developing artificial organisms for evolution to tinker with. On the most basic level, inside a single cell, the products that we call "internal" have affinity to promoters and control the expression of other genes. On the level of a whole organism, special class of products ("external products") may diffuse from the source cell. External products may have affinity to promoters but also to products belonging to another class: receptors. Thus external products are a metaphor for morphogens in natural embryogenesis, and their interactions provide a mechanism for cell differentiation and differential growth. Furthermore, cells develop in an environment with simple simulated physics: overlapping cells repel each other while daughter cells are attached to the mother cells with simulated springs (a metaphor of adhesive forces between cells).

Regulatory Units

Regulatory units are formed from a series of promoters followed by a number of genes and are the basic building blocks for the structure of the GRN. There are a metaphor of regulatory units in nucleic acid-based genomes, in which several protein- or RNA-coding regions can be under the control of several regions that affect gene expression at different levels: pre-transcriptional, post-transcriptional (stability, transport, and translation of transcripts), and post-translational (stability, transport and activity of proteins).

The majority of existing models of artificial embryogeny follows the scheme of multiple promoters and one product (e.g. Beurier et al., 2006; Steiner et al., 2006; Eggenberger-Hotz, 1997, 2003b,a, 2004). However, many-to-many relationship between the promoters and regulated genes is common both in prokaryotes and eukaryotes (Gerstein et al., 2007). Indeed, clustering of several genes in so-called operons is not, as originally thought, restricted to bacteria, but common also in eukaryotes (for a recent review, see Blumenthal, 2004; Gerstein et al., 2007). Such arrangement allows for co-regulation of co-transcribed genes that are closely related functionally (for example, involved in the same biochemical process). A similar logic applies to multiple transcripts sharing common regulatory regions (promoters, enhancers, silencers; Gerstein et al., 2007), polyproteins, and ineed, multidomain proteins (with domains responsible for separate functions).

To locate regulatory units in our model, each genome is scanned linearly. Whenever a sequence of elements consisting of at least one promoter followed by at least one gene is detected, it is treated as one unit that extends until the next promoter. Promoters and genes outside regulatory units are ignored. For example, in a genome "GGGppGpGGpGpp" (where each p is a promoter and each G a gene), three regulatory units (square brackets) exist: GGG [ppG] [pGG] [pG] pp.

Two types of promoters are introduced: additive and multiplicative. To compute the level of activation of a regulatory unit (the expression level of all of the products), we first compute the activity of all of its promoters:

$$p_i = \sum_{k=1}^{K} L_k w_{k,i} \,. \tag{1}$$

where p_i is the activity of a given promoter, K is the total number of regulatory factors in the genome (e.g. internal or external products, see below), L_k denotes the perceived level of the factor k, and $w_{k,i}$ is the promoter-factor affinity:

$$w_{k,i} = \begin{cases} d_{k,i} \leq 5 : \mathrm{sgn}(m_k m_i) \frac{2|m_k m_i|(5-d_{k,i})}{10d+|m_k m_i|} \,; \\ \\ d_{k,i} > 5 : 0 \,. \end{cases} \tag{2}$$

where $d_{k,i}$ is the Euclidean distance between the sequences of the promoter i and the gene of factor k, while m_k and m_i are the values of their modifier fields. In other words, affinity is 0 (no interaction) when the distance is larger than 5, and at a maximum (10) when the distance is 0. For intermediate distances, the affinity falls hyperbolically, rapidly for small $|m_k m_i|$ and approximately linearly for large $|m_k m_i|$. The signs of the modifiers determine if the effect is inhibitory or excitatory.

All the genes belonging to a given regulatory unit have the same level of expression:

$$L_\Omega = f(\prod_{i=0}^{I} p_{m,i} \sum_{j=0}^{J} p_{a,j}) \,. \tag{3}$$

where I and J denote the number of multiplicative and additive promoters (respectively) and $p_{m,1..i}$ and $p_{a,1..j}$ describe their activations. It is possible to allow for the bias in activation ($p_{m,0}$, $p_{a,0}$), but we use the identity element of, respectively, multiplication and addition ($p_{m,0} = 1$,$p_{a,0} = 0$). f is a sigmoidal function returning value from (0,1) with the threshold at 0.5 (the steepness of the sigmoid was kept constant in all the experiments).

The presence of a multiplicative promoter in a regulatory unit results in a strict requirement for the expression of the associated product, otherwise the whole unit remains inactive. This feature (where a subset of transcription factors is

necessary to initiate gene expression in an 'all-or-nothing' fashion) is known to be common in gene regulation but would not be easily captured by a purely additive function. Additionally, introducing multiplicative promoters provides evolution with a mechanism to switch off the whole regulatory unit with a single mutation.

Products

We introduce three types of products that can be coded by genes in a regulatory unit: internal, external and receptors. Internal products affect the expression of the regulatory units only in the cell in which they are produced. External products (morphogens) diffuse from the producer cell and bind to promoters and receptors in other cells. Receptors, on the other hand, interact with external products and influence the axis of cell division (the division vector) by shifting it toward or away from the source. Since the cells may differ in the pattern of gene expression, also their set of active receptors may be different. This allows each cell to orient its division in relation to the pattern of morphogens that are available at a given moment.

The affinity between a morphogen and a promoter or a receptor is defined by Eq. 2, but to simulate simple diffusion, the perceived concentration of the morphogen depends not only on the level of production but also on the distance from the producing cells (the sources). The level of morphogen m perceived by the cell c is:

$$L_{c,m} = \sum_{i=1, i \neq c}^{I} l_{i,m} \frac{1}{1 + D_{i,c}} . \quad (4)$$

where I denotes the number of cells at the current developmental stage, $D_{i,c}$ is the distance from the cell c to the source i in the 3D space of the developing organism and $l_{i,m}$ is the level of morphogen m this source produces (the expression level of the morphogen, Eq. 3).

Additionally, to simulate some of spatiotemporal effects generated by diffusion, the actual value of $l_{i,m}$ is delayed in time, depending on the distance from the source. This adds some additional memory cost of storing morphogen levels from previous time steps for each morphogen in each cell, but at basically no additional computational cost. Alternatively, one could obtain more realistic diffusion by simulating a 3D grid in which morphogens would diffuse (e.g. Beurier et al., 2006; Steiner et al., 2006). However, computational cost of updating fine-grained diffusion levels in 3D would be considerable.

Other Types of Genetic Elements

Three additional element types exist: pseudogenes, external factors and effectors. They are ignored when the genome is scanned for regulatory units.

Pseudogenes. If any genetic element is mutated to a pseudogene, its sequence will be shielded from the selective pressures until another mutation changes the element type.

External factors. External factors act in exactly the same way as external products (that is, they may interact with regulators and receptors), however their levels of expression are not regulated by the cell: the factors are provided externally at predefined levels. Also, for the positional factors (see below) the source locations are predefined. External factors can be thus viewed as inputs of the GRN and their possible interactions with receptors is an initial mechanism for breaking the symmetry of cell divisions.

Two subclasses of external factors are introduced. The first one consists of positional morphogens, emitted from four different points in three-dimensional space. They are a metaphor for maternal factors in natural embryogenesis. Thus each cell is provided with enough positional information to locate itself in 3D (four points is the minimal number that allows 3D trilateration). The perceived level of the positional external factors in a particular cell depends on the Euclidean distance to the source, in the same way as the level of external products (Eq. 4).

The perceived level of external factors in the second class does not depend on cell location. The level of one is constant throughout the development (a "1" signal), so it can be used as a simple threshold for any of regulatory unit in the GRN. The other provides a time signal, its expression increasing linearly from 0 to 1 during the developmental process. The next two are somewhat related: a generation counter (incremented in each daughter cell after division), and the energy depletion level, which increases from 0 to 1 (each cell division has some cost for both the daughter and the mother cell). The level of the last factor in this class depends on the number of neighbours (saturating at 1 for 8 cells in close proximity) and thus allows the cell to detect when it lies in a densely packed cell structure.

Effectors. Effectors can be viewed as outputs of the GRNs. They either correspond to actions each cell can take during the development or allow to adjust the parameters of the developmental process. Each effector is defined by its sequence in the N-dimensional gene space, and products that have their sequences close enough will add to its activation (using Eq. 2). In a way, this parallels the promoters, and indeed one can consider each effector as a special regulatory unit with a single promoter.

Cell actions consists of all-or-none responses when activation levels of corresponding effectors reach a certain threshold. These are: division, apoptosis (programmed cell death) and freezing (after which expression levels in the cell are no longer updated). In the second group of effectors, the following parameters are updated by a value corresponding to the activation level defined by Eq. 3: cell radius, spring

length, internal division vector length and internal division vector angles.

Since the list of possible effectors and external factors is predefined and we prefer to avoid defining a separate element type for each, the assignment of a given element coding an effector or an external factor to a particular function depends on its order in the genome. After all the functions are assigned, the rest is treated as pseudogenes.

Development

Development starts from a single cell and proceeds over discrete time steps. It stops either when a maximum time step is reached or when an individual embryo exhausts its initial energy.

The state of each cell is determined by the expression levels of all of the products in the genome, cell coordinates in 3D space (these are real values, no grid is used), cell radius, the orientation of the cell division vector, energy level, and several parameters related to the physical model. As the fate of the cells depends on the differential gene expression, it is essential to provide a mechanism that will break the initial symmetry of cell divisions. This is the function of the initial gradients of positional external factors. A similar mechanism is known to direct the initial stages of insect embryogenesis (for a popular introduction, see Carroll, 2005).

When a new cell is formed, it is attached to its mother with an elastic spring and placed at a very small distance, so the two cells initially overlap in space. The default length of the spring is equal to the sum of cells' radii and can be increased by designated effector gene, if present. During subsequent time steps after division, the elasticity of the spring will push away the cell in the direction of the mother's division vector. The spring ensures that the new cell moves away in the desired direction while remaining close to the mother, simulating simple adhesion. Repulsion between any two cells (at a certain distance) ensures that the cells do not overlap in space. To prevent brusque movements of the cells, their motion is slowed down with simulated viscosity. In a manner similar to spring length, the radius of the daughter cell is controlled by a dedicated effector. No activity means default value, maximum activity translates into a twofold increase.

The position of the daughter cell after division is influenced by two mechanisms, each corresponding to one of two auxiliary vectors maintained for each cell: the internal and external division vector. The direction of the sum of these two vectors (the cell division vector) gives the direction of the spring that attaches the daughter to the mother cell.

The first mechanism is directly based on the mechanism used in 3D L-systems (Prusinkiewicz and Lindenmayer, 1996). A daughter cell inherits the internal vector from the mother. At this point the vector is rotated in the daughter cell. Each of the three angles of rotation is affected by the expression of one of three effectors in the mother cell (Prusinkiewicz and Lindenmayer, 1996). If the effector is activated (Eq. 1), the rotation is positive, repression by inhibitory regulators results in negative rotation. An additional effector is used to determine the length of the internal vector. The default vector length is 0, which means that if there are no products acting on this effector (or the element corresponding to it is not present in the genome), the direction of the cell division vector will not be influenced by this mechanism.

The second mechanism allows to orient the vector towards or away from morphogen sources. High positive affinity between the sequence of an active receptor and the sequence of a morphogen perceived in the cell shifts the direction of the external vector toward the source of this morphogen. Negative affinity shifts the vector in the opposite direction. The overall effect is a sum of interactions of all receptors in the given cell with all morphogens produced by every source:

$$\vec{V_c} = \sum_{r=1}^{R} \sum_{m=1}^{M} \sum_{s=1, s \neq c}^{S} l_r w_{r,m} L_{c,m} \vec{\delta}_{s,c} \,. \qquad (5)$$

where R denotes the total number of receptors in the genome, M the total number of external products and external factors defined in the genome, S is the number of sources (cells and four positional external factors), l_r is the expression level of the receptor r in the cell (Eq. 3), $w_{r,m}$ is the morphogen-receptor affinity (Eq. 2), $L_{c,m}$ is the perceived level of the morphogen (Eq. 4), and $\vec{\delta}_{s,c}$ is the normalized vector from the given cell to the source.

To allow for a control of cell divisions, we provide an input to the GRNs that is a metaphor of the nutritional/energetical state of the cell. Since in our model each cell division has some energetical cost, the cell energy can be exhausted by rapid divisions. The same applies to the whole developing individual: there is a limit on the total energy that can be used during the development. As mutations causing uncontrolled cell divisions put a high drain on computational resources during evolution, early exhaustion of such individual's energy can help keep the problem in check. Additional biological realism is introduced by requiring a brief (10 simulation time steps) period of division arrest right after a division, both in the mother and in the daughter cell. Arrested cells update the state of their GRN normally but cannot divide no matter how high the expression of the corresponding effector. This has an additional advantage of giving the simulated physics the time to adjust the position of the new cell.

Fitness evaluation

The most obvious way to assess the fitness in simulations of morphological development is to count how many cells fit inside the desired shape, penalising for each cell outside the shape (e.g. Kumar, 2004). This approach works well when cells can only take certain locations on the grid, but leads to

undesired results when cells take arbitrary positions in space and can temporarily overlap. The possibility to reach high fitness by producing densely packed and highly overlapping cells would allow to exploit the simulated physics and other features of the model in an unintended way. We propose an alternative: a cuboid in 3D space that contains the target shape is divided into cubical voxels and each voxel is marked either as internal or external to the shape. To compute fitness, we iterate over each cell and check whether they occupy internal voxels, and if so, those voxels are marked as occupied. This approach has several advantages. First of all, it is efficient and allows to avoid repeated scoring of voxels occupied by overlapping cells. Secondly, it allows the cells to adopt different sizes (and even shapes, although this is not explored here). Finally, it is possible to give higher weights for some of the voxels to assist the evolution of morphologies that otherwise do not evolve easily.

Implementation

The computations are simplified by first transforming the genome into a GRN graph, in which only if the distance between the sequences is smaller than the threshold (5), an edge is drawn (see Eq. 2). During the development, it allows to update the state of the GRN using a list of factors that affect each promoter or receptor.

The dynamics of cell movement is simulated with simple Newtonian physics, using Runge-Kutta 4th-order integration. Springs behave according to Hooke's law and additional repulsive force is introduced between any two cells that overlap.

For complex GRNs it takes considerably less time to compute the new location of the cells compared to the time taken to update the state of the GRN. It is thus possible to update the GRN state, for example, only every 10 steps of simulated physics.

Genetic algorithm

All the results obtained in this work were obtained using a generational genetic algorithm with constant-size population of 300 individuals. A new generation was formed by copying 5 genomes without mutation (elitism), 150 with mutations and crossover, and 145 by mutation only. We allowed for multi-point crossover between genomes of different sizes. The candidate genomes for the next generation were chosen using tournament selection (which is not susceptible to the scaling of the fitness function). Elite individuals replaced the elite individuals from the previous generation if their fitness was equal. This allows elite genomes to wander through the neutral regions in the sequence space (which may allow for a more efficient evolutionary search, Shipman et al., 2000).

The elements in the genome are the lowest level of abstraction in our model, so the genetic operators were designed to work on the level of the genetic elements (rather than single bits or real values): each had a predefined probability of occurrence per element in a genome and per generation. The first operator results in a modification of the modifier or of the coordinates in the N-dimensional sequence space. The coordinates are modified by addition of a small value drawn from a Gaussian distribution. This operator corresponds to simple mutations in nucleic acid sequence (such as point mutations, short deletions and insertions in the coding or regulatory sequences). The second mutational operator, on the other hand, does not have any obvious biological interpretation, and allows to change the sign of the modifier. Another mutational operator allows for a change in the element type (with unequal probabilities for each type), in particular, a change of any element to a pseudogene and vice versa, with an obvious biological parallel. However, we allow any type change, which includes a direct change of a receptor into a morphogen or a promoter to a product (and vice versa), while conserving the sequence. In further work, we plan to explore if this feature helps evolution, at any rate, it does not have an obvious natural counterpart.

The remaining mutational operators act on the level of whole elements (element deletion, duplication, and insertion of a randomly created element) and the whole genome: deletion of a segment of the genome with random start and end point and a duplication of such a segment to a random position in the genome.

In the experiments described below, we set the probabilities of deletions to be around twice as high as probabilities of element duplications and insertions. Such deletion pressure restricts the accumulation of elements whose presence does not affect fitness (i.e. in which mutations are neutral) and so prevents the unnecessary growth of genome size. This particular solution to the genome size issue was partly motivated by biological realism (Charles et al., 1999), and partly by difficulties in properly balancing the fitness function faced by an alternative: a fitness cost to larger genomes. However, some level of neutral elements (which include pseudogenes) is beneficial. In natural genomes the presence of regions in which mutations do not affect the phenotype (neutral regions, junk DNA) allows for the appearance of innovations beneficial from the point of view of natural selection (Shipman et al., 2000). Such regions are shielded from the selective pressures which allows for bolder movements in the sequence space.

Results

In all our experiments, the evolution started from the same simple genome (Fig. 1A) designed by hand and containing four regulatory units, all regulated by external factors. The products in two units have effect on the division effector, two other induce rotation of the internal division vector and its length. The remaining external factors and effectors are defined but the nodes are not connected to the others in the GRN (and are not shown in Fig. 1B). The number of dimen-

sions of the gene sequence space was set to two.

In a way, the presented model attempts to trade simplicity for biological realism. In further work we plan to address the question whether the model can be simplified (or, indeed, complicated). Before it is possible, we need to ask if this initial version allows for efficient search in non-trivial fitness landscapes, by challenging the genetic algorithm with target shapes of different difficulty. While highly symmetric structures (spheres, ellipsoids) or slightly more demanding half-ellipsoids evolved quite easily (not shown), asymmetric morphologies, shown on Fig. 2 are a difficult task, and usually over 500 generations were needed to find a solution. Interestingly, the solutions found by the genetic algorithm did not rely on cell death (a mechanism that we observed to be used often in the development of half-ellipsoids; not shown) but rather on differential cell division and cell growth (changing cell radius) in different regions.

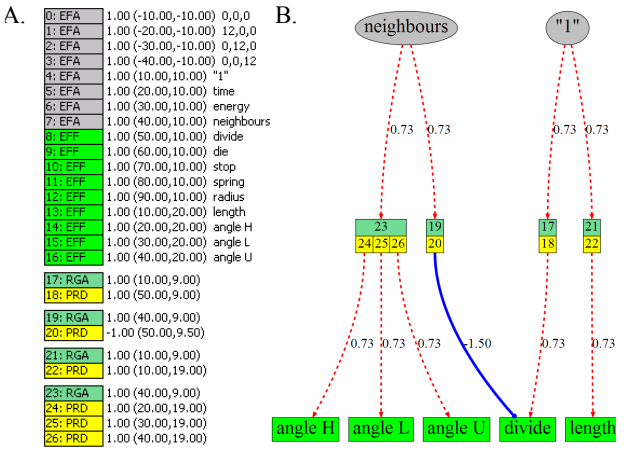


Figure 1: The seed genome (A) and the corresponding gene regulatory network (GRN; B). The genome consists of 27 elements (the value of the modifier, the coordinates in 2D sequence space are listed on the right): 8 external factors (the first 4 are positional factors, with 3 coordinates in 3D developmental space), only 2 of which are connected to the GRN, and 9 effectors, of which only 5 are connected, and 6 genes in 4 regulatory units.

The genome of the best stem-cap individual (Fig. 2A) codes for only three external products. To investigate their role, we have used a standard procedure used in molecular biology: knock-out experiments. Only the deletion of one of 3 morphogens (mpg3 or *capless*) had a large effect on fitness (Fig. 3A): no cap formation. Adding the third dimension to the sequence space allows to incrementally move the position of *capless* away from the plane in which all the other genes lie. This is a simple way to decrease the weight of a connection between a given gene and all the other nodes in the GRN, and corresponds to introducing point mutations as opposed to gene deletion. It can be seen (Fig. 3B) that

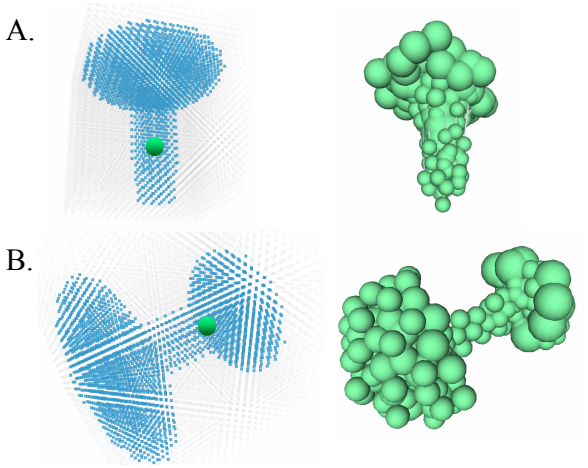


Figure 2: Difficult target shapes (left; the sphere marks the position of the first cell, dots represent target voxels): stem-cap (A) and asymmetrical dumb-bell (B), and best evolved solution phenotypes (right).

the effect of such operation on the development is "dose-dependent": the more the sequence was disturbed, the larger the effect. We can conclude that expression of *capless* allows for the development of a defined morphological structure.

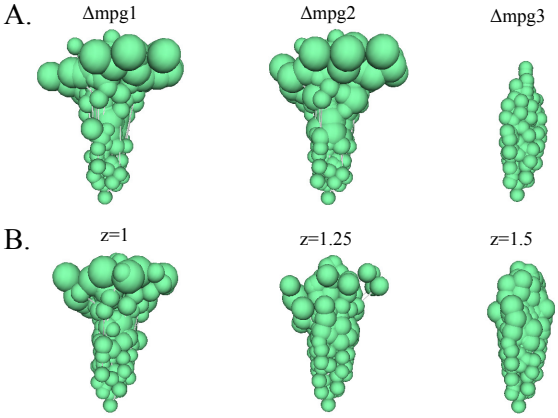


Figure 3: Mutational analysis of the best-solution individual to the stem-cap target. Only the deletion of one of 3 morphogens (mpg3 or *capless*) has a large effect on fitness (A). Shifting the location of this morphogen away from the XY 2D plane in which all the other genes lie (along the Z axis) has an incremental effect on fitness (B).

Interestingly, both the production and the perceived level of this morphogen in the developing structure is not asymmetrical along the main axis of development (Fig. 4A). It seems that all the cells produce this factor and its perceived concentration increases dramatically after the developmental step 60 when the cell number doubles (cell divisions are synchronised in the development of this individual, taking

the advantage of the division arrest mechanism). This means that it is not the asymmetry of *capless* expression that allows for the cap formation. Rather, the increase in concentration of this morphogen at step 60 causes asymmetric cell division and cell growth when the cell number doubles again after step 70. In other words, another mechanism must be used for cell differentiation along the embryo axis. We confirm this conclusion by creating an embryo in which all the cells express *capless* at a constant level (Fig. 4B).

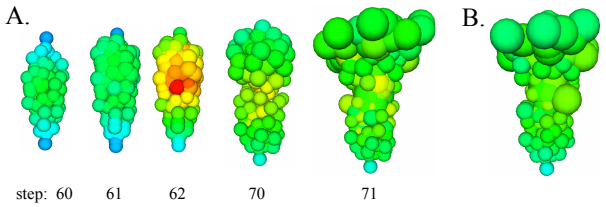


Figure 4: The perceived level of *capless* in the development of a stem-cap structure. Panel A shows the level of this morphogen in the best-solution individual at different developmental steps. Panel B: the phenotype of an individual in which all the cells produce *capless* at the same level throughout the development.

Fig. 5 presents a graph representation of a GRN controlling the development of asymmetric dumb-bell shown on Fig. 2B. It can be observed that majority of high-weight connections are inhibitory. It is also interesting that only some of the inputs (constant signal, only one spatial external factors out of four) and possible effectors are used. In other words, the development takes advantage of the changes in cell radius, internal division vector length and its rotations in two directions out of three allowed by the model. The developmental mechanism in this particular GRN does not use cell death, freezing or changes in spring length. The analysis of other GRNs evolved in our experiments also showed that only a subset of developmental mechanisms is actually needed to enable the morphogenesis of non-trivial shapes. However, quantitative analysis of hundreds of evolved GRNs would be necessary to infer any general properties of evolved networks.

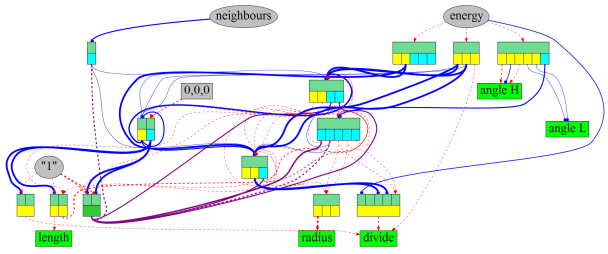


Figure 5: The GRN controlling the development of a asymmetrical dumb-bell shape in Fig. 2B. Dashed lines correspond to excitatory connections.

Discussion and Future Work

Our model extends the ideas first presented in several seminal papers by Eggenberger-Hotz who introduced GRNs with the affinity based on the similarity of real numbers (albeit in one dimension, 2003b; 2003a; 2004) and physics based on springs (2003b; 2003a). However, in this previous work cells grow on a grid and the embryo structure is further reshaped by controlling the forces between neighbouring cells. In contrast, in our model the development relies on cells of different size dividing freely in 3D space, and pushing each other away.

At the present stage, many features can be seen as unnecessary complications. Future work will show how many can be removed without compromising the ability of the evolutionary process to solve non-trivial tasks. Some parameters were included to allow future analysis of their effect on evolvability (this influenced, for example, our choice to use the insertion/deletion ratio as a way to control the genome size). At this preliminary stage, only a perfunctory analysis of the values of many parameters was possible (this applies, for example, to the thresholds in Eq. 2). Moreover, some parameters are related (for example the optimal value of the thresholds depends on the average size of mutational steps). However, the analysis of the effects of the inclusion of some parameters and their particular values on evolvability requires systematic experiments that need considerable time.

The main drain on computational resources in our experiments is frequent apparition of individuals which develop by uncontrolled cell divisions, and the energy depletion mechanism plus a limit on cell number, while keeping this problem in check, limits the potential of the artificial embryology. One of the possible solutions is to allow for a slow increase of energy in a manner that will reward slow controlled growth.

On the other hand, some features can be viewed as unnecessary simplifications. Perhaps tethering the daughter cells just to the mother cells is one of them, and since we already introduce the concept of cellular neighbourhood, attaching the cells to the neighbours (possibly taking advantage of the receptor compatibility) may provide a fruitful direction of further development (Bongard and Pfeifer, 2003; Eggenberger-Hotz, 2003b,a, 2004). Another direction is to allow dynamic changes in cell size and spring forces (in the present version both remain set after division; cf. Eggenberger-Hotz, 2004 where a similar feature allows for simple locomotion).

It would be also interesting to increase the realism of dynamics of gene expressions by introducing finite rates of change in product concentrations, represented with differential equations (e.g. as in Banzhaf, 2003; Kuo et al., 2004).

We might, however, argue that the shape of the parameter space in our model is not as complex as in the models presented previously. For example in the Eggenberger's model

each product is specified by as much as 7 different parameters. Since our model allows to compare the efficiency of the evolutionary search in gene sequence spaces of different dimensionalities, we will be able to investigate this issue in future work.

Primarily, however, we plan to go beyond the genetic algorithm as an approach to investigate the interplay between evolution and ontogeny using our model. The development of an actual artificial life setting, with competition for limited resources in a simulated world with a spatial structure (that would allow for at least temporal separation of subpopulations) will be the main objective of our further work. A genetic algorithm is a search method that allows only for a preliminary assessment of evolvability, but has obvious limitations: features like elitism, tournament selection, constant population size, fixed values of the parameters of the evolutionary process are not features of natural selection. Only an artificial life setting will allow to properly investigate such issues like the effects of the episodes of low population sizes, and the ability of the self-adapting systems that can tune their mutation rates to reach new adaptive peaks. We believe that only in such a setting some evolutionary questions considering the robustness of the network (and the related question epistasis), the role of mutations in regulatory regions for the evolutionary innovations, and the statistical properties of the evolved GRNs can be meaningfully explored.

Acknowledgements

The computational resources used in this work were obtained through the support of the Polish Ministry of Science and Education (project N303 291234), the Tri-city Academic Computer Centre (TASK) and the Interdisciplinary Center for Molecular and Mathematical Modeling (ICM, University of Warsaw; project G33-8).

References

Banzhaf, W. (2003). On the dynamics of an artificial regulatory network. In *Advances in Artificial Life*, volume 2801 of *Lecture Notes in Computer Science*, pages 217–227. Springer Berlin / Heidelberg.

Bentley, P. (2003). Evolving fractal proteins. In *ICES 2003, Evolvable Systems: From Biology to Hardware*, volume 2606 of *Lecture Notes in Computer Science*, pages 81–92. Springer Berlin / Heidelberg.

Beurier, G., Michel, F., and Ferber, J. (2006). A morphogenesis model for multiagent embryogeny. In *Proceedings of the ALife X: The Tenth International Conference on the Simulation and Synthesis of Living Systems*, pages 84–90. MIT Press.

Blumenthal, T. (2004). Operons in eukaryotes. *Briefings in Functional Genomics and Proteomics*, 3(3):199–211.

Bongard, J. and Pfeifer, R. (2003). Evolving complete agents using artificial ontogeny. In *Morpho-functional Machines: The New Species (Designing Embodied Intelligence)*, pages 237–258. Springer-Verlag, Berlin.

Carroll, S. (2005). *Endless Forms Most Beautiful: The New Science Of Evo Devo And The Making Of The Animal Kingdom*. WW Norton & Company.

Charles, H., Mouchiroud, D., Lobry, J., Gonalves, I., and Rahbe, Y. (1999). Gene size reduction in the bacterial aphid endosymbiont, *Buchnera*. *Molecular Biology and Evolution*, 16(12):1820–1822.

Eggenberger-Hotz, P. (1997). Evolving morphologies of simulated 3D organisms based on differential gene expression. In *Proceedings of the Fourth European Conference on Artificial Life*, pages 205–213. MIT Press.

Eggenberger-Hotz, P. (2003a). Exploring regenerative mechanisms found in flatworms by artificial evolutionary techniques using genetic regulatory networks. In *The Congress on Evolutionary Computation, CEC '03*, volume 3, pages 2026–2033.

Eggenberger-Hotz, P. (2003b). Genome-physics interaction as a new concept to reduce the number of genetic parameters in artificial evolution. In *The Congress on Evolutionary Computation, CEC '03*, volume 1, pages 191–198.

Eggenberger-Hotz, P. (2004). Asymmetric cell division and its integration with other developmental processes for artificial evolutionary systems. In *Artificial Life IX: Proceedings of the Ninth International Conference on the Simulation and Synthesis of Living Systems*, pages 387–393. MIT Press.

Gerstein, M., Bruce, C., Rozowsky, J., Zheng, D., Du, J., Korbel, J., Emanuelsson, O., Zhang, Z., Weissman, S., and Snyder, M. (2007). What is a gene, post-ENCODE? History and updated definition. *Genome Research*, 17(6):669.

Jakobi, N. (1995). Harnessing morphogenesis. *Proceedings of Information Processing in Cells and Tissues*, pages 29–41.

Kumar, S. (2004). *Investigating Computational Models of Development for the Construction of Shape and Form*. PhD thesis, Department of Computer Science, University College London.

Kuo, D. P., Leier, A., and Banzhaf, W. (2004). *Evolving Dynamics in an Artificial Regulatory Network Model*, volume 3242 of *Lecture Notes in Computer Science*, pages 571–580. Springer Berlin / Heidelberg.

Prusinkiewicz, P. and Lindenmayer, A. (1996). *The algorithmic beauty of plants*. Springer-Verlag New York, Inc., New York, NY, USA.

Quayle, A. P. and Bullock, S. (2006). Modelling the evolution of genetic regulatory networks. *Journal of Theoretical Biology*, 238(4):737–753.

Shipman, R., Shackleton, M., and Harvey, I. (2000). The Use of Neutral Genotype-Phenotype Mappings for Improved Evolutionary Search. *BT Technology Journal*, 18(4):103–111.

Steiner, T., Olhofer, M., and Sendhoff, B. (2006). Towards shape and structure optimization with evolutionary development. In *Proceedings of the ALife X: The Tenth International Conference on the Simulation and Synthesis of Living Systems*, pages 70–76. MIT Press.

Wolpert, L. (1969). Positional information and the spatial pattern of cellular differentiation. *Journal of Theoretical Biology*, 25(1):1–47.

Evolving Functional Symmetry in a Three Dimensional Model of an Elongated Organism

Ben Jones[1], Yaochu Jin[2], Bernhard Sendhoff[2], Xin Yao[1]
[1]School of Computer Science, The University of Birmingham, UK
[2]Honda Research Institute Europe, Offenbach, DE
B.H.Jones@cs.bham.ac.uk

Abstract

In evolutionary–developmental biology, it is well established that neural organization is coupled to a given organism's body-plan. Many theories attempt to underpin this coupling and the transitions involved during the organism's evolution, for example the transition from radial to bilateral symmetry. Before theoretically tackling these transitions however, we felt it essential to first address, in this paper, precisely why bilateral symmetry might be advantageous for a simple eel-like agent. We find that neural architectures affording the best motor-coordinated behavior (architectures that allow directional swimming of the agent), will readily emerge in a way that is functionally–bilaterally symmetric, suggesting therefore, that bilaterally symmetrical emergence for a long elongated creature can be essential if it needs to travel over some distance.

Introduction

The symmetrical properties of animals are mixed and varied. Typically, most higher organisms are bilaterally symmetric, that is to say, they can be partitioned into both dorsal and ventral halves. By comparison, more primitive organisms are radially symmetric and it is conjectured that the bilateral properties of higher organisms evolved from such radiata – and *both* from a common ancestor – during a process of *symmetry breaking* (e.g., Meinhardt (2002)). The general consensus is that the nervous systems of said organisms evolved in a coupled fashion so that they followed suit from body-plan architectural changes. As two fundamentally different examples, both the jellyfish (a radial organism) and the flatworm (a bilateral organism) demonstrate this principle in that their nervous system architectures have clearly evolved to reflect their body-plan morphologies.

Symmetry breaking is the evolutionary process that underlies the aforementioned change in body-plan symmetry. As discussed, this change is thought to have begun with a radial ancestor. Meinhardt (2002) considers gene homology as indicative of this common ancestry. More radical is the view that bilateral organization came about when a colony of individual polyps with Cnidarian (jellyfish) characteristics came together, see e.g., Collins and Valentine (2001); Holland (2003)). Further is the *Polyp with a half nerve net* scenario, attributed to Lacalli (1996). This argues that at some point during evolutionary history, a polyp started to crawl on its side, resulting in a build-up of the nervous system tissue in its ventral half and a concordant depletion in its dorsal half.

We will pick up on the issue of symmetry and although we do not account for the above theories, we will describe a very simple framework for testing the advantage of bilateral symmetry and the associated neural network (as a model nervous system) that emerges with this advantage (if indeed there is any advantage). On the one hand, we see this as a step in determining precisely why evolution favors particular bilateral body-plan nervous system couplings. If the above theories which all inherently argue that bilateral symmetry is evolutionarily advantageous, then we should hopefully observe its advantage for a simple agent in a simple environment. On the other hand, we are interested in how information processing might be structured in novel ways. We see our approach as one that enables us to study the coupling of neural architecture to body-plan symmetry. Although we do not strictly evolve the body-plan, we can still change the symmetry for a hypothetical system of muscles; and, since we fix the locations of these muscles around particular parts of the body, they can be considered as being part of the body-plan. Accordingly, if all muscles around the model organism are evolved to play a part in movement, then we will be able to partition the body-plan into several planes of symmetry and the muscle configuration can be regarded as being radially symmetric; whilst if only opposite muscles (those on the dorsal or ventral, or left or right parts of the animat) are evolved for movement, then we can hypothetically 'cut' the agent into halves and the muscle configuration can be regarded as being bilaterally symmetric. These symmetrical properties are not pre-defined, but will rather emerge if there is any evolutionary advantage.

We are not the first to investigate the coupling between body-plan morphology and neural network controllers. There are generally two bodies of researchers that have made related investigations. The first body is inter-

ested in modeling two dimensional models of undulatory organisms with a view to establishing some kind of undulatory behavior, and the type of neural controller that can bring this about; see for instance Zheng et al. (2004), Ekeberg (1993a,b), Sfakiotakis and Tsakiris (2006), Beauregard and Kennedy (2006), Ijspeert and Kodjabachian (1999). All of these studies share the common aim of understanding locomotion from a neuroscientific perspective. The second body of researchers are less interested in neuroscience but more interested in the behavior that can be evolved. Within this body, Karl Sims is one of the earliest proponents (Sims (1994a,b)) and many others have followed suit (Eggenberger (1997); Bongard and Paul (2000); also see Taylor and Massey (2001) for an extensive review). Most of these models are three dimensional and are implemented in part with powerful graphics libraries so as to provide the required visualizations and physical embodiments.

In terms of investigating body-plan symmetry, Bongard and Paul (2000) find that more locomotively efficient agents have a tendency towards evolving bilateral symmetry. In their model, they embed the neural controller into the agent's morphology so that both co-evolve. They further forgo any developmental process since they argue that one could inherently introduce symmetry; instead, they explicitly map the neuron weights and connectivities directly. However by doing so, the synaptic strengths and interconnectivities are de-coupled which in turn constrains the overall importance of neural network morphology. We argue that encoding a network at a greater level of morphological detail, so that neuron positional information has an actual bearing on connection strength, is essential, if we are to later observe any tendencies for different neurons to aggregate together and therefore potentially demonstrate central nervous system type characteristics. The 'GasNets' developed by Husbands et al. (1998) utilise similar neuron spatial information during a process of 'gas' diffusion, in which diffusing gasses play a crucial role in neuromodulation. In comparison, our model uses spatial information to determine connection strength rather than neuromodulatory signal effect.

Of further note is the work of Downing (2007) who constructed an evolutionary–developmental model of neurogenesis to bring about directional movement for a radially symmetric five-limbed 'starfish'. The model did not indicate how neural architecture may actually be coupled to body-plan morphology however.

Our own model has been constructed to meet our aim of investigating nervous system architecture/body-plan morphology couplings. In our simulations, both of these aspects co-evolve. Our motivation for this undertaking, is to initially elucidate the 'how' of this process (beginning with this paper), and our long-term goal is to better understand the 'why'. Thus we are interested in both information processing and the underlying evolutionary process. The model's task – that of directional swimming for an eel-like agent – is one of the simplest we could think of, yet it is also highly specialised inevitably requiring very task-specific couplings. This makes the problem non-trivial.

The rest of this paper is laid out as follows. We first outline and discuss some previous models of undulatory organisms. We secondly explain our model in more detail. Thirdly, we discuss our experimental results. We finally conclude this paper.

A Model of Undulatory Locomotion

Undulatory locomotion is a type of locomotion often employed by bilaterally symmetric creatures requiring directional movement (e.g. an eel). Refer to Gillis (1996) for a description of the underlying physics. Models of this type of behavior often adopt a spring mass damper system so that the mechanics are fluid and life-like. They secondly incorporate a friction model so that the modelled organism can actually move within its simulated world. Thirdly, they usually have a control mechanism, for example, a continuous time recurrent neural network (CTRNN). A CTRNN is often employed, because it is capable of exhibiting the *central pattern generating* dynamics that are essential for coordinated movement. A central pattern generator is a type of neural network that can by the very nature of its inherent dynamics, generate patterns of activity without any external input. For an extensive exploration of CTRNN dynamics, see Psujek et al. (2006); Beer and Gallagher (1992).

One of the earliest models attempting to use central pattern generators (CPGs) to model undulatory locomotion is that by Ekeberg (1993a,b), who hand-coded them using neurophysiological data available at the time, to control a lamprey-type agent. A similar approach is taken by Zheng et al. (2004) to model leech swimming. Others within the ALife community have taken the idea further by also incorporating evolution to derive the network architectures. Ijspeert and Kodjabachian (1999) applies a developmental as well as an evolutionary process in deriving the network structure, using swimming speed and muscular contortion for the fitness evaluation and a set of production rules for the developmental process. More realistic models include those of Sfakiotakis and Tsakiris (2006) who were able to replicate some of the biological movement data observed for the *Anguilla anguilla* eel (although their model omitted evolutionary mechanism). In the model, the 'eel' would navigate by incorporating sensory input from the front part of the animat. Beauregard and Kennedy (2006) further developed a model of an undulatory lamprey that could essentially track the movement of and follow, an object. This latter work was motivated out of a need to develop more realistic swimming algorithms for the computer animation industry. Our own model is explained in the following section.

Physical model

The Animat Fig. 1(a) represents a segment of the animat constructed out of layers. For clarity, not all springs have been depicted (in reality, each block in the animat has a 'crane-like' structure of springs to prevent it from collapsing in on itself). Since the animat is three dimensional, it is possible for it to undulate in multiple directions and/or demonstrate other types of movement depending on the output neurons of the neural network model.

The equations controlling the springs apply Hooke's Law with dampening dynamics; see Table 1 for the physical parameters of our system. Given a spring with mass points p_1 and p_2 on either end, it is compressed by forcing p_1 towards p_2 and vise-versa. Using p_1 as an example, the force exerted upon it by the internal dynamics of the spring, is computed as follows:

$$\overrightarrow{F_{p_1}} = -r \cdot \overrightarrow{V_{p1}} + k \cdot d, \tag{1}$$

where r is a dampening factor, $\overrightarrow{V_{p1}}$ is the velocity of p_1, k is a spring constant defining spring torque and d is the displacement of the spring from resting length. A change in the mass point's velocity, $\overrightarrow{V_{p1}}$, is governed by a change in its acceleration, $\overrightarrow{A_{p1}}$,

$$\overrightarrow{A_{p1}}\,(t + \Delta t) = \overrightarrow{A_{p1}}\,(t) + \frac{\overrightarrow{F_{p1}} + \overrightarrow{F_{p_1}^{E}} + \overrightarrow{F_{p_1}^{W}}}{m_{p1}}, \tag{2}$$

$$\overrightarrow{V_{p1}}\,(t + \Delta t) = \overrightarrow{V_{p1}}(t) + \overrightarrow{A_{p1}}\,(t + \Delta t) \cdot dt, \tag{3}$$

where m_{p_1} is the mass of p_1 and dt is the time-step (0.05) used during the integration process (20 integration steps). Note that $\overrightarrow{F_{p_1}^{E}}$ is an external force applied to the mass point whenever the output neuron controlling its associated spring, becomes activated, and $\overrightarrow{F_{p_1}^{W}}$ represents the current environmental force yielded by the surrounding 'water'. Finally, the position of the mass point, and hence the length of the spring is updated as follows,

$$\overrightarrow{P_{p1}}\,(t + \Delta t) = \overrightarrow{P_{p1}}\,(t) + \overrightarrow{V_{p1}}\,(t + \Delta t) \cdot dt. \tag{4}$$

The above equations afford a fluid and life-like representation.

The Environment The agent's environmental niche is modelled on movement through water. To keep things simple, we rely solely on the animat's current velocity to derive the environmental water force. This is the approach taken by most researchers (e.g. Sfakiotakis and Tsakiris (2006)). For all block faces, the water force, $\overrightarrow{F_W}$, is iteratively computed and applied to each constituent mass point,

$$\overrightarrow{F^{W}} = -\frac{1}{2} \cdot \nu \cdot \delta \cdot \alpha \cdot \overrightarrow{V} \cdot \left(\vec{V}\right)^2 \tag{5}$$

On the RHS of the equation, the velocity parameter, $\vec{V}$, is squared to give an indication of 'speed'. This determines the magnitude of the velocity, together with the viscosity, υ, and the drag, δ , in determining the amount of environmental force that should be applied; α is simply the area of the animat block face ($6.4 \cdot 0.35/8$). The environmental parameters that we use are given in Table 1.

Parameter	Value
Mass point masses	20.0
Layer springs	k=200,r=10.5
Block springs 'struts'	base k=25, r=50
Block springs 'crane'	k=500,r=50
Environmental viscosity, ν	10
Environmental drag, δ	1.0
Animat block count	8
Animat length	6.4
Animat width	0.35
Neurons per block	10

Table 1: Physical Parameters. Note that k=spring constant and r=spring dampenner. Note that the spring constants of the block 'struts' are controlled by the CTRNN. The base value of 25 sets an upper bound for one of these constants. All values are reflective of trial and error.

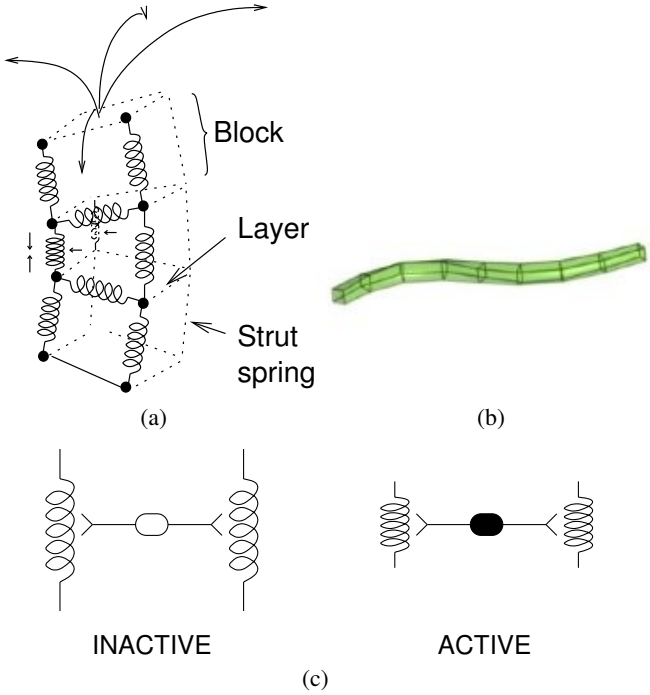


Figure 1: (a,c) Diagrams indicating how springs contract in pairs as activated by motor neurons; (b), a rendered visualization of the agent.

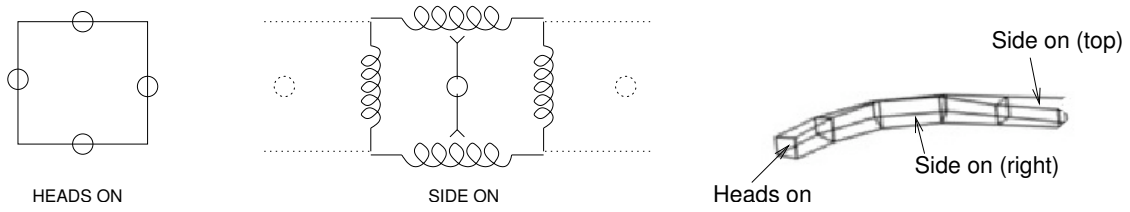


Figure 2: A diagram showing heads–on and side–on views of where the motor neurons (circles) are structurally located.

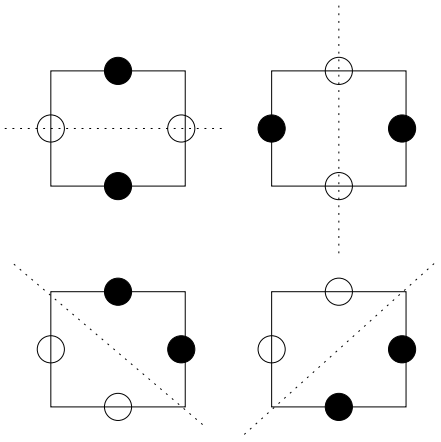


Figure 3: As additional gene values in our genome, different active motor configurations (those motors that play a part in movement) can be selected for during a process of evolution. Filled circle - active motor; dashed line - plane of symmetry. Taken from a heads–on perspective, looking down the animat from one of its ends.

Neural network implementation

A continuous time recurrent neural network (CTRNN) is employed to regulate the spring-pair compressions. The activation of a 'motor' neuron is used to calculate the spring constant of an associated spring, with a preset force of '200' but this force is only applied if the activation is between zero and one. The maximum spring constant is '25'. Each block is self-contained and houses the neural network architecture encoded for by the 'neural architecture' parameters labeled in Fig. 5. We could have chosen instead to encode a set of neural architecture parameters for each block, but this would have drastically increased the size of the search space during a process of evolution, reducing its tractability.

Weight values and inter-connectivity amongst neurons are entirely governed by neuron position. Neurons change position during evolution (because of mutation) except for the motor neurons which always reside within the centers of the block faces, see Fig. 2. Note further that the motor neurons can either bring about movement activity, or they can just serve as general interneurons. Accordingly, different 'active motor configurations' (Fig. 3) will have different impacts on the range of possible movements so are evolved along with the neural network architecture (see section 'Evolutionary Algorithm').

The membrane potential of a CTRNN neuron is computed according to its incoming pre-synaptic activity. In discrete time-steps, this activity, μ_i, of neuron i can be modelled as follows (based on Blynel and Floreano (2002)),

$$\mu_i(n+1) = \mu_i(n) + \frac{\left(-\mu_i(n) + \sum_{j=1}^{N} w_{ij} A_j(n) + I\right)}{\tau_i}, \tag{6}$$

where n is a discrete time-step and τ_i is the time constant for neuron i. The value A_j is the current output activity of presynaptic neuron j. The value I represents an external input current. Since the network never receives any 'sensory input', it has to be triggered and so we set this value to 1.0 for the first two neurons for the very first time-step of a simulation run. A neuron might also be inhibitory in which case the signs of all outgoing weights are flipped.

A weight value from neuron i to neuron j is derived according to the Euclidean distance between them, the impact of which is controlled by a parameter ξ (=2.0), Eq. 7. We also constrain the weights to fall within w^{max} (=20) and w^{min} (=0.0001), Eq. 8.

$$\lambda_{ij} = \frac{\xi}{d_{ij}}, \tag{7}$$

$$w_{ij} = \begin{cases} w^{max} & \lambda_{ij} \geq w^{max}, \\ w^{min} & \lambda_{ij} \leq w^{min}, \\ \lambda_{ij} & \text{otherwise.} \end{cases} \tag{8}$$

Connectivity Connectivity between a pair of neurons is established according to a minimum distance requirement. Hence we employ three threshold parameters. The first decides interneuron–interneuron connectivity; the second decides interneuron–effector connectivity and the third decides connectivity between neurons from a pair of contiguous subneural architectures. Since there are no sensory neurons currently employed in the model, there are no additional parameters as might be expected for more advanced architectures. A connection is formally decided as follows:

$$C_{ij} = \begin{cases} 1 & d_{ij} \leq \Gamma_q, \\ 0 & \text{otherwise.} \end{cases} \tag{9}$$

where Γ_q is a threshold parameter that we evolve, initialized to [0.01,2.5]. Note that in terms of connectivity between subnetwork architectures s_p and s_q, neuron *i* from s_p is only allowed to make one inter-subnetwork connection and that is explicitly chosen to be neuron *i* from s_q, see Fig. 4. Motor neurons never make inter-subnetwork connections.

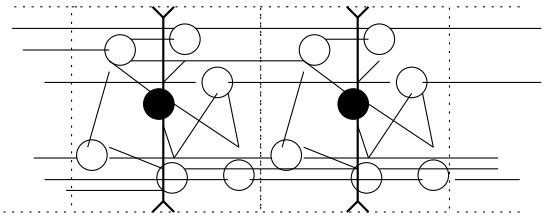


Figure 4: A diagram clarifying the repeated neuron architectures and how they are contiguously connected. Filled circles – motor neurons; unfilled circles – general interneurons. Non-dashed lines – interneuron connections.

Evolutionary Algorithm

The evolutionary algorithm optimizes the architectural parameters of the CTRNN network as described above together with the active motor configurations. These parameters form the individual's *genotype*. Both real and binary parameters are employed throughout so the algorithm employs a mixed real-valued and binary representation, see Fig. 5.

The implementation method that we employ utilizes self-adaptation of the mutation parameters. This affords a broader discovery of solutions during early evolution and a finer traversal during the later stages (e.g. Liang et al. (1998)).

Fitness measure. This is simply chosen to be the distance that the animat can move forwards during 200 time-steps. During testing of the simulation, we occasionally found that the physical interactions of the springs would oscillate out of control due to poor dampening dynamics causing either an animat 'implosion' or 'explosion'. Whenever this happened, the fitness of the individual would be set to -10000.

Mutation. All real-valued genes are mutated with values drawn from a normal distribution having an expectancy of 0 and a variance governed by the mutation parameter. This occurs for every gene with a preset probability, Φ, set to 0.02; when it occurs, the mutation parameter is also adapted. For a real-valued y-positional gene,

$$y_i = \begin{cases} y_i + N(0, \sigma_i) & \text{rand}() < \Phi, \\ y_i & \text{otherwise.} \end{cases} \quad (10)$$

whilst for a binary valued inhibitory or motor activity gene,

$$m_i = \begin{cases} !m_i & \text{rand}() < \Phi, \\ m_i & \text{otherwise.} \end{cases} \quad (11)$$

The adaptation of the mutation parameters relies on the setting of two strategy parameters, $\tau_o = 1.0/\sqrt{2D}$ and

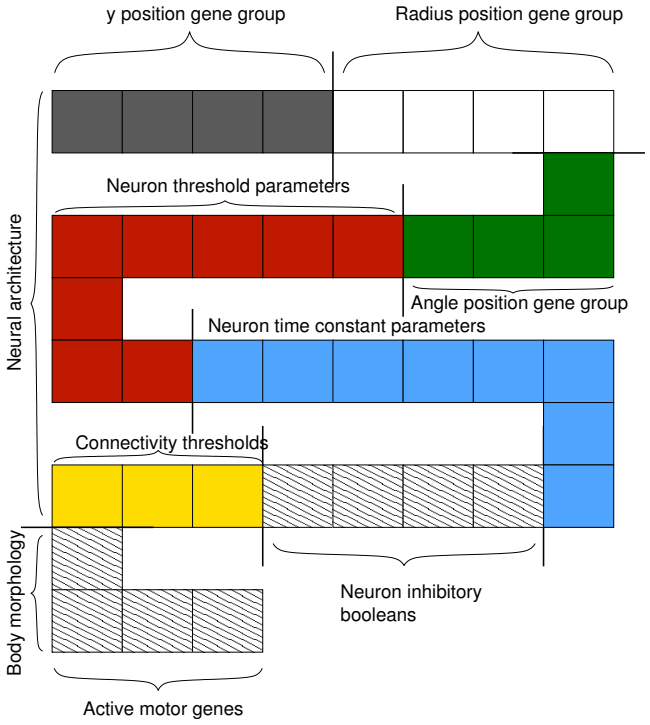


Figure 5: A representation of an individual chromosome where gene groups have been partitioned. The example is for an individual with 8 neurons per animat block. Note that in this example, there are only 4 genes per positional group because the positions of the four motor neurons remain fixed.

$\tau_1 = 1.0/\sqrt{2\sqrt{D}}$ which have been shown to be optimal in a process of self-adaptation (see Bäck and Schwefel (1993)). *D* is the dimensionality of the gene vector. Therefore, with respect to the example given in Fig. 5, *D*=4, for any of the positional groups; D=8 for the thresholds and time constants and lastly D=3 for the connectivity thresholds. The σ mutation parameters are then 'self-adapted' as shown,

$$\sigma_i \longleftarrow \sigma_i * \exp\left(N(0, \tau_0) + N_i(0, \tau_1)\right). \quad (12)$$

Crossover. All genes within a chromosome are subject to being exchanged with genes from another chromosome (*single point crossover*). This process occurs with a preset probability, χ, set to 0.2; when it occurs the mutation parameters are also crossed over between the same two chromosomes. Note that candidates for this operation are pulled out from the population at random, up to the size of the population. For any gene (both real-valued and binary types) the crossing over process can be summarized as follows:

$$\langle y_i, y_j \rangle = \begin{cases} \langle y_j, y_i \rangle & \text{rand}() < \chi, \\ \langle y_i, y_j \rangle & \text{otherwise.} \end{cases} \quad (13)$$

Selection. In our scheme we use binary tournament selection with an elitist strategy. To begin with, we rank the

population (of size 100) according to fitness and pick an elite number of individuals (=6) to form the start of the offspring. The remaining offspring population is then chosen randomly using binary tournament selection in which (until the offspring population reaches the population size), two population members are picked at random and the fittest is chosen with a preset probability (=0.9). Except for the elitist, all members are then subjected to the above mutation and crossing over operations. We use binary tournament selection since it facilitates diversity.

Results

Fig. 7 shows the progression of best fitness for a simulation run. Given the active motor configurations annotated on to the plot, we can see that fitter individuals favor a bilaterally symmetric configuration (motors opposite each other). This optimal configuration of up/down or left/right active motor configurations was found to emerge in six out of six simulation runs (results omitted). These configurations evolved with the neural network architecture as shown in Fig. 8. The neural network dynamics of the active motor neurons are given in Fig. 9. Interestingly, we can see that whilst all blocks demonstrate a variety of mostly CPG dynamics, only the first, second and sixth show both active neurons to have CPG dynamics. Blocks three, four, five, seven and eight all only show one of the active motor neurons to have CPG dynamics (either A or B); the other neuron for one of these blocks has an activity that is shown to trail off towards a negative value. Furthermore, this active motor neuron is seen to alternate in successive blocks for which only one of the neurons is active: block three's active motor neuron is motor neuron 'A', whilst block four's is 'B' and then block five's is 'A' again; block seven's is 'A' whilst block eight's is 'B'. These dynamics directly contribute to the movements of the animat since we know that the spring pairs for a given block compress whenever the associated motor neuron has an activity of between 0 and 1.

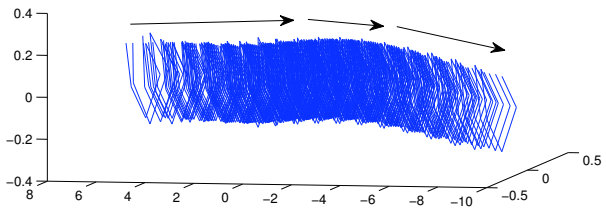


Figure 6: A motion–captured visualization of the animat travelling in the direction marked by arrows. Note, a more negative value on the lower axis indicates further forward travel.

Fig. 6 shows a motion–captured visualization of the animat travelling in the direction marked by arrows. It is difficult to observe any undulatory movements. In fact, the animat moves forward by crumpling and then extending its body segments (although there are also some very small undulatory-type movements).

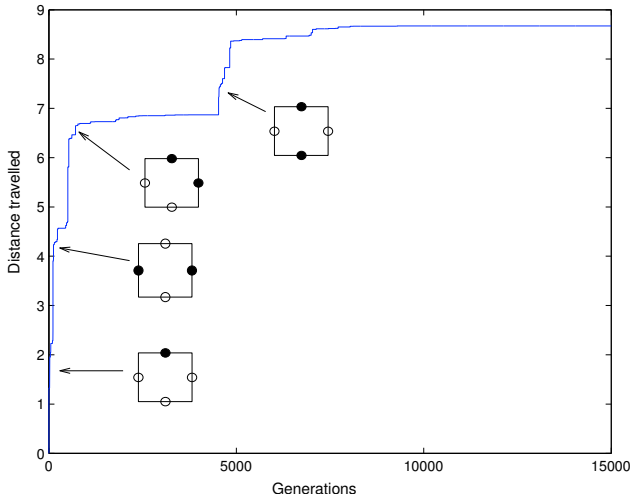


Figure 7: A graph showing the progression of the fittest population member over a simulated evolutionary period. Active motor configurations annotate different points of innovation; active motors are represented by filled circles.

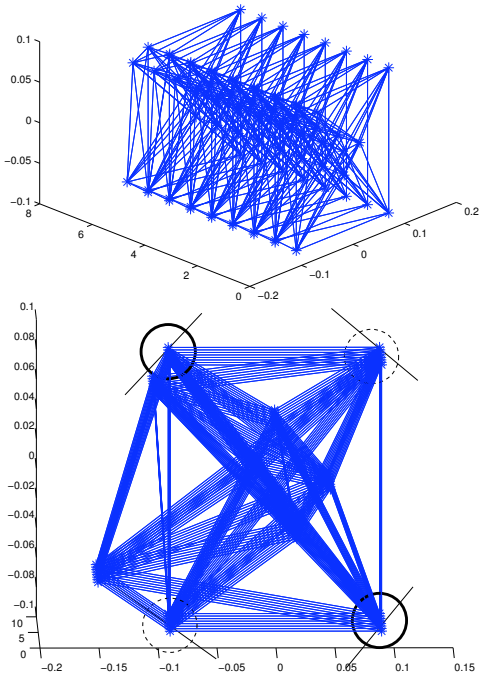


Figure 8: Visualizations of the neural network architecture from the fittest individual. In the lower visualization, the animat is shown head–on and the motor neurons that are used for movement are solidly circled (filled circles in Fig. 7); (motor neurons not used for movement – dashed circles).

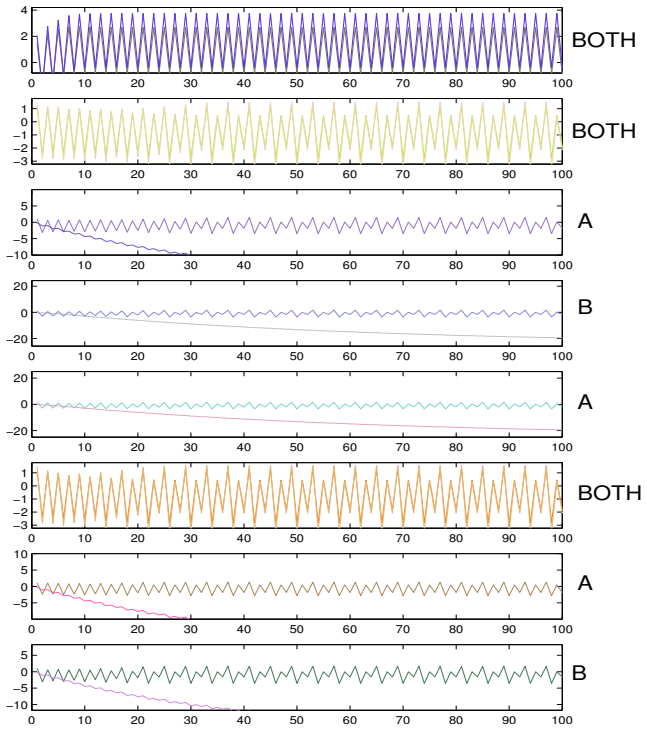


Figure 9: The CTRNN dynamics for the active motor neurons in each animat block, 1-8, each represented by a sub graph. They are labelled 'BOTH', 'A' or 'B' according to whether both active motor neurons or only one of them (A or B) demonstrate CPG dynamics.

Discussion

As noted in the introduction, nature has provided examples showing that nervous system architecture is coupled to body-plan morphology. Based on this knowledge and the hypothesis that different couplings are favored by different environments, we constructed a simple model to shed some light on how such a coupling could emerge during a process of simulated evolution.

The discovery that our simulated agent would in many ways reflect natural counterparts in terms of preferring a bilaterally symmetric motor configuration is interesting, since other than the physical characteristics of the agent (how the springs were interconnected) and the physical features of the environment (drag and viscosity), we placed no further constraints upon the movement mechanisms. Of course we intuitively know that bilateral symmetry is advantageous (consider how we walk), but the way that the control system should arrange itself – to *configure* itself – in concert with the body-plan, is less clear. The coupling is complex and the two components should not be considered separate; the architecture of the nervous system places indirect pressure on the type of body-plan morphology (configuration of motors, within our model), that can evolve and vise-versa. Our framework has helped us to elucidate this interplay of body, nervous system and environment. Our long term research goal is to better our understanding of this, especially with a fuller regard to a change in body-plan symmetry; with this paper, we have only begun to broach the subject of this major evolutionary transition.

We share the intuitive view that a change in body-plan symmetry likely occurred as organisms found themselves immersed in environments requiring directional movement. It is further our own view that this change would have been facilitated by an evolutionary drive towards those body-plan/nervous system couplings that minimize energy loss. Indeed, evolution could in part be pressured by those movement mechanisms requiring no energy. As an analogy, consider how we would compress a spring. One must of course apply energy, but upon releasing the compression, the spring passively returns to its natural resting length. Crucially, this 'self-stabilizing' process is inherent in muscle contractions and relaxations, (e.g. Pfeifer and Bongard (2006)).

In terms of our model, we can consider how simulated evolution strives to find a balance between the number of active spring compressions and the number of passive spring relaxations so that the above self-stabilizing process can be 'optimized'. If we attach an energy measure to this process, we might find that evolution prefers a maximization of passive relaxations, since this would perhaps conserve the most energy (but the springs would always have to be compressed first). However, defining such a measure will be hard, since in reality, energy can be lost from both the nervous system and from the 'muscles' and determining the levels of loss from each will directly determine the evolutionary process. Ideally, both energy losses will be coupled.

Future work

We are currently extending the model to address more fully the evolutionary transition from radial to bilateral symmetry. This will allow us to extensibly investigate the complex interactions between body, nervous system and environment and will bring us a step closer in answering why certain of these interactions emerge in a particular way. We plan to do this by (i) extending the range of morphological features; (ii) incorporating a more flexible body-plan/nervous system coupling representation; (iii) extending the flexibility of the environment so that at various stages of the simulation, specific couplings are pre-disposed.

Conclusion

In setting out to model an elongated agent that could move (e.g. swim) through water, we have shown an evolutionary preference for a bilaterally symmetric control system (a CTRNN) whose dynamics ultimately shape this movement mechanism. We further conclude that since the CTRNN architecture is coupled to the body-plan motor system, and that

movement depends on this coupling, forward movement requires a very specific coupling in order that the correct dynamics can be obtained; and moreover, evolution prefers a coupling that will, because of its inherent features, endow bilaterally symmetric functionality.

Acknowledgement

The first author is grateful to Honda Research Institute Europe for a PhD studentship that supports this work.

References

Bäck, T. and Schwefel, H.-P. (1993). An overview of evolutionary algorithms for parameter optimization. *Evolutionary Computation*, 1(1):1–23.

Beauregard, M. and Kennedy, P. J. (2006). Robust simulation of lamprey tracking. In *Parallel Problem Solving from Nature - PPSN IX*, pages 641–650, Berlin. Springer-Verlag.

Beer, R. D. and Gallagher, J. C. (1992). Evolving dynamical neural networks for adaptive behaviour. *Adaptive Behaviour*, 1(1):91–122.

Blynel, J. and Floreano, D. (2002). Levels of dynamics and adaptive behavior in evolutionary neural controllers. In *From Animals to Animats 7: Proceedings of the seventh international conference on simulation of adaptive behavior*. MIT Press.

Bongard, J. C. and Paul, C. (2000). Investigating morphological symmetry and locomotive efficiency using virtual embodied evolution. In *From Animals to Animats 6: Proceedings of the Sixth International Conference on Simulation of Adaptive Behavior*. MIT Press.

Collins, A. G. and Valentine, J. W. (2001). Defining phyla: evolutionary pathways to metazoan body plans. *Evolution and Development*, 3(6):432–442.

Downing, K. L. (2007). Supplementing evolutionary developmental systems with abstract models of neurogenesis. In *Proceedings of the 9th annual conference on Genetic and evolutionary computation*, pages 990–996. ACM.

Eggenberger, P. (1997). Evolving morphologies of simulated 3d organisms based on differential gene expression. In *Proceedings of the Fourth European Conference on Artificial Life*, pages 205–213. MIT Press.

Ekeberg, O. (1993a). An Integrated Neuronal and Mechanical Model of Fish Swimming. In *Computation of Neurons and Neural Systems (CNS)*, pages 217–222. Klumer.

Ekeberg, O. (1993b). A combined neuronal and mechanical model of fish swimming. *Biological Cybernetics*, 69:363–374.

Gillis, G. B. (1996). Undulatory locomotion in elongated aquatic vertebrates: Angulliform swimming since Sir James Gray. *American Zoology*, 36:656–665.

Holland, N. D. (2003). Early central nervous system evolution: and era of skin brains? *Nature reviews*, 4.

Husbands, P., Smith, T., and O'Shea, M. (1998). Better living through chemistry: Evolving gasnets for robot control. *Connection Science*, 10(3–4):185–210.

Ijspeert, A. and Kodjabachian, J. (1999). Evolution and development of a central pattern generator for the swimming of a lamprey. *Artificial Life*, 5(3):247–269.

Lacalli, T. (1996). Dorsoventral axis inversion: a phylogenetic perspective. *BioEssays*, 18(3).

Liang, K.-H., Yao, X., Liu, Y., Newton, C. S., and Hoffman, D. (1998). An experimental investigation of self-adaptation in evolutionary programming. In *Lecture Notes in Computer Science*, volume 1447, pages 291–300. Springer-Verlag.

Meinhardt, H. (2002). The radial-symmetric hydra and the evolution of the bilateral body plan: an old body become a young brain. *BioEssays*, 24(2002):185–191.

Pfeifer, R. and Bongard, J. C. (2006). *How the Body Shapes the Way We Think, A New View of Intelligence*. MIT Press.

Psujek, S., Ames, J., and Beer, R. D. (2006). Connection and coordination: The interplay between architecture and dynamics in evolved model pattern generators. *Neural Computation*, 18:729–747.

Sfakiotakis, M. and Tsakiris, D. (2006). Simuun: A simulation environment for undulatory locomotion. *International Journal of Modelling and Simulation*.

Sims, K. (1994a). Evolving 3d morphology and behaviour by competition. In *ALife IV Proceedings*, pages 28–39. MIT Press.

Sims, K. (1994b). Evolving virtual creatures. In *Computer Graphics, Annual Conference Series*, pages 15–22.

Taylor, T. and Massey, C. (2001). Recent developments in the evolution of morphologies and controllers for physically simulated creatures. *Artificial Life*, 7(1).

Zheng, M., Iwasaki, T., and Friesen, W. (2004). Systems approach to modeling the neuronal cpg for leech swimming. In *Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pages 703–706.

Enrichment of Interaction Rules in a String-Based Artificial Chemistry

Ciarán Kelly† and Barry McMullin and Darragh O'Brien
RINCE, Dublin City University, Glasnevin, Dublin 9, Ireland
†ciaran.kelly9@mail.dcu.ie

Abstract

In this paper, we discuss our reasoning and progress in adding a mapping between information and enzymatic function to our Molecular Classifier System (MCS). MCS takes a bottom-up approach to building artificial bio-chemical networks. Unlike Holland's LCS system, which it is loosely based on, MCS has no overt demarcation between *rules* and *messages*. In our previous work, we explored a version of this Artifical Chemistry which had an impoverished interaction scheme. While this system did present some interesting results, it had very limited potential for evolving greater complexity. We present here a mechanism for enriching the reaction rules used in our Artificial Chemistry. This mechanism is analagous to the folding of RNA to an enzymatically active form. To date, we have examined in detail the evolutionary trajectories of single reactors populated with this modified Artifical Chemistry and the results of this work are presented here.

Introduction

The field of Artificial Life was borne from the desire to understand "life" as a process, a complex dynamic system. Life, as we know it, evolved over billions of years to its current state. It is conjectured that there existed a "Last Universal Common Ancestor" (LUCA) from which every living thing descended (Forterre and Philippe, 1999). Granted, this ancestor lived a very long time ago, but to explain all the similarities between living things (e.g., DNA as a genetic molecule, the almost unique genetic code between DNA and protein etc.), the case for the existence of LUCA is strong. But, while LUCA would be an ancestor of all currently living things, it was surely not the *first* living thing. A key aspect of origin-of-life research therefore focuses on the time before LUCA—who or what were LUCA's ancestors?

The initial explorers of Artificial Life, such as Von Neumann (Von Neumann and Burks, 1966), examined the qualitative properties of life: what enables us to say that this thing is alive and this thing is not? In certain senses, the chemistry of life is now well understood. Genetic theorists since Mendel have understood the basic hereditary mechanics of life. Biochemists can already build arbitrary strands of DNA containing whatever nucleotide sequence they wish. The human genome project has sequenced and catalogued every gene in the human genome. Why, then, can we not just put together some carefully chosen pieces of DNA to create new life forms? Indeed, it would only need to be done once—the newly synthesized creatures could presumably reproduce, self-repair and evolve to cope with environmental perturbations. The answer, of course, is that there is much more to life than the sequence of monomers on a polymer, or even of genes on a chromosome. Artificial Life is, in part, the study of what that "more" is: how it can be characterised, where it comes from, and how it could be exploited it to advance human technology. We suggest that a return to the first principles of origin-of-life research may help us understand these fundamental qualities in a more abstract way.

In this context, a return to first principles means not so much replaying the tape of life (Gould, 1989), but rather examining, in detail, key stages in the evolution of living entities from non-living matter in order to abstract some *rules* that describe what life really is. Our approach is to build an Artificial Chemistry (Dittrich et al., 2001) that abstracts away from some of the "*chemical*-specific" problems, and focuses more on "*organisation*-specific" problems. That is not to say that the chemical problems are trivial, but that we want to separate the study of the organisation of simple life-forms from the specific requirments of terrestrial carbon-chemistry—"life-as-it-could-be" rather than "life-as-we-know-it" (Langton, 1989). Bedau has previously discussed the nature of life and presented the idea that life is an emergent macro-level property of systems rather than being dependant on the composition of the micro-level entities that make up these systems (Bedau, 1996, 1999). This is an important idea because it suggests a case for the exploration of digital life—life *in silico*. *In silico* experiments allow the direct study of the emergent properties of life, without the need to first solve the chemical problems that "life-as-we-know-it" has already solved.

Of course, before we can approach anything like *in silico* life, we need to be somewhat careful about how we define life in the first instance. We adopt here the definition proposed by Maynard Smith and Szathmáry (Maynard Smith

and Szathmáry, 1997)—an entity is alive if it has the properties of multiplication, variation and heredity, or if it is descended from entities which exhibit those properties. Populations of such entities which are forced to compete with one another will undergo Darwinian natural selection. This definition could clearly be applied to digital life, as it enforces no requirements on the material substance of life.

Bedau et al. (Bedau et al., 2001) have presented a list of open problems in Artificial Life. Our research is focussed on the exploration of the "Transition to Life, *in silico*", which was one of the open questions identified. Protocells are hypothesized as a transitional phase in the evolution of the biosphere (Maynard Smith and Szathmáry, 1997). In previous work we have constructed an Artificial Chemistry as a platform upon which to investigate the evolution of "computational" functionality in protocells (McMullin et al., 2007a,b). We started with a minimal "template-replicator world" in which there was only one level of Darwinian actor (the replicating "molecule"). This model incorporated the notion of unlimited heredity achieved through (catalysed) template replication of indefinite length polymers. In the simplest case we considered molecules which could act as "self-replicases"—a form of degenerate, one-element, hypercycle (Eigen and Schuster, 1977).

That work examined the inclusion of an elementary form of mutation. Molecular replication was made imperfect, with a fixed error rate per monomer (thus the molecular-level replication error rate increases with the length of the molecule).

We further introduced a simple rule for enzymatic coupling between different species (so that one species can function as replicase for another species as well as itself). This was deliberately made asymmetric. This introduced the possibility of exploitation between species. Even under the condition of hyperbolic growth[1], this allows effective displacement of a "host" species by a new "facultative parasitic" species; and, under the conditions of the model, this can happen repeatedly. In this particular model, this leads to the somewhat counter-intuitive effect of systematic, macro-evolutionary, *deterioration* in "intrinsic fitness" (as measured by replication *fidelity*).

To investigate the more interesting phenomenon of multi-level selection, populations of these molecules were injected into externally provided protocells, where protocell reproduction (by binary fission) is driven by molecular replication. By fixing the size of the protocell population, we imposed a distinct process of Darwinian selection at the higher hierarchical level of the protocell. The protocell level of selection is governed by the molecular level selection dynamic which still occurs within each protocell. We showed that the protocell level selection did effectively control parasitic exploitation at the molecular level; however, the molecular level selection is still effective in preventing positive evolution in the opposite direction (toward higher molecular-level replication fidelity). The result was a rather robust evolutionary "stalemate" in which the selectional dynamics at the two interacting levels were, in effect, precisely counteracting each other.

The system presented in these works was, of course, a radically simplified version of the phenomena that occur in real chemistry and biology. Its purpose was not to directly model such real systems. Rather it was presented as a deliberately minimal system which already illustrated how complex and counter-intuitive the evolutionary behaviour of such systems could be; but also, how the evolution could, indeed, be dramatically altered by the interaction between multiple levels of selection.

The broader intention of the current work is to develop a minimal abstract framework for understanding the evolutionary emergence of "computation" or, at least, coordinated signal processing and control, in protocellular systems. Presumably, any interesting molecular level computation must rely on a diversity of chemical species; but all of these in turn must be "replicated", directly or indirectly, to support protocell level reproduction.

The work presented in this paper addresses our progress towards the incremental widening of the repertoire of molecular interactions. Again, the broader context of this work is to explore the impact that these new interaction schemes will have on the multi-level (protocell-based) selection model, though we do not discuss such hierarchical selection in this paper.

The Molecular Classifier System

We propose a highly simplified Artificial Chemistry loosely based on John Holland's Learning Classifier Systems (Holland, 2006; Holland and Reitman, 1977), which we call the Molecular Classifier System (MCS).

The operation of our system depends on a population of "molecules", which take the form of binary strings. Each molecule has an informational structure (primary structure, or monomer sequence) and an enzymatic function ("folded" or secondary structure, or "shape"), as inspired by the ribozymes of the RNA world hypothesis (Joyce, 1991). The model also contains a rule-set which determines the enzymatic action to take, given a particular molecule. Our artificial protocells are then crudely modelled as containers for a dynamic mix of these molecules, which continuously interact and exert enzymatic actions on each other. This "informational chemistry" might then be evolved to realise some particular computation—provided that it is simultaneously capable of sustaining its own dynamic organisation. In particular, this informational or computational sub-system must grow (in absolute number of molecules) and divide in co-

[1] There is, of course, a large body of prior literature on replicator selection dynamics. We omit any extensive review here, in the interests of brevity; but (Szathmáry and Maynard Smith, 1997), for example, includes a comprehensive bibliography.

ordination with overall cell reproduction.

For the purposes of the specific model to be discussed here, the only supported enzymatic function is, by design, to make an error-prone bit-wise copy of the primary, informational, structure of the bound, substrate, molecule; that is, a *replicase* function. More specifically, if a particular molecule has the ability to bind to molecules with the same molecular structure as itself, it will effectively be able to function as a *self*-replicase.[2]

This restriction of enzymatic function to replication only is, of course, a radical simplification of any real chemistry; and, further, is a significant limitation of the potential dynamics. Additional significant simplifications are that all reaction *rates* are equal, and replication error rate (per monomer) is constant. Nonetheless, we suggest that it should be useful to fully understand the variety of selectional dynamics that are possible even in this simplified case first, before introducing the additional complications of more complex and varied enzymatic function, reaction rates etc. It is, of course, a longer term goal of the research to systematically re-introduce these more complex and realistic properties.

Model Setup

Our basic template-replicator world consists of a finite number of strings (polymers) drawn from a binary alphabet. The dynamics consists of a simple loop in which one random string is chosen as a replicase and a second as a template. If the replicase is determined to "match" the template (via a molecular transformation to be discussed later), then it "binds" to it, and replicates it, with a specific bit-wise error-rate. Another molecule is chosen at random and is replaced by the new molecule. It should be noted that there is no specific modelling of dilution flux (ϕ). If the replicase does not bind then the interaction is considered to be elastic.[3]

For the purposes of analysis we consider any specific pair of molecular species to give rise to a "binary replicase-reaction network", i.e., a network comprising the two distinct replicase-reactions that can occur between these species depending on which one functions as the enzyme and which as the substrate (or replication-template, as replication is, for the moment, the only supported enzymatic function).

In our previous work (McMullin et al., 2007a), we presented an approximate analysis of some particular binary replicase-reaction network using an appropriate set of ordinary differential equations (ODE). This allowed predictions of the concentration dynamics of a flow-reactor populated by a single pair of molecular species. We now extend that analysis to systematically examine and classify *all* possible binary replicase-reaction networks in this general type of model chemistry.

Binary Replicase-Reaction Networks

In order to construct the ODE representations of the reaction kinetics, we first derive a set of Binary Replicase-Reaction Networks classes. These classes are generic in the sense that any MCS-like system can be represented by them. They are constructed by considering the reaction kinetics: two molecules are chosen and a reaction is attempted. If we represent this scheme as a logical truth-table, we can easily enumerate and classify all possible such networks. The truth-table is constructed by considering two distinct molecular species. Each molecule may, or may not be a self-replicase—i.e., it may or may not be able to bind to copies of itself. At the same time, each molecule may or may not be able to act as a replicase for the other molecule—i.e., it may or may not be able to bind to copies of the other molecule. Taking all of these possibilities into consideration, there are 16 possible truth-tables which represent every possible combination of two molecules and two reaction rules. Allowing for certain symmetries and equivalences, these reduce to set of 10 properly *distinct* tables.

Any specific binary replicase-reaction network can be represented as follows:

$$\begin{bmatrix} (XX) & (XY) \\ (YX) & (YY) \end{bmatrix}$$

where a 1 in the (XX) position means "X is a self-replicase" and a 0 means that "X is not a self-replicase". Similarly, a 1 in the (XY) position means "X can replicate Y" and a 0 means that "X can not replicate Y".

In (McMullin et al., 2007a) we showed that it was possible to formulate an approximate differential equation model of this system. We consider two species (X and Y). Taking their respective relative concentrations as x and y, these are also the probabilities of choosing an instance of either species at random. As an example, assume X is a self-replicase. The probability of choosing two X molecules and the offspring displacing a Y molecule is evidently x^2y. Thus, the growth rate[4] of x is given by:

$$\dot{x} = x^2 y$$

Of course, this a deterministic approximation using continuous concentration values; real implementations will have discrete numbers of each molecular species and the dynamics will be stochastic. Nontheless, this ODE analysis should provide a qualitative baseline for the expected dynamic behaviour, at least as long as significant numbers of each species are present.

[2] To our knowledge, no real RNA self-replicase has yet been identified, though the conjectured existence of such molecules plays a core role in the RNA-world hypothesis.

[3] The system is therefore generically a "catalytic reaction network" in the sense of (Stadler et al., 1993).

[4] In this and subsequent equations there is an implicit multiplicative constant, effectively setting the time scale. This has been arbitrarily taken as unity.

By applying this method and discarding all the reactions which have zero effect on the concentrations, we can convert the truth-tables into differential equations. For this initial analysis we are neglecting mutation, so y, the concentration of Y, will trivially be $(1-x)$, and $\dot{y}$ will be $(1-\dot{x})$. In each case we therefore explicitly provide just the expression for $\dot{x}$.

The following terminology will be used when presenting the binary replicase-reaction networks:

- ***Sterile*** molecules can neither replicate themselves nor be replicated by another molecule.
- ***Self-Replicase*** molecules can replicate themselves, but cannot be replicated by another molecule.
- ***Obligate Parasite*** molecules cannot replicate themselves, but *can* be replicated by another molecule.
- ***Facultative Parasite*** molecules can both replicate themselves and be replicated by another molecule.

Once the relevant differential equations have been extracted for each binary replicase-reaction network we can make predictions about the flow-reactor dynamics that each network gives rise to.

Class 0:

$$\begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} = 0$$

The two molecular species are ***Sterile***. The ODE representing growth rate for both species is therefore, trivially, 0.

Class 1:

$$\begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} = 0$$

The two molecular species are ***Facultative Parasite***s. In this way, there is full cross-catalysis between the molecules, but since neither molecule has a distinct advantage over the other, the growth rate ODE for both species is again 0.

Class 2:

$$\begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} = -y^2x \quad \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} = x^2y$$

One molecule is a ***Self-Replicase*** and the other is a ***Sterile*** molecule. As would be expected, the ODE analysis states that the ***Self-Replicase*** will displace the ***Sterile*** molecule.

Class 3:

$$\begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix} = y^2x \quad \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} = -x^2y$$

One molecule is an ***Obligate Parasite*** and the other is a ***Sterile*** molecule. The ODE here shows that the ***Obligate Parasite*** will displace the ***Sterile*** molecule. As the concentration of the ***Sterile*** molecule decreases, so too does overall reaction rate (in the limit, when the final ***Sterile*** molecule is eventually eliminated, there will be no further reactions at all).

Class 4:

$$\begin{bmatrix} 0 & 0 \\ 1 & 1 \end{bmatrix} = 0 \quad \begin{bmatrix} 1 & 1 \\ 0 & 0 \end{bmatrix} = 0$$

One molecule is a ***Self-Replicase*** and the other is an ***Obligate Parasite***. In this case, a 0 growth rate is indicated by the ODE. This is explained by the fact that, by our reaction kinetics, the ***Self-Replicase*** will replicate a copy of itself exactly as often as replicating a parasite molecule.

Class 5:

$$\begin{bmatrix} 0 & 1 \\ 0 & 1 \end{bmatrix} = -x^2y - y^2x \quad \begin{bmatrix} 1 & 0 \\ 1 & 0 \end{bmatrix} = x^2y + y^2x$$

One molecule is a ***Facultative Parasite*** and the other is a ***Sterile*** molecule. Again, it is easy to see that the ***Sterile*** molecule will be completely displaced, but the reaction rate then continues at the maximum level (albeit with no further change in concentration).

Class 6:

$$\begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} = x^2y \quad \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} = -y^2x$$

One molecule is a ***Self-Replicase*** and the other is a ***Facultative Parasite***. The ODE analysis for this situation shows that the ***Facultative Parasite*** will completely displace the ***Self-Replicase***. This was the generic case considered in detail in (McMullin et al., 2007a).

Class 7:

$$\begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix} = -x^2y \quad \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix} = y^2x$$

One molecule is a ***Facultative Parasite*** and the other is an ***Obligate Parasite***. In this case the ***Facultative Parasite*** will always displace the ***Obligate Parasite***.

Class 8:

$$\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} = -x^2y + y^2x$$

Both molecular species are ***Obligate Parasite***s. This essentially means Class 8 networks are two-component hypercycles. Neither species can replicate itself but each can catalyse the replication of the other. The concentration of each species will therefore be maintained at exactly equal levels.

Class 9:

$$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} = x^2y - y^2x$$

Both molecular species are independent ***Self-Replicase***s; the "survival of the common" applies, so that whichever initially achieves a significantly higher concentration will then completely displace the other. Again, this case was detailed in (McMullin et al., 2007a).

Molecular Binding Rules

Bit-Wise Substring Binding

The binding rules that the system uses are now discussed. In (McMullin et al., 2007a), we explored perhaps the simplest

binding-rule—bit-for-bit sub-string matching with no distinction or mapping between informational (primary) structure and enzymatic (secondary) structure. If the replicase exactly matched the template in sequence, it was assumed to bind to it. This meant that the selection of binary replicase-reaction networks that could be observed was considerably smaller than the total number of networks. The first thing to notice is that every molecule is a self-replicase, and for two molecules of the same length, the only possible replicase-reaction network is the one named "Class 9" above, since it is logically impossible for two non-similar strings of the same length to be sub-strings of each other. The interesting results we achieved during that work were due to what we now call "Class 6" networks. As described above, a "Class 6" network is a network consisting of a ***Self-Replicase*** molecule and a ***Facultative Parasite***. Since the binding rule used was "bit-wise substring", the only possible way for a "Class 6" network to emerge from a mixture is if a lengthening mutation occurs during the copy of one of the templates such that the new molecule contains the "parent" molecule's structure as a sub-string.

We showed that in all cases (modulo statistical fluctuations), the "parasite" could invade a population of altruistic hosts—a pathology characterised by progressive lengthening of the average molecular string length over macro-evolutionary time. Of course, the key difficulty here is that since we are using a per-bit mutation rate, the "per-molecule" mutation rate will increase with molecular length. In a single-reactor the effect manifested as a reduced reaction rate—a lengthening of the time between successful reactions. This was due to the difficulty in finding two matching molecules to react: as mutation rate increases, so does the relative population of mutants. However, our protocell level experiments yielded an even more distinctive result. Once we added the new level of selection, at the protocell level, the parasitic behaviour of the interacting molecules was effectively halted. This was a direct consequence of hierarchical selection. The explanation is that lineages of protocells which have higher reaction rates will do better, on average, than those with lower reaction rates, since the faster a protocell can grow, the more often lineages of such protocells will undergo cellular division.

The actual effect of the "multi-level" selection was that of "selectional stalemate"—average molecular length neither increased nor decreased in the protocell model. The internal molecular dynamics ensured that under no circumstances could a "shorter" (relatively speaking) molecule come to dominate any protocell—"shorter" molecules are parasitised by "longer" ones, *"and so on, ad-infinitum"*. This tendency towards longer molecules leads to an increased mutational load on the individual protocell, and a corresponding decrease in reaction rate for that protocell.

The net effect is that, over a wide range, this system can be initialised with a protocell population with any arbitrary dominant molecular length; and the population will then remain dominated indefinitely by protocells which are individually dominated by this initial specific molecular species. Evolution toward protocells dominated by longer molecules will be prevented by the protocell level selection; and evolution toward protocells dominated by shorter molecules will be prevented by molecular level selection.

More Flexible Binding

The previous results summarised above are determined by the fact that there were really only two possible binary replicase-reactions possible with that implementation.

By opening up more reaction network possibilities, it was predicted that one could increase the variety of the system behaviour. One biochemically inspired method to go about this was to implement a mapping mechanism, similar to the folding of RNA to an enzymatically active form, that re-opens the possibility to have all possible binary replicase-reaction networks. We decided that the most incremental approach was to process the molecules in chunks of two bits, and to map these pairs into some secondary, functional, alphabet. This pair-wise processing allows for a secondary alphabet of 4 symbols. This alphabet, and the coding scheme is defined such that all 10 distinct binary replicase-reaction networks are realisable. Table 1 and Table 2 offer a comparison between the new and previous coding schemes.

Table 1: Previous Coding Scheme

chunk	function	description
0	L	match literal '0'
1	H	match literal '1'

Table 2: Enriched Coding Scheme

chunk	function	description
00	L	match literal '0'
01	L	"
10	H	match literal '1'
11	H	"

It is obvious that a molecule (bitString) which is processed by the scheme given in Table 2 will result in a functional string that is shorter than if the same molecule was processed by the scheme given in Table 1. In this version of MCS, the molecular binding rule is still bit-for-bit sub-string matching, but now, the matching happens between the functional string derived from Table 2 and the molecular bit string of the substrate molecule. In this system therefore, all 10 distinct binary replicase-reaction networks can be instantiated.

Conjecture (and Refutation) Let us assume a single-reactor with this new MCS chemistry inside. If we seed this reactor with a ***Self-Replicase*** molecule, we would once again expect this ***Self-Replicase*** to remain dominant in spite of mutation, until at some point a ***Facultative Parasite*** arises in the population and displaces it. An examination of the ODE for each of the binary replicase-reaction networks when focussing on a parasite attempting to invade the population would provide an understanding of the reasoning behind this form of hypothesis. For each reaction network we considered the growth potential of a molecule which, in combination with the seed species—the species with the highest concentration—forms such a binary replicase-reaction network. However, the ODE model would suggest that the only network where the new molecule could have a reliable advantage is that observed in a "Class 6" network. This is the same network class that allowed the parasitic take-over in the scheme represented by Table 1.

Predicted Results Our hypothesis was based upon an assumption that the reaction classes could be understood individually and that the evolutionary trajectory of a particular reactor could be predicted based on the reaction dynamics of the class that dominated the reactor. Furthermore, the ODE analysis of the reaction networks led us to believe that, once a reactor was dominated by a self-replicator, only one type of displacement event could reliably take place, namely "Class 6" ***Facultative Parasite*** driven displacement. We noted that due to the nature of the primary / secondary alphabet mapping, it was now possible to get a parasite that was shorter than its host. Further work is necessary to fully evaluate how things would be different in a multi-level, hierarchical selection situation. From our previous work (McMullin et al., 2007a), we know that hierarchical selection applies fitness pressure in the direction of better reaction rates, which results in shorter molecules. One could predict that with a parasite that is shorter than its host, the molecular level of selection and the cellular level of selection might become aligned given the correct initial conditions.

Observed Results We predicted above that a reactor, if seeded with a large number of a given self-replicase, would remain dominated by that species, at least for some reasonably extended period of time (i.e., until a facultative parasite results from mutation). In fact, it turned out that even with a low mutation rate (0.01 mutations per bit copied), a reactor seeded with a dominant replicator with some mutational copies will always result in the rapid displacement of the original seed species by a diverse variety of other species, none of them present in individually large concentrations. This result is shown in Figure 1, summarising 10 independent runs of the model.

If we further analyse one of these individual runs in more detail, we can observe that the second part of the hypothesis, that is, that the decay in concentration of the seed species would be due to the arrival of a ***Facultative Parasite***, is also flawed. Figure 2 shows the results obtained by graphing the concentration of molecules for each binary replicase-reaction network, relative to the initial seed, self-replicase, molecule. This shows that, rather than a Class 6 network emerging, Class 1 and Class 4 networks are the most prevalent contributing factors to the dilution in concentration of the seed species.

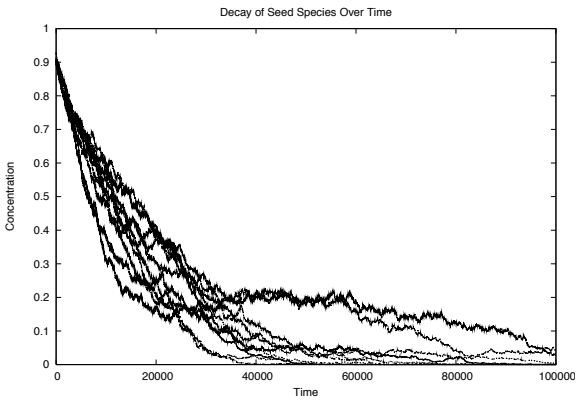


Figure 1: Concentration Decay of Seed Species

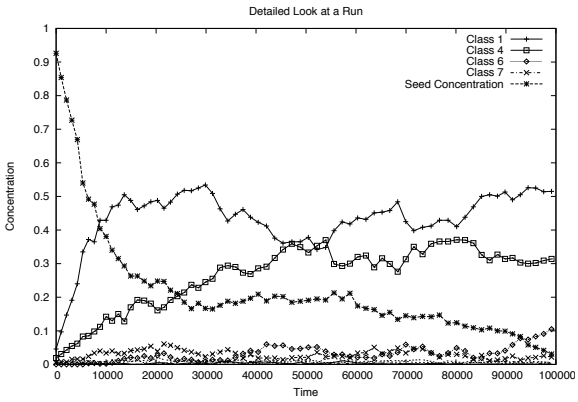


Figure 2: Detailed Analysis of a single Run

Analysis As a first step in understanding what was going on, we reviewed the earlier, ODE based, classification into 10 distinct binary replicase-reaction network classes. That analysis had suggested that the only way a molecule could reliably displace a currently dominant host was if that molecule was a ***Facultative Parasite*** of the host—they share a "Class 6" relationship. The ODE analysis for all other equivalence classes suggested that there could be no "invasion-from-rarity" displacement event. We tested these predictions in isolation, by seeding reactors with only two species of molecule and with mutation disabled. In all cases, the behaviour was as predicted by the ODE model. The only

set of experiments which led to selective displacement of the seed species were those involving displacement by "Class 6", ***Facultative Parasite***s. Of course, this does not explain the dynamics observed in Figure 2.

Further analysis has shown that the fate of these reactors is far more complicated. We have observed that "Class 1" mutants lead to the slow dilution of the seed species concentration. This dilution is a consequence of the asymmetric way in which mutation is applied. Our ODE analysis implicitly assumed that the rate of mutant generation from replication of the seed species would be balanced by an equal but opposite back-flow of mutant copies, since mutation rates were constant. Some brief analysis was carried out which highlighted the fact that for a given mutant offspring of the seed species, there are many possible mutational copies that could arise, compared with the one single mutational pathway back to the *specific* master species that gave rise to it. This asymmetric mutation pattern meant that there could be a consistent, nett mutational flow from the seed species into the collection of nearby mutants[5]. Further analysis showed that "Class 1" mutants arose most frequently in the nearby mutational neighbourhood of the seed species, and thus would be expected to arise most often.

Our experiments have clearly shown that the interplay between ***Facultative Parasite***s, ***Obligate Parasite***s and "Class 1" 'promiscuous' mutants can cause the decay of the seed species concentration. Upon further analysis, "Class 1" mutants arise initially, gain ground against the seed species and begin the one-way dilution of its concentration. Once the seed species begins to lose dominance the mutants become evermore involved in successful reactions so that their combined effect cannot be discounted any longer.

Conclusion & Future Work

In our previous published work, we demonstrated a system, MCS, consisting of two interdependent, opposing levels of selection. The macro-evolutionary outcome was that of selectional stalemate—the selectional pressures at each level exactly balanced. These previous experiments showed that for any appropriate set of initial conditions, the system would stabilise exactly where it began and evolutionary growth would essentially cease. In this paper however, we present some modifications to the chemical reaction rules of the MCS which allow for a richer set of interactions.

Initially we described our efforts to enrich the rules governing "chemical" interactions in the MCS. We accomplished this by taking inspiration from biochemistry, i.e., folding of RNA into an enzymatically active form. We added the concept of a more flexible binding by adding the simplest secondary, functional, structure to each molecule. This opened up the possibility of having molecular interactions that did not rely on exact substring matching between molecular bit-strings.

Our hypothesis was that a reactor could be initialised with a seed species of dominant concentration, and that *that* seed species would remain at dominant concentration until it was displaced by a "Class 6" ***Facultative Parasite***. However, our experiments have proved to be only partially successful in supporting this hypothesis. We found that the concentration of the seed species would decay, but that this decay was not necessarily associated with the arrival of a ***Facultative Parasite***. We believe that further experimentation with single-reactor dynamics is required before we attempt any experiments with a system which implements hierarchical selection.

Our work addresses some of the issues surrounding the understanding of "life-as-it-could-be" rather than what is currently examinable *in-vitro*—"life-as-it-is". Our work takes a bottom-up (ie. from level zero) approach to the simulation of evolutionary systems which appear to display obvious dynamics which may have been taken for granted until now.

[5]Mutants are considered to be "nearby" if they occur within a reasonably low *levenshtein* (string-edit) distance from the seed species

Acknowledgements

This work has received financial support under EU FP6 Integrated Project PACE (contract number 002035). The authors would also like to thank George G. Mitchell, James Decraene and Pietro Speroni di Fenizio for helpful discussions at the aLife lab that shaped the core material of this work.

References

Bedau, M. A. (1996). The nature of life. *The Philosophy of Artificial Life*, pages 332–357.

Bedau, M. A. (1999). Can unrealistic computer models illuminate theoretical biology. *Proceedings of the 1999 Genetic and Evolutionary Computation Conference Workshop Program*, pages 20–23.

Bedau, M. A., McCaskill, J. S., Packard, N. H., Rasmussen, S., Adami, C., Green, D. G., Ikegami, T., Kaneko, K., and Ray, T. S. (2001). Open Problems in Artificial Life. *Artificial Life*, 6(4):363–376.

Dittrich, P., Ziegler, J., and Banzhaf, W. (2001). Artificial Chemistries - A Review. *Artificial Life*, 7(3):225–275.

Eigen, M. and Schuster, P. (1977). The Hypercycle, A Principle Of Natural Self-Organization. *Die Naturwissenschaften*, 64(11):541–565.

Forterre, P. and Philippe, H. (1999). The Last Universal Common Ancestor (LUCA), Simple or Complex? *Biological Bulletin*, 196(3):373–377.

Gould, S. J. (1989). *Wonderful Life*. Penguin Group, London.

Holland, J. H. (2006). Studying Complex Adaptive Systems. *Journal of Systems Science and Complexity*, 19(1):1–8.

Holland, J. H. and Reitman, J. (1977). Cognitive systems based on adaptive algorithms. *ACM SIGART Bulletin*, pages 49–49.

Joyce, G. F. (1991). The rise and fall of the RNA world. *The New Biologist*, 3(4):399–407.

Langton, C. G. (1989). Artificial Life. In Langton, C. G., editor, *Artifical Life*, pages 1–47. Addison-Wesley Publishing Company, Inc., Redwood City, California. Proceedings of an interdisciplinary workshop,Los Alamos, New Mexico, September, 1987.

Maynard Smith, J. and Szathmáry, E. (1997). *The Major Transitions in Evolution*. Oxford Press.

McMullin, B., Kelly, C., and O'Brien, D. (2007a). Multi-Level Selectional Stalemate in a Simple Artificial Chemistry. *ECAL 9, Lisbon, Portugal*.

McMullin, B., Kelly, C., and O'Brien, D. (2007b). Preliminary Steps toward Artificial Protocell Computation. *International Conference on Morphological Computation, Venice, Italy*.

Stadler, P., Fontana, W., and Miller, J. (1993). Random catalytic reaction networks. *Physica D*, 63(3-4):378–392.

Szathmáry, E. and Maynard Smith, J. (1997). From Replicators to Reproducers: the First Major Transitions Leading to Life. *Journal of Theoretical Biology*, 187:555–571.

Von Neumann, J. and Burks, A. W. (1966). *Theory of Self-Reproducing Automata*. University of Illinois Press Champaign, IL, USA.

Evolution and Morphogenesis of Differentiated Multicellular Organisms: Autonomously Generated Diffusion Gradients for Positional Information

Johannes F. Knabe[1], Maria J. Schilstra[2] and Chrystopher Nehaniv[1,2]

[1]Adaptive Systems and [2]BioComputation Research Groups
Centre for Computer Science & Informatics Research
University of Hertfordshire, AL10 9AB Hatfield, United Kingdom
j.f.knabe@herts.ac.uk

Abstract

Development is the powerful process involving a genome in the transformation from one egg cell to a multicellular organism with many cell types. The dividing cells manage to organize and assign themselves special, differentiated roles in a reliable manner, creating a spatio-temporal pattern and division of labor. This despite the fact that little positional information may be available to them initially to guide this patterning. Inspired by a model of developmental biologist L. Wolpert, we simulate this situation in an evolutionary setting where individuals have to grow into "French flag" patterns. The cells in our model exist in a 2-layer Potts model physical environment. Controlled by continuous genetic regulatory networks, identical for all cells of one individual, the cells can individually differ in parameters including target volume, shape, orientation, and diffusion. Intercellular communication is possible via secretion and sensing of diffusing morphogens. Evolved individuals growing from a single cell can develop the French flag pattern by setting up and maintaining asymmetric morphogen gradients – a behavior predicted by several theoretical models.

Introduction

The development of multicellular organisms from a single fertilized egg cell has fascinated humans at least since Aristotle's speculations more than 2000 years ago (34). In the more recent past our understanding of how interacting genes direct developmental processes has greatly increased (31; 12; 34). Cell differentiation, the inducing effects of intercellular signaling, changes in cell form like contraction, the self-organizing properties of adhesion and cell sorting in animal morphogenesis (13) are among the important principles better understood now. And although every cell is controlled by a Genetic Regulatory Network (GRN), the resulting multicellular dynamics are also strongly influenced by physical constraints.

Development has also caught the attention of computer scientists. Traditionally their evolutionary algorithms (EAs) would neglect development for a relatively direct mapping from genotype to phenotype. To overcome problems with these EAs, many models have been proposed that incorporate development in some way – reviews of these are given in Stanley and Miikkulainen (28), Kumar and Bentley (21).

One of the earliest researchers looking for a theoretical explanation of how cells in a developing embryo could establish their different roles was Turing (30). He proposed a general symmetry breaking mechanism via the setting up of chemical gradients with reaction diffusion systems. Somewhat later, Wolpert (32, 33) came up with the very illustrative French flag model as an attempt to explain how morphogen gradients could give cells positional information as a general biological process. "Stem cells" placed along a given morphogen gradient would only have to read the morphogen contraction at their position and react to threshold values to decide whether they are in the blue, white or red part of the flag. Note that this assumes the existence of a gradient but does not explain how such a gradient could be set up by the cells. Jaeger and Reinitz (16) proposed a revised French flag model which to some degree takes the dynamic, feedback-driven nature of pattern formation into account.

These theoretical models have inspired the present work. From a single cell, placed in the middle of a 60×40 pixel grid, a multicellular organism has to grow. The organism not only has to span the grid but its cells also have to express different colors in different areas (blue, white, red – from left to right).

Several other "French flag" inspired implementations exist which are related to our work to varying degrees. Miller (25) evolved cells to match a 12×9 French flag pattern, with a particular focus on self repair in the evolved individuals. There was a concept of cell division, however the possibility of overwriting of neighboring cells during division might have been crucial for the outcomes. This work was extended by Federici (9), who evolved individuals to match less regular 9×6 patterns. An interesting analysis of why evolved developmental individuals are often also fault tolerant is given in follow-up work (10).

More recently, Devert et al. (8) evolved neural network controlled cells to match 32×32 pixel patterns, without division (cells would be everywhere from the beginning).They focused on robustness and an adaptable stop criterion – an individual was considered final when a stable state was

reached.
Crucially, in all these models cells were homogenous in size, namely 1 pixel on a grid each. Also, cell-cell communication was predefined as directed between nearest neighbors (with fixed neighbors).
Bongard and Pfeifer (3) and Bongard (2) use a spatial, developmental system ("Artificial Ontogeny") to create impressive critters, however their model is not very biologically faithful as the units their creatures are composed of have a pre-specfied cylindrical shape (different length possible) and complex internal elements like motoric joints. Symmetry breaking activation is induced by application of morphogens to the opposing ends of all new units.
Hogeweg (14) has used a model were cells also have spatial extent and can move relative to each other. Cell types were under control of a boolean GRN, where the GRN's expression pattern was interpreted as one of the predefined cell types. Differential gene expression was initiated by two prescheduled asymmetric cell divisions (i.e. one gene would be on in one daughter cell and off in the other). Communication was direct between nearest neighbor cells; a cell had the states of two nodes from neighboring cells as input to two of its own nodes. Unlike in the other mentioned works, selection was not for particular cell arrangements, but the number of exhibited cell types was used as fitness criterion – an evolutionary algorithm would select for individuals that expressed many different types. The cell type would determine some cell properties like adhesion (the possible adhesion values were pre-specified), so that certain multicellular forms could be observed as an evolutionary byproduct.
On the contrary, in our model there are no predefined communication channels; cells can only sense morphogen concentrations. Morphogens are excreted by cells and diffuse on the grid without preference for direction. Cells can actively aim to adopt heterogenous sizes and shapes – the importance of which for morphogenesis has been shown by other researchers, e.g. Merks et al. (23), Zajac et al. (35). A cell's GRN also individually and independently controls other cell properties like morphogen secretion and cadherin expression. So there is no *a priori* notion of cell types.

Genetic Regulatory Network (GRN) model

In (20), where our proposed GRN model was first described in its basic form, we used it to evolve single-celled biological clocks with the circadian rhythm abstracted to a sinusoidal wave or other periodic function. GRNs producing such cyclic behavior in response to various periodic environmental stimuli could easily be evolved. Reproducing the phase of their input as well as the production of the inverse or shifted phase was demonstrated[1], however in that investigation every evolutionary run had only one of these objectives. In later experiments we showed that it is possible to integrate two functionalities in one GRN instantiated in different contexts within a multicellular entity (18). However, *differentiation* was still induced by a given signal instead of evolving through the interaction of GRNs. Here GRNs of this type are used for the first time as "control units" of *spatially extended cells*. [2]
Every GRN consists of proteins and a genome made up of genes. Gene activation is controlled by regulatory sites (cis-sites or cis-modules), each composed of – possibly – several protein binding sites. Depending on the attachment of proteins to the binding sites the corresponding cis-modules positively or negatively influence the production of (not necessarily different) proteins. In molecular biology, proteins acting in such a way are called Transcription Factors (TFs). In our model all proteins are potentially regulatory and there are no restrictions on recurrence. A main difference from the `Biosys` GRN model by Quick et al. (27) is that there can be any number of cis-modules per gene and every cis-module can have any number of protein binding sites. This is to model a second, non-linear, level of regulation: Molecular biologists have found that TFs not only show additive behavior in influencing a gene's transcription. Some TFs interact with each other or even form protein-protein compounds, resulting in synergistic changes to their influence, see e.g. (Schilstra and Nehaniv; 7).In logical terms (but note that values are actually continuous) one can think of this grouping of inputs as an OR of ANDs. The AND level certainly constitutes a canalizing function in the sense of Kauffman (17) as a single zero value there causes the whole term to be zero no matter what the other stimuli are. In summary the model, as compared to previous models, is designed to facilitate the evolution of complex dynamics, coming a little closer to nature than previous models in terms of regulatory logic, where "5-10 regulatory sites are the rule that might even be occupied by complexes of proteins" (1) and non-linear synergetic effects are possible (7).
The following subsections describe our GRN model. For a more formal description and analysis of the model and its representation please see (20; 19).

Genetic Representation

The genome is represented as a string of base four digits, encoding several genes and some global parameters of the network. Digits 0 and 1 are *coding* digits that may be involved in regulation or protein coding. To differentiate between a sequence of coding bits, a cis-module boundary and a gene boundary the genetic alphabet was increased to four values, with digit 2 delimiting the end of a cis-module and digit 3 delimiting the end of a gene. In the version of the model used here there is a predefined number $2^4 = 16$ of different protein types, so that always four bits encode a protein type.

[1] For results from those experiments see also http://panmental.de/GRNclocks/.

[2] Please see the associated web page at http://panmental.de/ALifeXIflag for more results, videos, and the full source code.

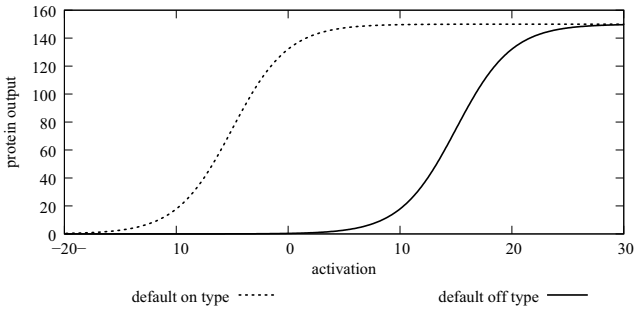


Figure 1: **Activation Types.** Every gene produces proteins according to the cumulative activation level of its cis-modules and its activation type: either even when no activation is present ("default on" - left) or only with positive activation ("default off" - right).

In the experiments described here we used a fixed number of genes, namely twenty, to facilitate analysis. After parsing the genome into genes, the last four coding digits of every gene determine its output behavior, a number of bits for the protein type produced and the last bit for the gene's activation type, which can be "default on" – active unless repressed or "default off" – silent until activated by regulatory sites,see fig. 1. The genome also encodes several evolvable variables global to the GRN. These are the *protein-specific decay rates* (four bits for every protein, indexing into a fixed look-up table of values), the global *binding proportion* (also four bits indexing into a look-up table, but identical for all proteins), and finally the global *saturation value* (three bits indexing to a look-up table, again identical for all proteins). These latter variables especially facilitate changes in the strength and timing of gene expression important in the evolution of multicellular individuals, cf. Buss (4).

Regulatory Logic

The model is run over a series of discrete time steps, its lifetime. In each time step initially a fraction of the free proteins, determined by the global binding proportion parameter, are bound to matching sites. The fraction of proteins available for binding is assigned to the binding site that has the same binary code as the protein. If there is more than one binding site competing for the same protein the fraction is equally distributed between all matching sites. In this process all protein binding sites are treated the same, regardless of the cis-module they belong to. Calculation of every gene's activation level is done by adding (activatory) or subtracting (inhibitory) the values per cis-module but only the lowest value of bound protein per cis-module is used to allow for non-linear effects. The cumulative activation level of all cis-modules then serves as input to one of two activation functions, depending on the gene's type, "default on" or "default off" as shown in fig. 1. The output of the gene's activation function is added to the unbound concentration of that gene's output protein type. After this calculation the concentrations of all unbound proteins are, if necessary, reduced to the global saturation value and all proteins, free or bound, are decayed by the protein-specific rate. Finally environmental input occurs by increasing the unbound concentration of certain proteins by some value and output by reading protein concentration values. Simple scaling is used to map stimulus input levels from the signal range to a protein concentration, and *vice versa* for output protein levels.

Spatial model

The Cellular Potts Model (CPM) has been introduced by Glazier and Graner (13), who developed it to simulate differential adhesion driven cell arrangement. Since then CPM has been used for a variety of cell-level modeling tasks, recently reviewed by Merks and Glazier (24). Although quite complex models have been realized with CPM, like the development of a cellular slime mold by Marée and Hogeweg (22), the cell level CPM simulation has not been combined with a GRN controlling cell parameters individually, without predefined cell types, before.

CPM is a two level system: On the lower level there is a grid of pixels, while the higher level cells consist of any number of lower level pixels. Cells have properties like target volume, shape, etc. and deviations from these targets incur energy penalties. Every pixel on the other hand has an integer value assigned, designating it as belonging to the cell with that identifier or zero if it is part of the "medium" (empty space). The system changes by trying to copy over one pixel's value to a randomly chosen neighbor pixel (morphogen concentration values are not copied, instead they diffuse in a separate process described below). In every time step ("Monte Carlo Step") one copy attempt is undertaken for every grid pixel on average. Copying is limited by energy constraints: Changing a pixel will change the energy properties of one (if either pixel is part of the medium) or two cells. The overall energy E is the weighted sum of all constraining properties. Copying is accepted with probability:

$$P = \begin{cases} e^{-(\Delta E - \delta)/kT} & \Delta E \geq -\delta \\ 1 & \Delta E < -\delta \end{cases}$$

In the work presented here we set offset $\delta = 0.0$, Boltzmann constant $k = 1.0$ and temperature $T = 2.0$. A two-dimensional, non-toroidal 60×40 pixel grid was used.

We use a flexible open source CPM implementation called CompuCell3D, see (6; 5) for implementation and formalism details. Through a modular plugin structure (where usually one plugin adds one energy constraint) it is easy to write extensions for the existing software (standard plugins include volume, surface, mitosis and connectivity constraints).

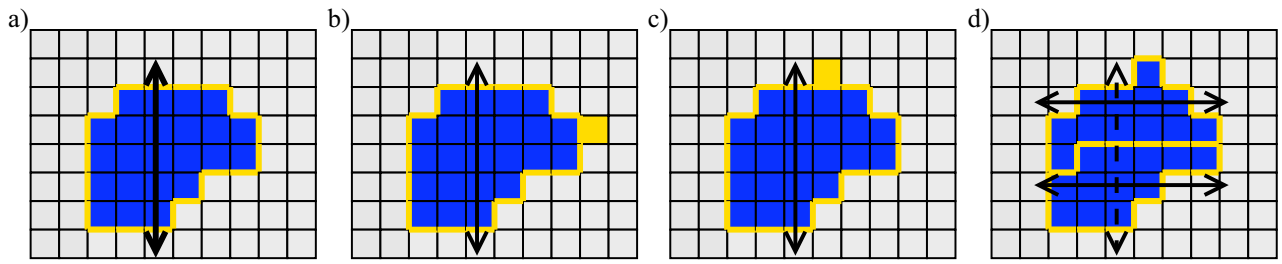


Figure 2: **Cell change example.** The blue cell in a) has volume 23 and the orientation shown by the arrow. Assuming for this cell a target ratio of 1 between the length in arrow direction and the length in the direction orthogonal to the arrow. In b) the yellow pixel is tested for becoming part of the cell. However, as this would change the ratio from 6:5 to 7:5, the transaction is energetically unfavorable. The yellow pixel in c) brings us closer to the target length ratio so it is more likely to be accepted (target volume and other constraints permitting). In d) the cell reached the mitosis volume of 24 so it split evenly along an axis orthogonal to its orientation. The daughter cells inherit an orientation rotated by 90 degrees (which can be continuously modified by GRN dynamics).

GRN controller

A GRN is able to control the cell's volume via a protein level mapping to the ratio of the size required for mitosis. So a maximal protein level meant the cell would try to grow until mitosis took place while a zero protein level would initiate shrinking and usually lead to apoptosis within a few steps. The actual volume of the cell could however differ from the target size due to neighboring cells for example. To take this and the fact that externally enforced size differences can affect the behavior of biological cells (11; 15) into account, the actual volume ratio served as input to the GRN, i.e. determined another protein's level.

Two protein levels were used to determine a cell's color. The continuous protein levels were interpreted as boolean values for this (above 0.5 threshold: true, below threshold: false). The color however does not correspond to cell type as in other models, where the type would determine cell parameters. All other parameters are controlled by the GRN independent of the color chosen. Also the color was not chosen once and for all but protein levels would be interpreted in every time step anew.

The GRN could also control the expression of "cadherins", factors that influence adhesion to other cells expressing them, i.e. it could be to some degree energetically favorable for cells with adhesion to "stick together". The adhesion strengths of the three expressible cadherin proteins were prespecified (see web page for table).

Diffusion Morphogens diffuse and decay on the underlying grid (substrate) – so it does not immediately matter for diffusion whether a cell is present or not. A cell can however increase the concentration of a morphogen on the pixels it consists of. Unlike earlier models with directed cell-cell communication mechanisms, where a cell receives as input some output (often state) of its direct neighbors, this allows for long distance communication. On the other hand meaningful communication might be harder to evolve this way as it is less directed and morphogens do not correspond to other cell variables.

The diffusion plugin "FlexibleDiffusionSolver" we used comes already with CompuCell3D and is explained (especially the numerical approximation) in its manual (6) so we only give a brief description here. The concentration c_i of morphogen i with diffusion constant d_i and decay constant k_i changes as:

$$\frac{\partial c_i}{\partial t} = d_i \nabla^2 c_i + k_i c_i + secretion_{ixy}$$

The term $secretion_{ixy}$ is the increase of morphogen i in pixel (not cell) xy. Here we used two morphogens, with the constants $d_1 = 0.2$, $k_1 = 0.009$ and $d_2 = 0.2$, $k_2 = 0.003$. A cell receives the average morphogen concentration of the pixels it consists of as input (determining one protein level per morphogen). Two other protein levels determined the secretion of the corresponding morphogen, realized as increasing the average concentration of the cell's pixels.

Cell Shape Control Apart from the plugin which manages GRN control over parameters determining cell dynamics we developed another major new module: CellShapeControl. Earlier work successfully used a plugin to model cell elongation along the cell's longest axis (23; 35). However, in these earlier works elongation followed the cell's longest axis, so orientation was due to initial random effects. In our model the orientation is partially inherited and partially under the GRN's control (see mitosis section below for details). Furthermore, better results were found during evolution when not the length directly was under GRN control but the ratio of the length along the orientation axis to the length along the orthogonal axis, see fig. 2 a)-c).

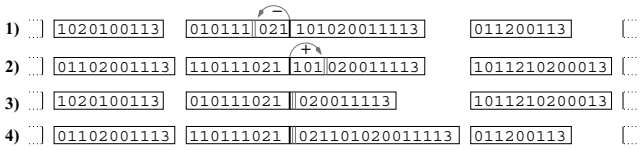


Figure 3: **Gaussian offset crossover.** Genomes of (1) parent 1, (2) parent 2, (3) offspring 1, (4) offspring 2. Only the compartment chosen for crossover and two neighboring genes are shown. Both children get digits up to the crossover point (solid bar) from their respective parent, but then continue in the other parent's genome with opposite gaussian-distributed offsets (-3 and $+3$, respectively, here).

Mitosis

Mitosis was initiated automatically once the cell reached a volume of 24 pixels, implemented as equal split along the axis orthogonal to the cell's orientation, see panel d) in fig. 2. The heritable part of the cell's orientation was shifted by 90 degrees and target volume set to half the parent's value in both daughter cells. Also, protein levels were divided equally.

Initial State In the experiments reported here we neglected "maternal factors", i.e. the first cell started without spatial extension at a size of 1 pixel, in the middle of the grid. All protein levels were set to zero for the GRN and no morphogen was present.

Evolutionary setup

A standard Genetic Algorithm with elitism, tournament selection and replacement was used. An evolutionary run lasted 250 generations containing 250–300 individuals[3]. The initial population started with one cis-module per gene and one protein binding site per cis-module, all coding bit values being randomly assigned; in network terms the nodes were randomly connected, with at most one incoming arc.

Selection

Later generations are formed by carrying over the best-performing individual of the last generation automatically and the other individuals are replaced by offspring. To generate each pair of offspring, 15 (not necessarily different) individuals of the prior generation are chosen randomly and of these the best two selected to be "parents".

[3]The variable number of individuals is due to our particular setup where clustering software Condor operated on student lab machines, so that sometimes PC usage would interrupt computations. In order not to wait for the last results to arrive we always started 500 individuals in batches of 20 but would progress to the next generation as soon as more than half of them were back. Due to batch size and step timing it could however happen that more than 260 jobs arrived before the next check.

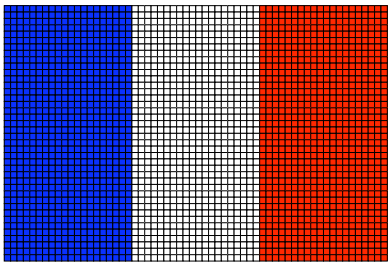


Figure 4: **French flag.** Evolutionary target was a 60×40 pixel three striped pattern.

Variability

A single-point crossover between the parent genomes occurred 90 percent of the times and every coding bit was flipped with a mutation probability of one percent. As there could be a variable number of cis- and of protein binding sites per gene their lengths will vary, so a standard bit-string crossover could change their numbers drastically. To conserve all but (at most) one of the genes as basic building units, the genomes of the parents were divided into compartments: one compartment for every gene and one compartment for the global variables. Then (with a probability of 0.9) a single compartment was chosen for crossover and in this compartment a point allocated for crossover.[4] Every pair of compartments was aligned regardless of the lengths of other compartments, as indicated in fig. 3. This process is inspired by the biological mechanism known as synapsis, the pairing of homologous chromosomes where mostly "similar" sectors pool together. To achieve variable length genes, the unequal crossing-over observed in biology is mimicked: When crossing over from parent 1's genome to the second parent's genome copying does not necessarily continue at the same position of parent 2's genome but is shifted by an offset (see fig. 3).

This offset is randomly drawn from a gaussian distributed random variable with mean 0 and standard deviation 4. The relatively large number four was chosen to increase the chance of duplicating genetic information, the importance of which was already pointed out by Ohno (26) for the evolution of biological complexity. Ohno put emphasis on whole-genome duplications while it is now, with better techniques, becoming ever clearer that "both small- and large-scale duplication events have played major roles" (29, page 320ff). Note that the offset point is limited to stay within the boundaries of the compartment, hence if crossover point + offset is smaller/larger than the left/right boundary it is set to the corresponding boundary value. So the number of 2s (cis-modules) might increase by crossover – mutation was only applied to coding digits (0s and 1s) – but not the number of 3s as these are the compartment boundaries. When

[4]This is why 'at most' one gene is changed: The crossover point could be zero or equal to the gene's coding length.

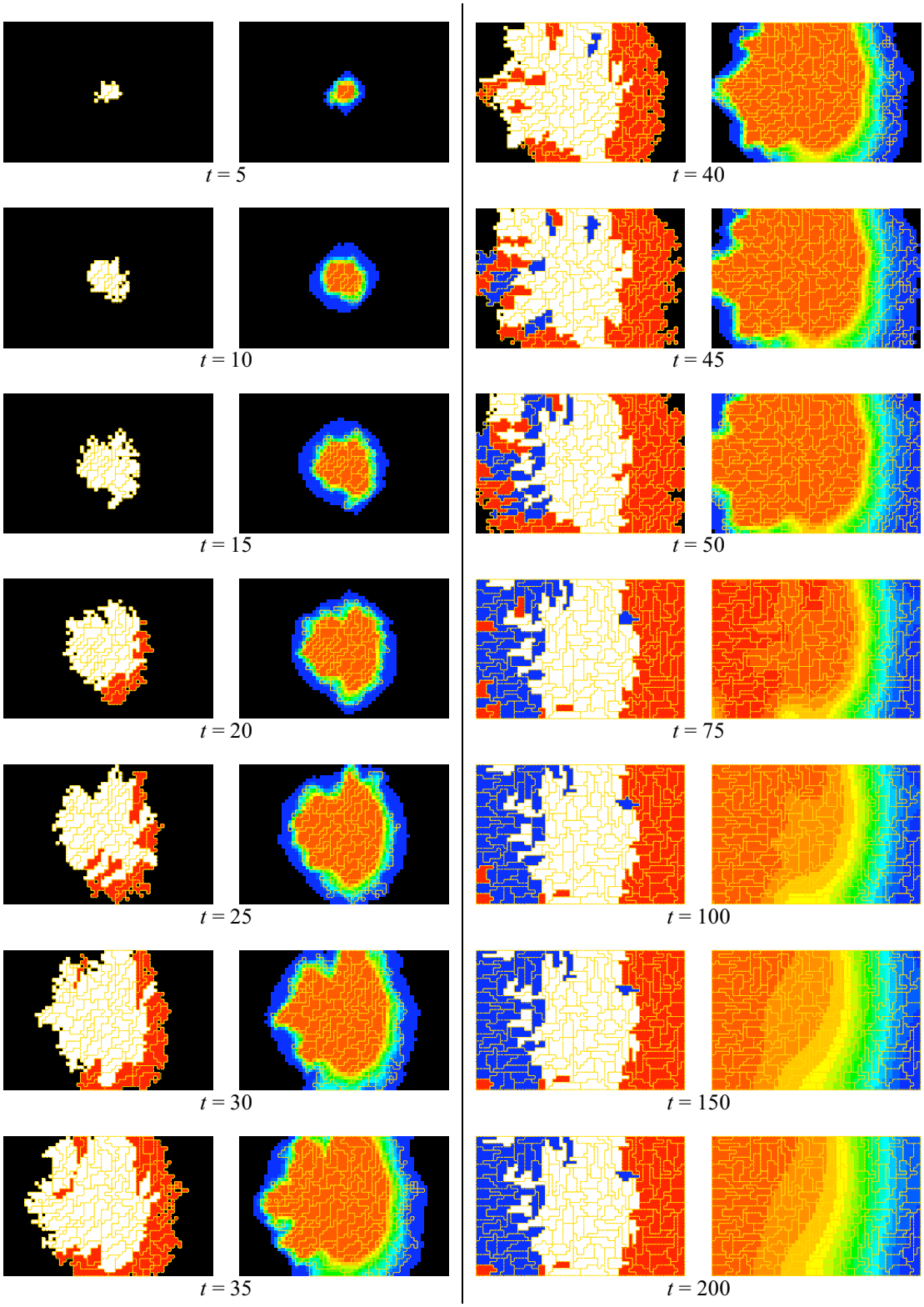


Figure 5: **Morphogenesis example.** The left picture shows the colors of the cells while the right one shows adjusted morphogen level (red=high to blue=low), respectively. Cell outlines are given in yellow, t is the time step the snapshot was taken at. See online version for color. In the left column it becomes quite clear that morphogen diffuses on the underlying grid as you can see low levels of it in areas without cells. Similarity to the target French flag pattern at $t = 200$ is 87 percent.

crossover occurs in the part encoding for global parameters the offset is always set to 0 as offsets would be meaningless here.
These processes allow both neutral crossover and mutational changes, as degenerate cis-modules (i.e. too short to bind a protein) have no regulatory effect. Additionally this means that genes could become dysfunctional, in a similar manner to the so called pseudo-genes found in nature, e.g. if there were not a single cis-module and the gene had an activation type of "off by default".

Fitness Evaluation

The lifetime of each individual was 200 time steps. Its fitness was defined simply as the match of the cell formation at the last time step compared with the 60×40 pixel target pattern comprising the entire spatial field, see fig. 4. Formally, the difference d between individual i's result pattern R^i and the target T, both of size $w \times h$, is:

$$d(R^i, T) = \frac{1}{wh} \sum_{x=0}^{w-1} \sum_{y=0}^{h-1} |sgn(R^i_{xy} - T_{xy})| \in [0, 1]$$

Note that this measure does not specify the number, size or position of cells, only the lower level pixels are taken into consideration. Accordingly, the correctness of the match or similarity s is simply $s(R^i, T) = 1 - d(R^i, T)$.
One refinement we introduced was that the difference between the patterns would be multiplied by four minus the number of colors present in the final individual. An individual using all three possible colors would get a factor of one, while individuals using fewer colors would be penalized with a higher factor. The aim of this was to "encourage" individuals early on in evolution to use all colors – without this measure runs were prone to get stuck in local maxima with huge single-color individuals.
To improve robustness, every individual was run 10 times with different random seeds and the overall fitness calculated as the average of these repetitions.

Results and Discussion

The proposed model combines for the first time spatially extended cells with GRN control over individual cell parameters like size, shape, adhesion, morphogen secretion and orientation without predefined cell types. Evolved multicellular organisms were able to autonomously set up an asymmetric morphogen gradient and organize into a close match of the French flag pattern, so the model will certainly be of good use for future research.
A majority of evolutionary runs achieved final pattern matches of over 75 percent[5]. In particular, some individuals used morphogens to set up, and maintain, a gradient across the grid, as did the example in fig. 5. The cells would usually quickly divide and spread over the grid early on during an individual's lifetime while becoming more stable later – pattern similarities at time step 210 were very close to the originally measured fitness at time step 200. A detailed assessment of intra- and extracellular dynamics is unfortunately beyond the scope of this paper.
The fact that the initial cell in the first time step had an absolute orientation of zero degrees is clearly biologically unrealistic. The reason was simply that this made assessing fitness easier. We do not think this is a major drawback as when we started the initial cell with an orientation of 90 degrees and rotated the grid and target pattern by 90 degrees as well, the similarity achieved was just as good.
Due to the inherent randomness in the spatial process we can probably not expect a perfect pattern match on such a low level – nobody expects identical twins to be identical down to the cell level. However statistical analysis of many runs is needed to quantify these effects. Also it would be desirable to investigate robustness as well as evolvability of the system. Self-repair properties after perturbations to development would be expected as for example Miller (25) observed in his work.
In the future we plan to use more natural fitness measures and to extend the evolution of morphogenesis to more complex, three dimensional forms instead of 2D patterns (as the name CompuCell3D suggests this is relatively straightforward with the software used). Also, biological cells are even more flexible and irregular in the forms they can take; for example it is known that contraction, leading to a wedge like shape, is important for gastrulation (34). It would be interesting to see if evolvability is increased further when the GRN gets more control over cell shape.

[5] Seven out of the ten runs conducted with the particular configuration of plugins and parameters described here ended with similarities above 0.75. Recall, this is the average match over 10 repetitions with different random seeds.

Acknowledgments

We would like to thank the CompuCell3D.org team for making their software available as open source project. The free distributed computing software Condor (http://cs.wisc.edu/condor/) made it possible to compute the fitness of many individuals in parallel. Also much recommended are the well documented algorithms provided by GeometryAlgorithms.com, which were very useful.

References

Banzhaf, W. (2003). On the Dynamics of an Artificial Regulatory Network. In *Advances in Artificial Life, 7th European Conference, ECAL'03*, volume 2801 of *Lecture Notes in Artificial Intelligence*, pages 217–227. Springer Verlag.

Bongard, J. (2002). Evolving modular genetic regulatory networks. In *Proc. of the Congress on Evolutionary Computation 2002*, pages 1872–1877, Washington, DC, USA. IEEE Computer Society.

Bongard, J. C. and Pfeifer, R. (2001). Repeated structure and dissociation of genotypic and phenotypic complexity in artificial ontogeny. In L. Spector et al., editor, *Proc. of the Ge-*

netic and Evolutionary Computation Conference (GECCO-2001), pages 829–836, San Francisco, CA, USA. Morgan Kaufmann.

Buss, L. W. (1987). *The Evolution of Individuality*. Columbia University Press, New York.

Cickovski, T., Aras, K., Swat, M., Merks, R. M. H., Glimm, T., Hentschel, H. G. E., Alber, M. S., Glazier, J. A., Newman, S. A., and Izaguirre, J. A. (2007). From genes to organisms via the cell: A problem-solving environment for multicellular development. *Computing in Science and Eng.*, 9(4):50–60.

CompuCell3D (2007). Software package and manual, http://compucell3d.org/.

Cornish-Bowden, A. (2001). *Fundamentals of Enzyme Kinetics*. Portland Press, London.

Devert, A., Bredeche, N., and Schoenauer, M. (2007). Robust multi-cellular developmental design. In *GECCO '07: Proc. of the 9th Annual Conference on Genetic and Evolutionary Computation*, pages 982–989, New York. ACM.

Federici, D. (2004). Increasing evolvability for developmental programs. In Miller, J., editor, *Proc. of the Workshop on Regeneration and Learning in Developmental Systems, WORLDS 2004*.

Federici, D. and Ziemke, T. (2006). Why are evolved developing organisms also fault-tolerant? In S. Nolfi et al., editor, *SAB*, volume 4095 of *Lecture Notes in Computer Science*, pages 449–460. Springer Verlag.

Folkman, J. and Moscona, A. (1978). Role of cell shape in growth control. *Nature*, 273(5661):345–349.

Gerhart, J. C. and Kirschner, M. (1997). *Cells, Embryos And Evolution*. Blackwell Publishing.

Glazier, J. A. and Graner, F. (1993). Simulation of the differential adhesion driven rearrangement of biological cells. *Phys. Rev. E*, 47(3):2128–2154.

Hogeweg, P. (2000). Shapes in the shadow: Evolutionary dynamics of morphogenesis. *Artificial Life*, 6(1):85–101.

Huang, S. and Ingber, D. E. (2000). Shape-dependent control of cell growth, differentiation, and apoptosis: Switching between attractors in cell regulatory networks. *Experimental Cell Research*, 261(1):91–103.

Jaeger, J. and Reinitz, J. (2006). On the dynamic nature of positional information. *BioEssays*, 28(11):1102–1111.

Kauffman, S. A. (1993). *The Origins of Order: Self-Organization and Selection in Evolution.* Oxford University Press.

Knabe, J. F., Nehaniv, C. L., and Schilstra, M. J. (2006a). Evolutionary robustness of differentiation in genetic regulatory networks. In Artman, S. and Dittrich, P., editors, *Proc. of the 7th German Workshop on Artificial Life 2006 (GWAL-7)*, pages 75–84. Akademische Verlagsgesellschaft Aka, Berlin.

Knabe, J. F., Nehaniv, C. L., and Schilstra, M. J. (2008). Genetic regulatory network models of biological clocks: Evolutionary history matters. *Artificial Life*, 14(1):135–148.

Knabe, J. F., Nehaniv, C. L., Schilstra, M. J., and Quick, T. (2006b). Evolving biological clocks using genetic regulatory networks. In L. M. Rocha et al., editor, *Artificial Life X: Proc. of the Tenth International Conference on the Simulation and Synthesis of Living Systems*, pages 15–21. MIT Press/Bradford Books.

Kumar, S. and Bentley, P., editors (2003). *On Growth, Form and Computers*. Elsevier Academic Press, London.

Marée, A. F. M. and Hogeweg, P. (2001). How amoeboids self-organize into a fruiting body: Multicellular coordination in *Dictyostelium discoideum*. *Proc. Natl. Acad. Sci. USA*, 98(7):3879–3883.

Merks, R. M. H., Brodsky, S. V., Goligorsky, M. S., Newman, S. A., and Glazier, J. A. (2006). Cell elongation is key to in silico replication of in vitro vasculogenesis and subsequent remodeling. *Dev Biol*, 289:44–54.

Merks, R. M. H. and Glazier, J. A. (2005). A cell-centered approach to developmental biology. *Physica A: Statistical Mechanics and its Applications*, 352(1):113–130.

Miller, J. F. (2004). Evolving a Self-Repairing, Self-Regulating, French Flag Organism. In K. Deb et al., editor, *Genetic and Evolutionary Computation - GECCO 2004*, Lecture Notes in Computer Science, pages 129–139, Berlin, Heidelberg. Springer Verlag.

Ohno, S. (1970). *Evolution by Gene Duplication.* Springer Verlag.

Quick, T., Nehaniv, C. L., Dautenhahn, K., and Roberts, G. (2003). Evolving Embodied Genetic Regulatory Network-Driven Control Systems. In *Advances in Artificial Life, 7th European Conference, ECAL'03*, volume 2801 of *Lecture Notes in Artificial Intelligence*, pages 266–277. Springer Verlag.

Schilstra, M. J. and Nehaniv, C. L. Bio-logic: Gene expression and the laws of combinatorial logic. *Artificial Life*, pages 121–134.

Stanley, K. O. and Miikkulainen, R. (2003). A taxonomy for artificial embryogeny. *Artificial Life*, 9(2):93–130.

Taylor, J. S. and Raes, J. (2005). Small-scale gene duplications. In Gregory, T. R., editor, *The Evolution of the Genome*. Elsevier Academic Press.

Turing, A. M. (1952). The chemical basis of morphogenesis. *Philosophical Transactions of the Royal Society of London. Series B, Biological Sciences*, 237(641):37–72.

West-Eberhard, M. J. (2003). *Developmental Plasticity and Evolution*. Oxford University Press.

Wolpert, L. (1969). Positional information and the spatial pattern of cellular differentiation. *Journal of Theoretical Biology*, 25:1–47.

Wolpert, L. (1996). One hundred years of positional information. *Trends in Genetics*, pages 359–364.

Wolpert, L., Jessell, T., Lawrence, P., Meyerowitz, E., Robertson, E., and Smith, J. (2007). *Principles of Development*. Oxford University Press, Oxford, 3rd edition.

Zajac, M., Jones, G. L., and Glazier, J. A. (2003). Simulating convergent extension by way of anisotropic differential adhesion. *Journal of Theoretical Biology*, 222(2):247–259.

Exploiting Open-Endedness to Solve Problems Through the Search for Novelty

Joel Lehman and Kenneth O. Stanley

School of Electrical Engineering and Computer Science
University of Central Florida, Orlando, FL, 32816
{jlehman,kstanley}@eecs.ucf.edu

Abstract

This paper establishes a link between the challenge of solving highly ambitious problems in machine learning and the goal of reproducing the dynamics of open-ended evolution in artificial life. A major problem with the objective function in machine learning is that through deception it may actually prevent the objective from being reached. In a similar way, selection in evolution may sometimes act to *discourage* increasing complexity. This paper proposes a single idea that both overcomes the obstacle of deception *and* suggests a simple new approach to open-ended evolution: Instead of either explicitly seeking an objective or modeling a domain to capture the open-endedness of natural evolution, the idea is to simply *search for novelty*. Even in an objective-based problem, such *novelty search* ignores the objective and searches for behavioral novelty. Yet because many points in the search space collapse to the same point in behavior space, it turns out that the search for novelty is computationally feasible. Furthermore, because there are only so many simple behaviors, the search for novelty leads to increasing complexity. In fact, on the way up the ladder of complexity, the search is likely to encounter at least one solution. In this way, by decoupling the idea of open-ended search from only artificial life worlds, the raw search for novelty can be applied to real world problems. Counterintuitively, in the deceptive maze navigation task in this paper, novelty search significantly outperforms objective-based search, suggesting a surprising new approach to machine learning.

Introduction

The problem of overcoming deception and local optima to find an objective in machine learning is not often linked to the goal of creating a truly open-ended dynamic in artificial life. Yet this paper argues that the same key idea addresses both challenges.

The concept of the objective function, which rewards getting closer to the goal, is ubiquitous in machine learning [22]. However, objective functions come with the pathology of local optima; landscapes from objective (e.g. fitness) functions are often deceptive [9, 21]. As a rule of thumb, the more ambitious the goal, the more likely it is that search can be deceived by local optima. The problem is that the objective function does not necessarily reward the *stepping stones* in the search space that ultimately lead to the objective.

For example, it is difficult to train a simulated biped without first suspending it from a string because it simply falls down on every attempt, obfuscating to the objective function any improvements in leg oscillation [30].

For these reasons, ambitious objectives are often carefully sculpted through a curriculum of graded tasks, each chosen delicately to build upon the prior [8, 10, 30]. Yet such incremental training is difficult and ad hoc, requiring intimate domain knowledge and careful oversight.

In contrast to the focus on objective optimization in machine learning, researchers in artificial life often study systems without explicit objectives, such as in *open-ended evolution*. An ambitious goal of this research is to reproduce the unbounded innovation of natural evolution. A typical approach is to create a complex artificial world in which there is no final objective other than survival and replication [4, 32]. Such models assume that biologically-inspired evolution supports creating an open-ended dynamic that leads to unbounded increasing complexity [3, 4, 16].

However, a growing yet controversial view in biology is that the drive towards complexity is a passive force, i.e. not driven primarily by selection [15, 18, 19]. In fact, in this view, the path towards complexity in natural evolution can sometimes be *inhibited* by selection pressure. Thus although open-endedness is often framed as an adaptive competition in artificial life worlds [3, 16], this paper decouples the idea of open-endedness from the domain by capitalizing on a simpler perspective: An open-ended evolutionary system is simply one that continually produces novel forms [25].

This perspective leads to a key idea that addresses the problems in both artificial life and machine learning: Instead of modeling natural evolution with the hope that novel individuals will be continually discovered, it is possible to search *directly* for novelty. Thus this paper introduces the *novelty search* algorithm, which searches with no objective other than continually finding novel behaviors in the search space. By defining novelty in this domain-independent way, novelty search can be applied to real world problems as directly as artificial life worlds. In fact, because there are only so many ways to behave, some of which must be more com-

plex than others [6], the passive force in nature that leads to increasing complexity is *accelerated* by searching for behavioral novelty.

To demonstrate the power of novelty search, in this paper it is compared to objective-based search in a deceptive two-dimensional robot maze navigation task. Counterintuitively, novelty search, which ignores the objective, evolves successful maze navigators that reach the objective in significantly fewer evaluations than the objective-based method. For harder mazes, the objective-based method almost always fails, while novelty search is successful in nearly every attempt. These results defy the premise in much of machine learning that the objective is the proper impetus for search.

The conclusion is that by abstracting the process through which natural evolution discovers novelty, it is possible to derive an open-ended search algorithm that applies naturally to *both* real-world machine learning problems and artificial life worlds. Novelty search overcomes the problems of deception and local optima inherent in objective optimization by ignoring the objective, suggesting the surprising conclusion that ignoring the objective in this way may often *benefit* the search for the objective.

Background

This section reviews open-endedness in natural evolution and evolutionary computation, as well as the neuroevolution method used in the experiments.

Open-endedness in Natural Evolution

Natural evolution fascinates practitioners of search because of its profuse creativity, lack of volitional guidance, and perhaps above all its drive towards complexity.

A subject of longstanding debate is the arrow of complexity [3, 19], i.e. the idea that evolutionary lineages sometimes tend towards increasing complexity. What about evolutionary search in nature causes complexity to increase? This question is important because the most difficult problems in search, e.g. an intelligent autonomous robot, may require discovering a prohibitive level of solution complexity.

The topic of complexity in natural evolution is much in contention across biology, artificial life, and evolutionary computation [15, 19, 23, 28]. One important question is whether there is a selective pressure towards complexity in evolution. A potentially heretical view that is gaining attention is that progress towards higher forms is not mainly a direct consequence of selection pressure, but rather an inevitable passive byproduct of random perturbations [15, 19]. Researchers like Miconi [19] in artificial life, and Lynch [14, 15] in biology are arguing that natural selection does not always explain increases in evolutionary complexity. In fact, they argue that to the extent that fitness (i.e. in nature, the ability to survive and reproduce) determines the direction of evolution, it can be deleterious to increasing complexity. In other words, rather than laying a path towards the next major innovation, fitness (like the objective function in machine learning) in effect prunes that very path away.

In particular, Miconi [19] points out that natural selection restricts the breadth of evolution because only designs with high fitness can be further explored. Lynch [15], a biologist, goes even further, arguing that selection pressure in general does not explain innovation, and that nonadaptive processes are often undeservedly ignored.

These arguments lead to the main idea in this paper that it may be most effective to simply search explicitly for novel behaviors.

Open-Ended Evolutionary Computation

The open-ended evolution community in artificial life aims to produce simulated worlds that allow a similar degree of unconstrained exploration as Earth. Tierra [24], PolyWorld [32] and Geb [4] are typical examples. There is no objective beyond that of survival and reproduction. The motivation behind this approach is that as evolution explores an unbounded range of life forms, complexity will inevitably increase [4, 19].

Bedau and Packard [1] and Bedau et al. [2] have contributed to formalizing the notion of unbounded open-ended dynamics by deriving a test (called *activity statistics*) that classifies evolutionary systems into categories of open-endedness. Geb and others have passed this test [4, 16], but the results nevertheless do not appear to achieve the levels of diversity or complexity seen in natural evolution. This apparent deficiency raises the question of what element is missing from current models [25]. Many suggest that more detailed, lifelike domains must be constructed to facilitate the open-ended dynamic of natural evolution [20, 25, 32].

However, this paper presents a more general approach to open-ended evolution that is motivated well by the following insight from Standish [25]: "The issue of *open-ended evolution* can be summed up by asking under what conditions will an evolutionary system continue to produce novel forms." Thus, instead of modeling natural selection, the idea in this paper is that it is more efficient to search directly for novel behaviors. While not intended to replace previous approaches to open-ended evolution, the advantage of this approach is that it decouples the concept of open-endedness from the problem domain because novelty can be sought in *any* domain. Therefore, it can apply to real-world tasks as easily as artificial life worlds.

It is important to acknowledge that this view of open-endedness contrasts with the more commonly accepted notion of prolonged production of *adaptive* traits [1, 2]. Nevertheless, the simpler view of open-endedness merits consideration on the chance that a dynamic that *appears* adaptive might be possible to capture in spirit with a simpler process.

The experiment in this paper combines this approach to open-ended evolution with the NEAT method, which is explained in the next section.

NeuroEvolution of Augmenting Topologies (NEAT)

The NEAT method was originally developed to evolve artificial neural networks (ANNs) to solve difficult control and sequential decision tasks [26, 27, 29]. Evolved ANNs control agents that select actions based on their sensory inputs. Like the SAGA method [11] introduced before it, NEAT begins evolution with a population of small, simple networks and *complexifies* the network topology into diverse species over generations, leading to increasingly sophisticated behavior. A similar process of gradually adding new genes has been confirmed in natural evolution [17, 31], and fits well with the idea of open-ended evolution.

However, a key feature that distinguishes NEAT from prior work in complexification is its unique approach to maintaining a healthy diversity of complexifying structures simultaneously, as this section reviews. Complete descriptions of the NEAT method, including experiments confirming the contributions of its components, are available in Stanley et al. [26], Stanley and Miikkulainen [27], and Stanley and Miikkulainen [29]. Let us review the key ideas on which the basic NEAT method is based.

To keep track of which gene is which while new genes are added, a historical marking is uniquely assigned to each new structural component. During crossover, genes with the same historical markings are aligned, producing meaningful offspring efficiently. Speciation in NEAT protects new structural innovations by reducing competition between differing structures and network complexities, thereby giving newer, more complex structures room to adjust. Networks are assigned to species based on the extent to which they share historical markings. Complexification, which resembles how genes are added over the course of natural evolution [17], is thus supported by both historical markings and speciation, allowing NEAT to establish high-level features early in evolution and then later elaborate on them. In effect, then, NEAT searches for a compact, appropriate network topology by incrementally complexifying existing structure.

In the experiment in this paper, NEAT is combined with novelty search, which is is explained next.

The Search for Novelty

Recall that the problem identified with the objective function in machine learning is that it does not necessarily reward the intermediate stepping stones that lead to the objective. The more ambitious the objective, the harder it is to identify *a priori* these stepping stones.

The suggested approach is to identify novelty as a *proxy* for stepping stones. That is, instead of searching for a final objective, the learning method is rewarded for finding any instance whose functionality is significantly different from what has been discovered before. Thus, instead of an objective function, search employs a *novelty metric*. That way, no attempt is made to measure overall progress. In effect, such a process performs explicitly what natural evolution does passively, i.e. gradually accumulating novel forms that ascend the complexity ladder.

For example, in a maze navigation domain, initial attempts might run into a wall and stop. In contrast with an objective function, the novelty metric would reward simply running into a *different* wall regardless of whether it is closer to the goal or not. In this kind of search, a set of instances are maintained that represent the most novel discoveries. Further search then jumps off from these representative behaviors. After a few ways to run into walls are discovered, the only way to be rewarded is to find a behavior that does not hit a wall right away. In this way, the complexity bucket fills from the bottom up. Eventually, to do something new, a navigator will have to successfully navigate the maze even though it is not an objective!

At first glance, this approach may seem naive. What confidence can we have that a search process can solve a problem when the objective is not provided whatsoever? Yet its appeal is that it rejects the misleading intuition that objectives are an essential means to discovery. The idea that the objective may be the enemy of progress is a bitter pill to swallow, yet if the proper stepping stones do not lie conveniently along its gradient, we must begin to leave behind its false security.

Still, what hope is there that novelty is any better when it contains no information about the direction of the solution? Is not the space of novel behaviors unboundedly vast, creating the potential for endless meandering? One might compare novelty search to exhaustive search: Of course a search that enumerates all possible solutions will eventually find the solution, but at enormous computational cost.

Yet there are good reasons to believe that novelty search is not like exhaustive search, and that in fact the number of novel behaviors is reasonable and limited in many practical domains. The main reason for optimism is that task domains on their own provide sufficient constraints on the kinds of behaviors that can exist or are meaningful, without the need for further constraint from an objective function.

For example, a robot navigating a maze can only do so many things; the robots in the experiments in this paper have only two effectors. Although the search space is effectively infinite because of NEAT's ability to add new genes, the *behavior space* into which points in the search space collapse is limited. For example, after an evaluation in the maze, a robot finishes at a specific location. Suppose the robot's behavior is characterized only by this ending location. While there are many ways to encode a policy that arrives at a particular point, under this measure of novelty, they *all* collapse to the *same* behavior. In fact, the search space collapses into a manageable number of novelty points, significantly differentiating novelty search from exhaustive enumeration.

Furthermore, novelty search succeeds where objective-based search fails by rewarding the stepping stones. That is, anything that is genuinely different is rewarded and pro-

moted as a jumping-off point for further evolution. While we cannot know which stepping stones are the right ones, if we accept that the primary pathology in objective-based search is that it cannot detect the stepping stones at all, then that pathology is remedied.

The next section introduces the novelty search algorithm by replacing the objective function with the novelty metric and formalizing the concept of novelty itself.

The Novelty Search Algorithm

Evolutionary algorithms like NEAT are well-suited to novelty search because the population that is central to such algorithms naturally covers a wide range of expanding behaviors. In fact, tracking novelty requires little change to any evolutionary algorithm aside from replacing the fitness function with a *novelty metric*.

The novelty metric measures how different an individual is from other individuals, creating a constant pressure to do something new. The key idea is that instead of rewarding performance on an objective, the novelty search rewards diverging from prior behaviors. Therefore, novelty needs to be *measured*.

There are many potential ways to measure novelty by analyzing and quantifying behaviors to characterize their differences. Importantly, like the fitness function, this measure must be fitted to the domain.

The novelty of a newly generated individual is computed with respect to the *behaviors* (i.e. *not* the genotypes) of an *archive* of past individuals whose behaviors were highly novel when they originated. In addition, if the evolutionary algorithm is steady state (i.e. one individual is replaced at a time) then the current population can also supplement the archive by representing the most recently visited points. The aim is to characterize how far away the new individual is from the rest of the population and its predecessors in *novelty space*, i.e. the space of unique behaviors. A good metric should thus compute the *sparseness* at any point in the novelty space. Areas with denser clusters of visited points are less novel and therefore rewarded less.

A simple measure of sparseness at a point is the average distance to the k-nearest neighbors of that point, where k is a fixed parameter that is determined experimentally. Intuitively, if the average distance to a given point's nearest neighbors is large then it is in a sparse area; it is in a dense region if the average distance is small. The sparseness ρ at point x is given by

$$\rho(x) = \frac{1}{k}\sum_{i=0}^{k} \mathrm{dist}(x, \mu_i), \quad (1)$$

where μ_i is the ith-nearest neighbor of x with respect to the distance metric *dist*, which is a domain-dependent measure of behavioral difference between two individuals in the search space. The nearest neighbors calculation must take into consideration individuals from the current population and from the permanent archive of novel individuals. Candidates from more sparse regions of this behavioral search space then receive higher novelty scores. It is important to note that this novelty space cannot be explored purposefully, that is, it is not known *a priori* how to enter areas of low density just as it is not known a priori how to construct a solution close to the objective. Thus, moving through the space of novel behaviors requires exploration. In effect, because novelty is measured relative to other individuals in evolution, it is driven by a coevolutionary dynamic.

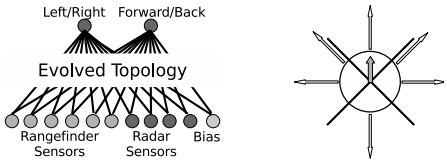
If novelty is sufficiently high at the location of a new individual, i.e. above some minimal threshold ρ_{min} , then the individual is entered into the permanent archive that characterizes the distribution of prior solutions in novelty space, similarly to archive-based approaches in coevolution [7]. The current generation plus the archive give a comprehensive sample of where the search has been and where it currently is; that way, by attempting to maximize the novelty metric, the gradient of search is simply towards what is *new*, with no other explicit objective.

It is important to note that novelty search resembles prior diversity maintenance techniques (i.e. speciation) popular in evolutionary computation. The most well known are variants of fitness sharing [5]. These also in effect open up the search by reducing selection pressure. However, in these methods, as in Hutter's fitness uniform selection [13], the heretical step of eschewing the fitness function entirely is not taken. In contrast, novelty search *only* rewards behavioral diversity with no concept of fitness or a final objective.

It is also important to note that novelty search is not a random walk; rather, it explicitly maximizes novelty. Because novelty search includes an archive that accumulates a record of where search has been, backtracking, which can happen in a random walk, is effectively avoided in behavioral spaces of any dimensionality.

The novelty search approach in general allows any behavior characterization and any novelty metric. Although generally applicable, novelty search is best suited to domains with deceptive fitness landscapes, intuitive behavioral characterization, and domain constraints on possible expressible behaviors. Changing the way the behavior space is characterized and the way characterizations are compared will lead to different search dynamics, similarly to how researchers now change the objective function to improve the search. The intent is not to imply that setting up novelty search is *easier* than objective-based search. Rather, once novelty search is set up, the hope is that it can find solutions beyond what even a sophisticated objective-based search can currently discover. Thus, the effort is justified in its returns.

Once objective-based fitness is replaced with novelty, the NEAT algorithm operates as normal, selecting the highest-scoring individuals to reproduce. Over generations, the population spreads out across the space of possible behaviors, continually ascending to new levels of complexity (i.e. by



(a) Neural Network (b) Sensors

Figure 1: **Maze Navigating Robot.** The artificial neural network that controls the maze navigating robot is shown in (a). The layout of the sensors is shown in (b). Each arrow outside of the robot's body in (b) is a rangefinder sensor that indicates the distance to the closest obstacle in that direction. The robot has four pie-slice sensors that act as a compass towards the goal, activating when a line from the goal to the center of the robot falls within the pie-slice. The solid arrow indicates the robot's heading.

expanding the neural networks in NEAT) to create novel behaviors as the simpler variants are exhausted.

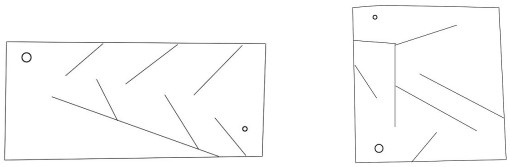
Experiment

A good domain for testing novelty search should have a deceptive fitness landscape. In such a domain, the search algorithm following the fitness gradient may perform worse than an algorithm following novelty gradients because novelty cannot be deceived; it ignores fitness entirely. A compelling, easily-visualized domain with this property is a two-dimensional maze navigation task. A reasonable fitness function for such a domain is how close the maze navigator is to the goal at the end of the evaluation. Thus, dead ends that lead close to the goal are local optima to which an objective-based algorithm may converge, which makes a good model for deceptive problems in general.

This paper's experiments utilize NEAT, which has been proven in many control tasks [27, 29], including maze navigation [26], the domain of the experiments in this paper.

The maze domain works as follows. A robot controlled by an ANN must navigate from a starting point to an end point in a fixed time. The task is complicated by cul-de-sacs that prevent a direct route and that create local optima in the fitness landscape. The robot (figure 1) has six rangefinders that indicate the distance to the nearest obstacle and four pie-slice radar sensors that fire when the goal is within the pie-slice. The robot's two effectors result in forces that respectively turn and propel the robot. This setup is similar to the successful maze navigating robots in NERO [26].

Two maps are designed to compare the performance of NEAT with fitness-based search and NEAT with novelty search. The first (figure 2a) has deceptive dead ends that lead the robot close to the goal. To achieve a higher fitness than the local optimum provided by a dead end, the robot must travel part of the way through a more difficult path that requires a weaving motion. The second maze (figure 2b) provides a more deceptive fitness landscape that requires the search algorithm to explore areas of significantly lower fitness before finding the global optimum (which is a network that reaches the goal).



(a) Medium Map (b) Hard Map

Figure 2: **Maze Navigation Maps.** In both maps, the large circle represents the starting position of the robot and the small circle represents the goal. Cul-de-sacs in both maps that lead toward the goal create the potential for deception.

Fitness-based NEAT, which will be compared to novelty search, requires a fitness function to reward maze-navigating robots. Because the objective is to reach the goal, the fitness f is defined as the distance from the robot to the goal at the end of an evaluation: $f = b_f - d_g$, where b_f is a constant bias and d_g is the distance from the robot to the goal. Given a maze with no deceptive obstacles, this fitness function defines a monotonic gradient for search to follow. The constant b_f ensures all individuals will have positive fitness.

NEAT with novelty search, on the other hand, requires a novelty metric to distinguish between maze-navigating robots. Defining the novelty metric requires careful consideration because it biases the search in a fundamentally different way than the fitness function. The novelty metric determines the behavior-space through which search will proceed. It is important that the type of behaviors that one hopes to distinguish are recognized by the metric.

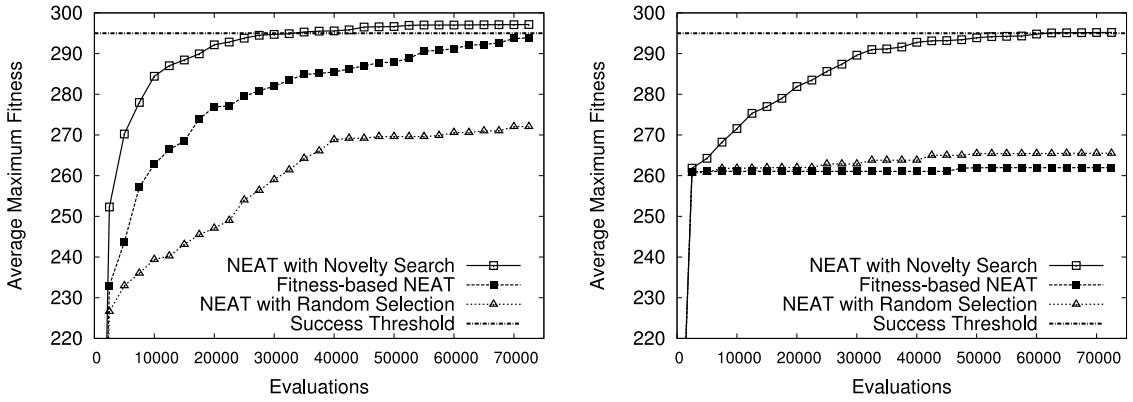
Thus, for the maze domain, the behavior of a navigator is defined as its ending position. The novelty metric is then the Euclidean distance between the ending positions of two individuals. For example, two robots stuck in the same corner appear similar, while one robot that simply sits at the start position looks very different from one that reaches the goal, though they are both equally viable to the novelty metric.

This novelty metric rewards the robot for ending in a place where none have ended before; the method of traversal is ignored. This measure reflects that what is important is reaching a certain location (i.e. the goal) rather than the method of locomotion. Thus, although the novelty metric has no knowledge of the final goal, a solution that reaches the goal will appear novel. Furthermore, the comparison between fitness-based and novelty-based search is fair because both scores are computed only based on the distance of the final position of the robot from other points.

Finally, to confirm that novelty search is indeed not anything like random search, NEAT is also tested with a random fitness assigned to every individual regardless of performance, which means that selection is random. If the maze is solved, the number of evaluations is recorded.

Experimental Parameters

Because NEAT with novelty search differs from original NEAT only in substituting a novelty metric for



(a) Medium Map (b) Hard Map

Figure 3: **Comparing Novelty Search to Fitness-based Search.** The change in fitness over time (i.e. number of evaluations) is shown for NEAT with novelty search, fitness-based NEAT, and NEAT with random selection on the medium (a) and hard (b) maps, both averaged over 40 runs of each approach. The horizontal line indicates at what fitness the maze is solved. The main result is that novelty search is significantly more effective. Only the first 75,000 evaluations (out of 250,000) are shown because the dynamics remain stable after that point.

a fitness function, it uses the same parameters. All experiments were run using a modified version of the real-time NEAT (rtNEAT) package (available from http://www.cs.utexas.edu/users/nn/keyword?rtneat) with a population size of 250. The steady-state rtNEAT evolutionary algorithm performs equivalently to generational NEAT [26]. Offspring had a 0.5% chance of adding a node, a 10% chance of adding a link, and the weight mutation power is 0.8. Parameter settings are based on standard NEAT defaults and were found to be robust to moderate variation. Runs consisted of $250,000$ evaluations, which is equivalent to $1,000$ generations of 250 individuals in a generational evolutionary algorithm.

The number of nearest neighbors checked in novelty search, k, was set to 15, and is robust to moderate variation. The minimum threshold of novelty before adding to the permanent archive of points, ρ_{min}, was initialized to 3.0, but changed dynamically: If 2,500 evaluations pass, and no new individuals have been added to the archive, the threshold is lowered by 5%. If over four are added in the same amount of evaluations, it is raised by 20%. In addition, any evaluated point has a 0.1% chance to be added to the archive.

A robot is allotted 400 timesteps to navigate through a maze. This number was chosen experimentally to make navigation more difficult; because time is limited, the robot must make efficient movements to reach the goal. The fitness bias f_b was 300.0, which ensures that a positive fitness is awarded to all individuals.

Results

On both maps, a robot that finishes within five units of the goal counts as a solution. On the medium map, both fitness-based NEAT and NEAT with novelty search were able to evolve solutions in every run (figure 3a). Novelty search took on average $18,274$ evaluations ($sd = 20,447$) to reach a solution, while fitness-based NEAT was three times slower, taking $56,334$ evaluations ($sd = 48,705$), averaged over 40 runs. This difference is significant ($p < .0001$). NEAT with random selection performed much worse than the other two methods, finding successful navigators in only 21 out of 40 runs, which confirms the difference between novelty search and random search.

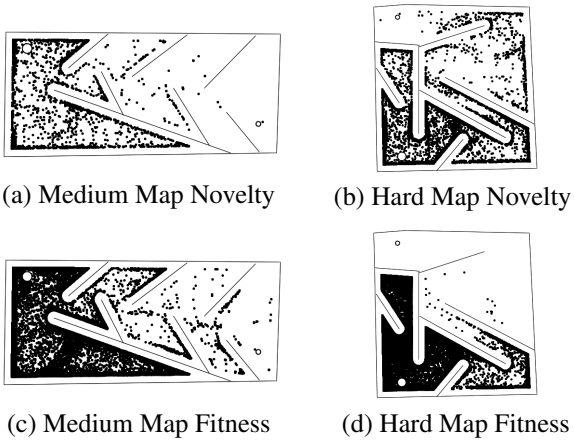
Interestingly, the genomic complexity of solutions evolved by fitness-based NEAT for the medium map (66.74 connections, $sd = 56.7$) was almost three times greater ($p < 0.05$) than those evolved by NEAT with novelty search (24.6 connections, $sd = 4.59$), even though both share the same parameters.

On the hard map, fitness-based NEAT was only able to evolve a successful navigator in three out of 40 runs, while NEAT with random selection fared marginally better, succeeding in four out of 40 runs, showing that deception in this map renders the gradient of fitness no more helpful than random search. However, novelty search was able to solve the same map in 39 out of 40 runs, in $35,109$ evaluations ($sd = 30,236$) on average when successful, using 33.46 connections on average ($sd = 9.26$). Figure 3b shows this more dramatic divergence. Remarkably, because the second maze is so deceptive, the same rtNEAT algorithm can almost never solve it when solving the maze is made the explicit objective, yet solves it almost every time when finding novel behavior is the objective!

Typical Behavior

Figure 4 depicts behaviors (represented as the final point visited by an individual) discovered during typical runs of NEAT with novelty search and fitness-based NEAT on each map. Novelty search exhibits a more even distribution of points throughout both mazes. Fitness-based NEAT shows areas of density around local optima in the maze.

The typical behavior of a successful robot on either maze was to directly traverse the maze for both methods.



(a) Medium Map Novelty (b) Hard Map Novelty

(c) Medium Map Fitness (d) Hard Map Fitness

Figure 4: **Final Points Visited Over Typical Runs.** Each maze depicts a typical run, stopping at either 250,000 evaluations or when a solution is found. Each point represents the end location of a robot evaluated during the run. Novelty search is more evenly distributed because it is not deceived.

Discussion and Future Work

Novelty search casts the performance of evolutionary algorithms in a new perspective. Based on the performance of fitness-based NEAT on the second maze, one might conclude that NEAT is no better than random search for solving this particular problem. Yet NEAT-based novelty search, which changes the reward function to *ignore* the objective while preserving the rest of the algorithm, shows that the pathology is *not* in NEAT but rather in the pursuit of the objective itself. In fact, the second maze is consistently solved by NEAT when it is given no specific objective other than to produce individuals that are different functionally from those seen before. It is thus when NEAT is charged simply with continually searching for something new that it effectively solves the problem.

However, novelty search has limitations as well; because it ignores the objective, there is no bias towards optimization once a solution is found. An optimized solution may be produced by novelty search only if an individual can appear novel by reaching such performance. However, it is likely more efficient to take the most promising results from novelty search and further optimize them based on an objective function. This idea exploits the strengths of both approaches: Novelty search effectively finds approximate solutions, while objective optimization is good for *tuning* approximate solutions. Alternatively, novelty search could be applied when a traditional evolutionary algorithm converges, to replenish diversity in the population. These ideas for combining novelty and fitness-based search will be explored in future experimentation.

While glaringly counterintuitive, the idea that the search for novelty can outperform the search for the objective introduces critical insight: Objective fitness by necessity instantiates an imposing landscape of peaks and valleys. For complex problems it may be impossible to define an objective function where these peaks and valleys create a direct route through the search space. Yet in novelty search, the rugged landscape evaporates into an intricate web of paths leading from one idea to another; the concepts of higher and lower ground are replaced by an agnostic landscape that points only along the gradient of novelty.

This idea further hints at a novel perspective on open-endedness that is fitness-agnostic. Rather than viewing open-ended evolution as an adaptive competition, it can be viewed simply as a passive drift through the lattice of novelty. As Lynch [15] and Miconi [19] suggest, it is often when the reigns of selection pressure are *lifted* that evolution innovates most prolifically. Novelty search is simply an accelerated version of this passive force in natural evolution; unlike in nature it *explicitly* rewards drifting away in the phenotype/behavior space, thereby pushing the innovating process ahead. While this perspective bypasses a long-standing notion of adaptive innovation in open-ended evolution [3, 4, 16], it offers a complementary view that is recommended by its intuitive simplicity: Open-endedness can be defined simply as the continual production of novelty.

The benefit of this view is that it means that we can now endow *any* domain with this kind of open-endedness. No longer are we restricted to complex artificial life worlds in our pursuit of open-ended discovery. As long as novelty can be defined (which will not always be easy), it can be sought explicitly in every domain from simple XOR to the most complex artificial world, putting many practical problems in machine learning within its reach.

For example, it is difficult to evolve a checkers player from scratch against a fixed world-class opponent because early generation individuals are always completely defeated. Yet novelty search abandons the idea that winning is the goal; rather it can simply try to *lose in a different way*. As the approaches to losing are exhausted one by one, eventually it will cross the path to winning, avoiding all deception and providing an entirely new kind of practical search that is nevertheless open-ended.

In addition, in the context of artificial life, it is interesting to consider how novelty search relates to natural evolution. Novelty is preserved in nature as long as a novel individual meets minimal selection criteria. It is also encouraged through niching. Moreover, there is evidence of active novelty search in natural evolution as well: Intersexual selection sometimes biases mate choice towards novelty [12]. Thus it is not unreasonable to view natural evolution as a kind of novelty search in addition to an adaptive competition.

Finally, novelty search provides a new hope for an artificial arrow of complexity. For, as Dawkins has said [6], once all the simple ways to live have been exhausted, the only way to do anything different is to become more complex. In a passive way, this idea explains the arrow of complexity in nature. In novelty search, the principle should also hold true.

In fact, the result that the solutions to the medium maze discovered by NEAT with novelty search contain almost

three times fewer connections than those discovered by fitness-based NEAT suggests that novelty search climbs the ladder of complexity *more efficiently*. While this intriguing result merits further study, a possible explanation is that compact solutions are often missed by objective-based search because they are hidden behind deceptive landscapes. Novelty search is more likely to encounter the most compact solution on the way up the ladder of complexity because it is not susceptible to such deception.

The problem with the objective is that it fails to identify the stepping stones. The more ambitious and complex the problem, the more difficult it is to formalize an objective that rewards the stepping stones along the way. Yet it is exactly those stepping stones that ultimately must be identified and rewarded if search is to find its way up the ladder of complexity [19]. Novelty search is designed to build gradients that lead to stepping stones. By abandoning the objective, all the steps along the way come into greater focus. While the trade-off is a more expansive search, it is better to search far and wide and eventually reach a summit than to search narrowly and single-mindedly yet never come close.

The implications of this approach are far-reaching because it is relevant to all of machine learning. The idea that search is more effective without an objective challenges fundamental assumptions and common intuitions about why search works. It is also the first machine learning approach to take seriously the growing (yet controversial) consensus in biology and artificial life that adaptive selection does not explain the arrow of complexity in nature [14, 15]. Novelty search asks what is left if the pressure to achieve the objective is abandoned. Thus its potential reach is broad. Furthermore, the implication for artificial life is that the adaptive competition is not necessary to promote an open-ended dynamic, suggesting a new approach to modeling evolution in artificial worlds.

In summary, almost like a riddle, novelty search suggests a surprising new perspective on achievement: *To achieve your highest goals, you must be willing to abandon them.*

Conclusions

This paper introduced novelty search, a domain-independent method of open-ended search. Motivated both by the problem of deceptive gradients in objective-based search and the desire for a simple approach to open-ended evolution, novelty search ignores the objective and instead searches only for individuals with novel behaviors. Counterintuitively, experiments in a deceptive navigation task showed that novelty search can significantly outperform objective-based search. Novelty search thus makes it possible to effectively apply the power of open-ended search to real-world problems.

Acknowledgements

This research was supported in part by a gift from Dr. Charles Bailey. Special thanks to Li Yuan for her support in preliminary investigations.

References

[1] M. A. Bedau and N. H. Packard. Measurement of evolutionary activity, teleology, and life. In C. G. Langton, C. Taylor, J. D. Farmer, and S. Rasmussen, editors, *Proc. of Art. Life II*, pages 431–461, Redwood City, CA, 1991. Addison-Wesley.

[2] M. A. Bedau, E. Snyder, and N. H. Packard. A classification of longterm evolutionary dynamics. In C. Adami, R. Belew, H. Kitano, and C. Taylor, editors, *Proc. of Art. Life VI*, pages 228–237, Cambridge, MA, 1998. MIT Press.

[3] Mark Bedau. Four puzzles about life. *Artificial Life*, 4:125–140, 1998.

[4] A. Channon. Passing the alife test: Activity statistics classify evolution in geb as unbounded. In *Proceedings of the European Conference on Artificial Life(ECAL-2001)*. Springer, 2001.

[5] Paul Darwen and Yin Yao. Every niching method has its niche: Fitness sharing and implicit sharing compared. In Hans-Michael Voigt, Werner Ebeling, Ingo Rechenberg, and Hans-Paul Schwefel, editors, *Parallel Problem Solving from Nature – PPSN IV*, pages 398–407, Berlin, 1996. Springer.

[6] Richard Dawkins. Genetic and evolutionary computation conference (GECCO-2007) Keynote Debate, July 2007.

[7] E. D. De Jong. The incremental pareto-coevolution archive. In *Proc. of the Genetic and Evol. Comp. Conf. (GECCO-2004)*, Berlin, 2004. Springer Verlag.

[8] Jeffrey L. Elman. Incremental learning, or the importance of starting small. Technical Report 9101, CRL, La Jolla, CA, 1991.

[9] David E. Goldberg. Simple genetic algorithms and the minimal deceptive problem. In L. D. Davis, editor, *Genetic Algorithms and Simulated Annealing, Research Notes in Artificial Intelligence*. Morgan Kaufmann, 2007.

[10] Faustino Gomez and Risto Miikkulainen. Incremental evolution of complex general behavior. *Adaptive Behavior*, 5:317–342, 1997.

[11] Inman Harvey. *The Artificial Evolution of Adaptive Behavior*. PhD thesis, School of Cognitive and Computing Sciences, U. of Sussex, Sussex, 1993.

[12] Kimberly A. Hughes, Linh Du, F. Helen Rodd, and David N. Reznick. Familiarity leads to female mate preference for novel males in the guppy, poecilia reticulata. *Animal Behavior*, 58(4):907–916, 1999.

[13] Marcus Hutter and Shane Legg. Fitness uniform optimization. *IEEE Transactions on Evolutionary Computation*, 10:568–589, 2006.

[14] Michael Lynch. The evolution of genetic networks by non-adaptive processes. *Nature Reviews Genetics*, 8:803–813, 2007.

[15] Michael Lynch. The frailty of adaptive hypotheses for the origins of organismal complexity. In *Proc Natl Acad Sci USA*, volume 104, pages 8597–8604, 2007.

[16] C. C. Maley. Four steps toward open-ended evolution. In *Proc. of the Genetic and Evol. Comp. Conf. (GECCO-1999)*, pages 1336–1343, San Francisco, 1999. Kaufmann.

[17] Andrew P. Martin. Increasing genomic complexity by gene duplication and the origin of vertebrates. *The American Naturalist*, 154(2):111–128, 1999.

[18] Daniel W. McShea. Complexity and evolution: What everybody knows. *Biology and Philosophy*, 6(3):303–324, 1991.

[19] Thomas Miconi. Evolution and complexity: The double-edged sword. *Artificial Life: Special Issue on the Evolution of Complexity*, 2007.

[20] Thomas Miconi. *THE ROAD TO EVERYWHERE: Evolution, Complexity and Progress in Nature and in Computers*. PhD thesis, U. of Birmingham, 2007.

[21] Melanie Mitchell, Stephanie Forrest, and John H. Holland. The royal road for genetic algorithms: Fitness landscapes and ga performance. In F. J. Varela and P. Bourgine, editors, *Proceedings of the First European Conference on Artificial Life*, Cambridge, MA, 1992. MIT Press.

[22] Tom M. Mitchell. *Machine Learning*. McGraw-Hill, New York, 1997.

[23] C. L. Nehaniv and J. L. Rhodes. On the manner in which biological complexity may grow. In *Math. and Comp. Biology*, volume 26 of *Lectures on Mathematics in the Life Sciences*, pages 93–102. American Mathematical Society, 1999.

[24] T. Ray. Evolution, ecology and optimization of digital organisms. Technical Report Working paper 92-08-042, Santa Fe Institute, 1992.

[25] Russell Standish. Open-ended artificial evolution. *International Journal of Computational Intelligence and Applications*, 3(167), 2003.

[26] Kenneth O. Stanley, Bobby D. Bryant, and Risto Miikkulainen. Real-time neuroevolution in the NERO video game. *IEEE Trans. on Evol. Comp. Special Issue on Evolutionary Computation and Games*, 9(6):653–668, 2005.

[27] Kenneth O. Stanley and Risto Miikkulainen. Evolving neural networks through augmenting topologies. *Evolutionary Computation*, 10:99–127, 2002.

[28] Kenneth O. Stanley and Risto Miikkulainen. A taxonomy for artificial embryogeny. *Artificial Life*, 9(2):93–130, 2003.

[29] Kenneth O. Stanley and Risto Miikkulainen. Competitive coevolution through evolutionary complexification. *Journal of Art. Int. Research*, 21:63–100, 2004.

[30] Michiel Van de Panne and Alexis Lamouret. Guided optimization for balanced locomotion. In D. Terzopoulos and D. Thalmann, editors, *Sixth Eurographics Workshop on Animation and Simulation*, pages 165–177. Springer Verlag, 1995.

[31] James D. Watson, Nancy H. Hopkins, Jeffrey W. Roberts, Joan A. Steitz, and Alan M. Weiner. *Molecular Biology of the Gene Fourth Edition*. The Benjamin Cummings Publishing Company, Inc., Menlo Park, CA, 1987.

[32] Larry Yaeger. Computational genetics, physiology, metabolism, neural systems, learning, vision and behavior or polyworld: Life in a new context. In C. G. Langton, editor, *Art. Life III, Proc. Vol. XVII*, pages 263–298. Addison-Wesley, 1994.

Adaptive multi-robot bucket brigade foraging

Adam Lein Richard T. Vaughan

Autonomy Lab, Simon Fraser University
Burnaby, British Columbia, Canada
{alein,vaughan}@sfu.ca

Abstract

Bucket brigade foraging improves upon homogeneous foraging by reducing spatial interference between robots, which occurs when robots are forced to work in the same space, and must spend time avoiding one another instead of carrying out useful work. Bucket brigade foraging algorithms restrict the motion of each robot to at most some fixed distance from its starting location. We reproduce the performance of known bucket brigade foragers, and then present a new controller in which robots adapt the size of their foraging area in response to interference with other robots, improving overall performance. This approach also has the potential to cope with nonuniform resource distributions.

Introduction

The *foraging problem* is a task in which one or more agents must find and collect target objects and deliver them to a "home area". The simplest adaptation of this problem to multi-robot systems is for many robots to independently solve the foraging task, a solution known as *homogeneous foraging*.

Shell and Matarić (2006) explore foraging strategies in large-scale multi-robot systems. The key variable in their experiments is spatial interference, which refers to destructive interactions between robots forced to perform work in the same space. Multi-robot systems are advantageous only when what might be called the marginal performance – the benefit to performance of adding a single robot to the system – is positive. However, as those authors demonstrated, there comes a time when this is no longer so, when the loss of performance due to interference between robots outweighs the gain in work done by having more workers.

Goldberg and Matarić (2003) and Østergaard et al. (2001) describe *bucket brigading*, in which each robot is restricted to a finite search area, and instead rely on their coworkers to both deliver pucks into their search area, and remove pucks out of it. By restricting each robot to a finite area whose size is determined a priori, interference is ameliorated. Shell and Matarić (2006) empirically investigate the performance of large groups of robots as a function of varying search area sizes; homogeneous foraging corresponds to search areas of infinite radius. Østergaard et al. (2001) describe the expected performance of multi-robot, space-constrained systems as a curve to which the $R = 40m$ curve in Figure 1 roughly corresponds; the curve has a local maximum, after which the marginal performance is negative.

In this paper, we investigate an approach to bucket brigading that does away with *a priori* search radii by allowing robots to adapt their search radii in response to interference with other robots, improving overall perfomance.

Related work

Foraging is a common analogy for a wide variety of robot tasks, including exploration and mapping, search, and actual objects retrieval.

Beckers et al. (1994) noted the use of *stigmergy* in insect swarms. Stigmergy is a process by which insects (in general, agents) communicate implicitly and indirectly by modifying the environment – their coworkers' future behavior is affected by these changes. The authors built robots which utilized stigmergy to collect objects and gather them into a pile – essentially analogous to the foraging task discussed in this paper. They found that increasing the number of robots decreases the mean time required to complete the task, *but only up to a certain point* (three robots), after which the mean time increased, due to intereference between robots.

Holland and Melhuish (1999) examined stigmergy and self-organization in physical robots: using very simple behavioral rules, the robots were able to cluster and sort Frisbees, despite possessing no memory or capacity for spatial orientation. The authors argued that their results hinged on the robots' behavior taking advantage of real-world physics.

Wawerla and Vaughan (2007) applied the rate-maximizing foraging model to a single robot performing the task of foraging over a long period of time. The robot had a finite energy supply, and was required to travel to a charging station to recharge its batteries. While recharging, and while traveling between the work site and the charging site, the robot is not doing work. The authors presented a scalable, online, heuristic algorithm for the robot to recharge efficiently, maximizing the proportion of its time it spends

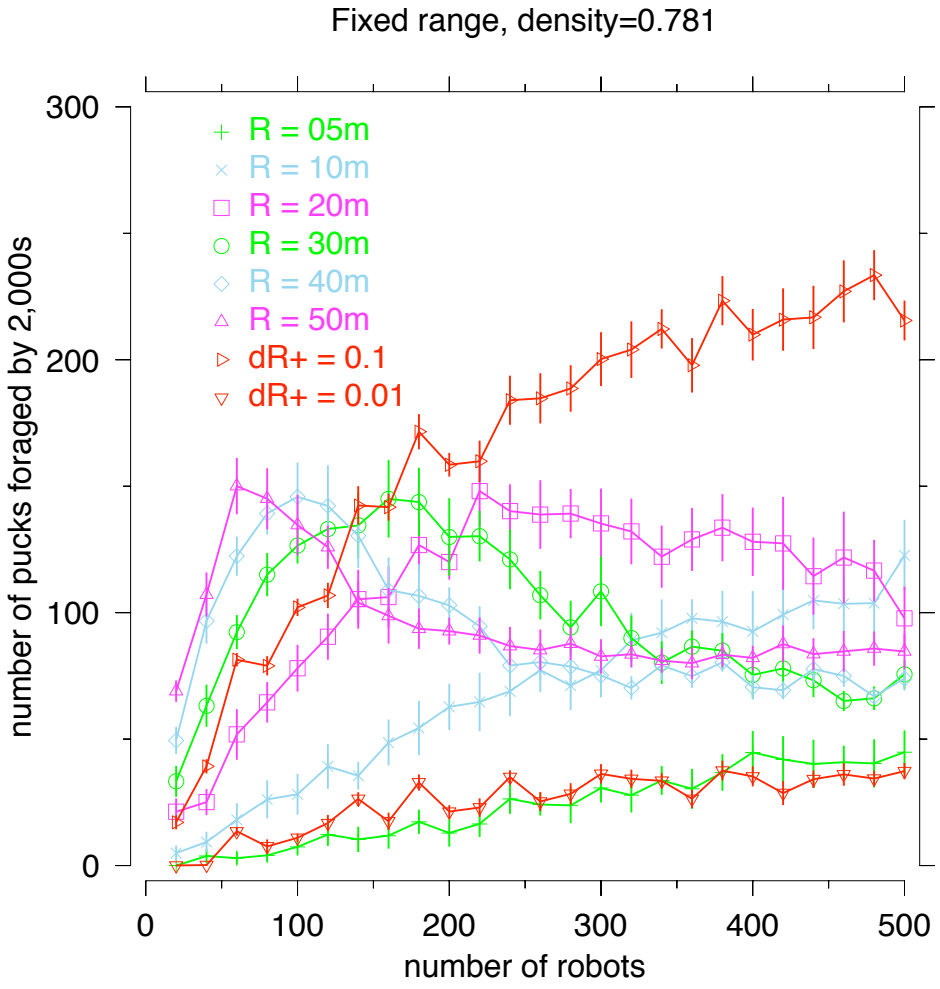


Figure 1: Performance with fixed and adaptive search radii at 0.781 pucks/m^2. R is the search radius of each robot in the trial. The curves labeled dR^+ show the results for adaptive range selection.

working.

In Zuluaga and Vaughan (2005), building on Brown et al. (2005), the problem of spatial interference in multi-robot systems was addressed through the use of aggressive display behaviors. Several robots were required to perform a transportation task (akin to our foraging task) in shared space. Robots selected an "aggression level" based on the amount of work they had invested up to that point. The discrepancy in aggression levels between interfering robots was used to break the symmetry that would otherwise have lead to deadlock. The authors showed their approach to be effective, both in simulation and in a real-world implementation.

Lerman and Galstyan (2002) formally modeled the effect of interference on the performance of a swarm of foraging robots. Their model formulated as a system of coupled first-order nonlinear differential equations. They found that group performance grows sublinearly with group size, so that individual performance actually decreases with increasing group size. Simulations verified the predictions of their model.

Rybski et al. (2004) performed experiments in which real robots perform a foraging task using a variety of simple communication methods. Robots communicated by flashing a light bulb under various circumstances. The authors showed that communication can reduce the variance in the robots' performance. In contrast, this research does not use explicit communication between robots; communication is implicit, however, in that robots must alter their behavior in the short term in response to the presence of other robots (collision avoidance), and in the long term by adapting a parameter of their behavior (discussed later).

Simulation

In order to reproduce the results of Shell and Matarić (2006), and compare them to the performance of the adaptive bucket brigaders, we developed a simulator similar to that in Shell and Matarić (2006). A description follows:

Robots are located in an arena, a 64 meter-square plane

scattered with pucks, in the northeast corner of which plane is a 3 m quarter circle "home area". Robots are equipped with grippers which they can use to retrieve these pucks, and when a puck is dropped off in the home area, it is said to have been foraged.

Robots can move forward at a rate of 0.1 m/sec, and can turn to either side at a rate up to once every five seconds. It takes four seconds to retrieve a puck, but a carried puck can be dropped instantaneously.

Robots can sense walls and other robots within 0.6 m of their centers without error (compared with 0.5 m in Shell and Matarić (2006)), via the use of 12 radially oriented sensors. These sensors report the range in meters to the nearest obstacle (wall or robot). Specifically, each robot is idealized as a line segment, and if the center or either endpoint of that line segment is within sensor range, the sensor most closely oriented towards that point reports the distance to that point.

Robots cannot sense pucks unless the puck in question is located directly within the grip of their grippers, and this sensing is binary and also without error. Robots can determine the direction towards the home zone at any time; Shell and Matarić (2006) explain this as the use of a "four-bit compass". Robots use the data from their proximity sensors to avoid walls and other robots.

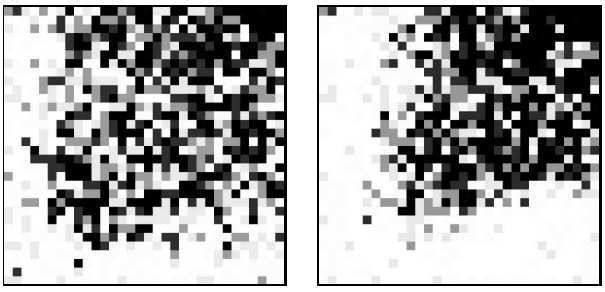
Shell and Matarić (2006) add a variety of noise to each parameter in their simulation. For simplicity, we have dispensed with this noise, except in the case of odometry noise (described in greater detail below). While this does detract from the physical realism of the experiments, the conclusions in this work are drawn by comparing two controllers in identical worlds; we do not attempt to compare our new controller with one tested under different conditions.

Parametrized bucket brigading

The purpose of the overall robot system is to retrieve pucks and deliver them into the home area – to forage the pucks. In the bucket brigade approach to this problem, individual robots do not attempt to carry a puck all the way to the home zone themselves, but rather merely to shift the distribution of pucks towards the home area.

Each robot will attempt to stay within a fixed distance from its initial location. This zone is known as the robot's "work area". Through odometry, the robot can determine how far it is from the center of this zone, and can tell the direction towards the center of the zone. However, this odometry is noisy; as a result, the center of each robot's work area drifts on a random walk at a rate of 0.01 m/sec.

The robot searches via a naïve algorithm within its work area. If it ever leaves the work area, it will drop off any puck it may be carrying, return towards its work area, and continue searching. If it ever discovers a puck, it will retrieve it and head towards the home area. The effect is that a "brigade" of similarly-behaving robots will slowly transport pucks from one robot (work area) to the next, gradually bringing the puck closer to the home area. Of course, ultimate delivery of the puck requires a connected sequence of overlapping work areas ending in the home area, but this may be achieved over time (even if never simultaneously) due to the drift of the work areas.



(a) (b)

Figure 2: Typical density distribution of pucks after (a) 40 minutes and (b) 80 minutes. One square is 2 m $\times$ 2 m; darker squares indicate a higher concentration of pucks. The home area is in the northeast corner of the world. In this simulation, pucks were not added to the world after the start of the experiment.

Shell and Matarić's robots all share the same work area radius, or "range". In the following sections, we will explore other approaches to assigning these ranges to robots. In any given experiment, every robot uses the same approach to range selection.

Radial parametrization

Given that the emergent behavior of the robots is to shift the distribution of the pucks towards the home area, the first natural modification to the controller would be to allow the range parameter of robots to vary with the distance to the home area. Effectively, we are discarding the assumption that the distribution of pucks is uniform, but supposing that they may be more densely distributed near the home area, and therefore a different range parameter would be optimal. To demonstrate this, we simulated robots whose range parameter varied linearly with distance from the home area.

Let us consider the following idealized, one-dimensional situation: robots and pucks sit uniformly distributed on a line of length L, with the home zone at one end. Every robot collects a puck at the same time, drives a fixed distance D towards the "home" end, deposits its puck, and then returns to its starting location, at which point the process repeats. Let $p_n(r)$, where r is the distance from a given point to the home end, be the density of pucks at that point after n of these cycles has taken place. $p_0(r)$ is some constant (the initial puck density). Then

$$p_{n+1}(r) = \begin{cases} (1-c)p_n(r) & \text{if } r > L - R, \\ (1-c)p_n(r) + cp_n\left(\frac{Lr}{L-D}\right) & \text{otherwise.} \end{cases} \quad (1)$$

In our simulation, an analogous process occurs in two dimensions. Puck densities at two time-steps are displayed in Figure 2 during a typical run of an experiment in which pucks are not added to the system as fast as they are removed from the system by foraging (pucks in the other experiments in this work are added just as fast as they are foraged, to maintain constant average puck density).

In all experiments, pucks are initially distributed at random. However, it can clearly be seen that as soon as the robots interact with the pucks, the distribution becomes less random, biased toward the home area — the system has a form of entropy that decreases as a result of the work of the robots. Since the optimal search radius for a robot depends on the density of pucks, once the density of pucks changes, the robot's original choice for search radius may no longer be optimal. Ideally, robots would know and select the optimal choice for search radius on an ongoing basis, but in these experiments, robots are not given enough information to achieve this ideal.

In the next section, we propose a method for approximating this ideal.

Adaptive range selection

The aforementioned ideal puts an extra burden on the robot — it must be constantly aware of the distance between the center of its zone and the center of the home area. Alternatively, it must be able to measure the local puck density. In place of these assumptions we may also allow robots to adaptively select their range parameter using purely local information. In *adaptive range selection*, a robot will continuously increase its range parameter at rate dR^+, except while it is avoiding a collision with another robot, to which situation the robot will react by shrinking its zone at rate dR^-, thus making it less likely to interfere with other robots in the future.

Consider the extreme example of a robot alone in the 64×64 meter-square arena. The robot's work area has some initial radius - say, 10m. The robot will remove pucks from his work area and carry them outside in the direction of the home area. In parametrized bucket-brigade foraging, once the robot has removed all pucks from his work area, there will be no work for him to do, but he will not know this, since he cannot sense pucks, or their absence, from afar – so he will continue searching. His restricted search radius doesn't improve efficiency since there are no robots to interfere with his navigation. Adding adaptive search radii into the picture, the robot's search radius grows at a rate of dR^+ and never shrinks (in practice, we limit the growth so that the radius of any search area is at most the diagonal of the arena).

Now consider the addition of a second robot into our example. As long as the two robots stay outside of each others' sensor ranges, their search areas will continue to grow as before; this scenario will be the same as the above-described scenario. However, each robot is sensitive to other robots that come within a certain range, less than their sensor range. If such an encounter occurs, each robot will take action to avoid colliding with the other. At the same time (i.e. as long as the collision-avoidance behavior persists), each robot's search area will decrease at the rate of dR^-. Eventually, it may happen that one robot's search area shrinks so that the robot is no longer inside it; at this point, instead of continuing to search for pucks, that robot will try to return to his search area, thus lessening the chance that he will encounter and interfere with the other.

As mentioned above, no robot's search area will grow without bound – there is a maximum useful radius (the diagonal of the arena). In addition, no robot's search area is allowed to decrease below the space needed for the robot to drive in a full circle (at fixed forward/turn speeds).

Results for adaptive range selection are included in Figures 1 and 3.

Experimental design

Initially, we followed Shell and Matarić (2006) in experimental design. The following parameters were varied: puck density ($0.781/m^2$ and $3.125/m^2$), search area radius (5, 10, 20, 30, 40, or 50 m), and number of robots (20, 40, 60, 80, ..., 500). The task was simulated for each combination of parameters for 2000 simulated seconds, and the number of pucks foraged after that time was recorded. Twenty such trials were run, each with a different initial distribution of pucks; to control for robot position, robots were initially placed on a square lattice. The reported results, in Figures 1 and 3, are the averages of those twenty trials. Error bars indicate the standard deviations of the twenty-trial experiments.

Next, we tested our adaptive range selection controller using the same experimental setup. For these experiments, search radii were allowed to increase by $dR^+ = 0.1$ m/s and decrease by $dR^- = 0.05$ m/s, biasing the robots towards the limit of homogeneous foraging in the absence of significant interference. Each robot began with a small range of 5 m. For each parameter set, twenty trials were run, and mean performances were plotted with standard deviations shown.

Results

Our first step was to reproduce the results in Shell and Matarić (2006). Inspection of the data in Figures 1 and 3 indicates that this was accomplished, in that increasing the radius of robots' search areas in the fixed-radius regime led to an increase in the marginal benefit of adding robots (i.e., to the slope of the curves in those graphs), but only up to a point: eventually, adding more robots decreases the performance of the system as overcoming interference begins to dominate the robots' behavior. These critical points are clearly visible as the significant local maxima in the $R = 30$ m, $R = 40$ m, and $R = 50$ m curves.

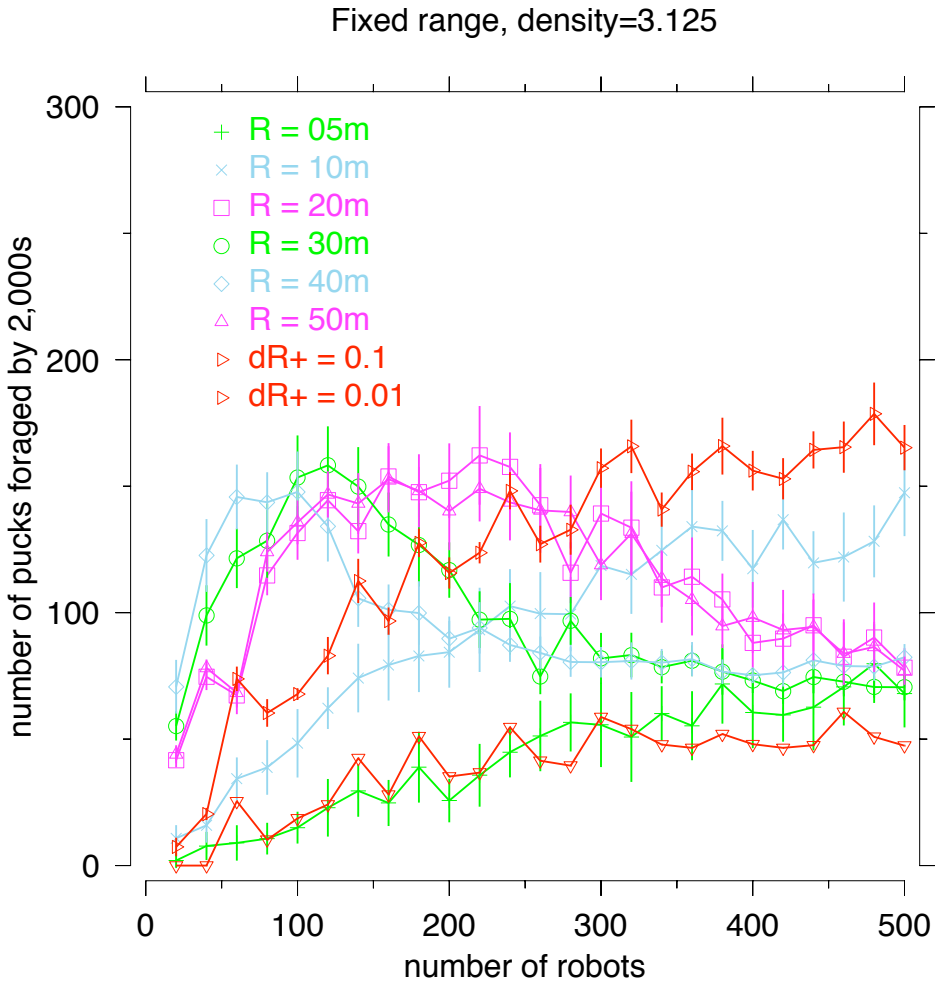


Figure 3: Performance with fixed and adaptive search radii at 3.125 pucks/m^2. R is the search radius of each robot in the trial. The curves labeled dR^+ show the results for adaptive range selection.

There is a noteworthy distinction in that robots in our experiments only foraged approximately half the pucks that those of Shell and Matarić (2006) did; this indicates more a quantitative difference in the efficacy of the robots' controller programs than a qualitative failure of our simulations to agree at the heart of the matter: that interference affects a growing population later when the individuals' foraging spaces are larger, which is indicated by the relative shapes of the curves.

Our adaptive range selection algorithm performed at least as well as the fixed ranged controllers in simulation, and scaled better. Fixed range algorithms suffered from one of two problems: robots with small search areas did not gather many pucks, and robots with large search areas interfered too much and the critical point at which the marginal benefit of increasing the number of robots was reached when the number of robots was still small. While the robots using adaptive range selection did not gather as many pucks when the number of robots was small as did robots with large, fixed search radii, increasing the number of robots always increased the performance of the group.

Also noteworthy is that the adaptive controllers performed more consistently, as indicated by tighter error bars on those curves than on the fixed-radius performance curves. Since adaptation to interference is a form of implicit communication, this is in agreement with the findings of Rybski et al. (2004).

Adaptive range selection was sensitive to variations in dR^+ and dR^-. If dR^+ was too small, adaptive selection underperformed the fixed range foragers. Figure 1 shows results when $dR^+ = 0.01$, a fifth of dR^-. In that case, the adaptive controller performs no better than the worst-performing fixed-range controller we tested. This is not altogether surprising, since the search radius in the worst fixed-range controller and the initial search radius in the $dR^+ = 0.01$ m adaptive controller were both 5 m.

Figure 3 shows qualitatively similar improvements; however, in this scenario (where puck density is 3.125 pucks per

square meter, the improvement is only slight, even for large group sizes).

Future work

In this paper, we have explored only some of the ways — and naïve ones at that — of improving on the bucket brigading algorithm. Future work may explore generalization of the problem to discard hidden assumptions, or, increasing the adaptability of the system. For instance, a more plausible biological analog might be ants foraging for food. In such a scenario, the ants would not initially be uniformly distributed through the field, nor would the food.

The performance of the adaptive-parameter bucket brigade forager needs to be more deeply tested. In none of our experiments did we note negative marginal performance; the limits of the algorithm in terms of scalability remain to be seen. The parameter space for the adaptive controller — the possible values of dR^+ and dR^- — needs to be explored in greater depth. We should also test the controller in more topologically complex spaces, such as corridors, as did Østergaard et al. (2001), and investigate why the adaptive controller does not outperform as significantly in denser environments.

Briefly mentioned above is the point that the adaptive approach causes a net increase in the number of a priori parameters: new parameters added are the rates at which zones increase and decrease in size. In this work, those parameters were set experimentally. Future work may provide motivation behind values of these parameters, investigate a relationship between optimal values for dR^+ and dR^-, or do away with these parameters entirely.

Future work may also formalize a closed-form relationship among optimal search radius, puck density, and robot density. Additionally, we have briefly touched on a notion of entropy in robots' work space — a quantitative function of the environment that decreases through *useful* interactions between agents and the environment — this notion should be more formally analyzed.

Conclusions

To summarize the contributions of this paper: we replicated the results of Shell and Matarić (2006), confirming their findings with an independent implementation. Further, we propose a simple modification of their foraging scheme in which each robot's foraging area is adapted in response to interference. The new method was shown to improve performance, particularly in large population sizes.

Acknowledgments

Special thanks to Jens Wawerla, for saving me from learning the hard way, and to Yaroslav Litus, for braving the cluster with me.

References

Beckers, R., Holland, O., and Deneubourg, J.-L. (1994). From local actions to global tasks: Stigmergy and collective robotics. In Brooks, R. and Maes, P., editors, *Artificial Life IV*, pages 181–189. Cambridge, MA: MIT Press.

Brown, S., Zuluaga, M., Zhang, Y., and Vaughan, R. (2005). Rational aggressive behaviour reduces interference in a mobile robot team. In *Proceedings of the International Conference on Advanced Robotics (ICAR)*, Seattle, Washington.

Goldberg, D. and Matarić, M. (2003). Maximizing Reward in a Non-Stationary Mobile Robot Environment. *Autonomous Agents and Multi-Agent Systems*, 6(3):287–316.

Holland, O. and Melhuish, C. (1999). Stigmergy, Self-Organization, and Sorting in Collective Robotics. *Artificial Life*, 5(2):173–202.

Lerman, K. and Galstyan, A. (2002). Mathematical Model of Foraging in a Group of Robots: Effect of Interference. *Autonomous Robots*, 13(2):127–141.

Østergaard, E., Sukhatme, G., and Matarić, M. (2001). Emergent bucket brigading: a simple mechanisms for improving performance in multi-robot constrained-space foraging tasks. In *Proceedings of the fifth international conference on Autonomous agents*, pages 29–30. ACM Press New York, NY, USA.

Rybski, P., Larson, A., Veeraraghavan, H., LaPoint, M., and Gini, M. (2004). Communication strategies in Multi-Robot Search and Retrieval: Experiences with MinDART. In *Proceedings of the Seventh International Symposium on Distributed Autonomous Robotic Systems (DARS-04)*, pages 301–310.

Shell, D. A. and Matarić, M. J. (2006). On foraging strategies for large-scale multi-robot systems. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2717–2723, Beijing, China.

Wawerla, J. and Vaughan, R. T. (2007). Near-optimal mobile robot recharging with the rate-maximizing forager. In *Proceedings of the European Conference on Artificial Life*.

Zuluaga, M. and Vaughan, R. (2005). Reducing spatial interference in robot teams by local-investment aggression. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Edmonton, Alberta.

Tracking the Evolution of Chemical Computing Networks

Thorsten Lenser, Naoki Matsumaru, Thomas Hinze, and Peter Dittrich

Bio Systems Analysis, Friedrich-Schiller-University Jena and Jena Centre for Bioinformatics
Institute of Computer Science, Ernst-Abbe-Platz 1-4, D-07743 Jena, Germany
{thlenser, naoki, hinze, dittrich}@minet.uni-jena.de

Abstract

How do chemical reaction networks that process information evolve? This is not only a fundamental question in the study of the origin of life, but also in diverse fields like molecular computing, synthetic biology, and systems biology. Here, we study the evolution of chemical flip-flops by means of chemical organisation theory. Additionally, we compare evolved circuits with manually constructed ones. We found that evolution selects for an organisational structure that is related to function. That is, the resulting computation can be explained as a transition between organisations. Furthermore, an evolutionary process can be tracked as a change of the organisational structure, which provides a fundamentally different view than looking at the structural changes of the reaction networks. In our experiments, 90% of evolutionary improvement coincide with a change in the organisational structure. We conclude that our approach provides a novel and useful perspective to study evolution of chemical information processing systems.

Introduction

In every living entity, cellular functions emerge from the astonishing interplay of connected reaction processes. Three essential types of biochemical networks can be distinguished: metabolic, cell signalling, and gene regulatory networks (Alberts et al., 2003). While metabolism consists of coupled enzymatically catalysed reactions supplying energy, cell signalling, and gene regulation perform information processing of external and internal signals (Cooper et al., 2001). Taking this information processing as a metaphor, biochemical reaction networks (or rather mathematical models of these) can be designed to perform specific computational tasks.

While a bottom-up approach has been pursued (Guido et al., 2006), top-down approaches, specifically evolutionary algorithms, have gained growing interests recently in order to design or program reaction systems. Efforts have been undertaken to evolve simple computational units (Deckard and Sauro, 2004), small biological networks (Koza et al., 2001; François and Hakim, 2004; Soyer et al., 2006), genetic regulatory networks (Dwight Kuo et al., 2006) or components thereof (Paladugu et al., 2006). Most of this work, however, has been focused on the final product, that is, the networks evolved to reproduce a certain specified behaviour. Here, we rather concentrate on the process of evolution. For that purpose, new methods are required that can deal with constructive systems (Fontana and Buss, 1994), that is, systems where new components (molecular species) or new interactions between existing components appear so that the network topology changes dynamically. Matsumaru et al. (2006) used chemical organisation theory (Dittrich and Speroni di Fenizio, 2007) in order to study the evolutionary dynamics of (artificial) chemical systems. In this paper, we analyse the trajectory of evolving chemical reaction networks that compute. That is, in particular, networks that function as flip-flops.

In previous work, the authors have developed a software designed to evolve biological networks (called the SBMLevolver) and measured the performance impact of certain design decisions for this algorithm (Lenser et al., 2007). That software package is adopted to evolve network models for this study. In the following section, the theory of chemical organisation is briefly reviewed. Then, the experimental setting to evolve a reaction network capable of flip-flop operation is presented. As results, three aspects of the evolutionary process are given in the Results section. In addition to the traditional aspect of the dynamical behaviour of the evolution, we analyse the dynamical change in terms of the chemical organisation within the reaction networks. We also show a reaction network evolved for the flip-flop function.

Reaction Networks and Chemical Organisations

Here, we utilized the notation of a chemical organisation developed by Dittrich and Speroni di Fenizio (2007) to analyse reaction networks. Following Fontana and Buss (1994), an organisation is defined as a set of molecular species that is closed and self-maintaining. The hierarchy of all organisations of a reaction network represents its organisational structure, which can be used to describe the dynamical (qualitative) behaviour of a reaction system as a movement between organisations (Speroni di Fenizio et al., 2001). Choosing a proper coding scheme, the organisational struc-

ture can be interpreted as a repertoire of behaviour patterns of the reaction system. For example, Dittrich and Speroni di Fenizio (2007) have shown that only species that form an organisation can makeup a stationary state.

The basic concepts needed here are now described more formally: A **reaction network** $\langle \mathcal{M}, \mathcal{R} \rangle$ consists of a set of (molecular) species $\mathcal{M}$ and a set of reaction rules $\mathcal{R}$. A **reaction rule** $\rho \in \mathcal{R}$ can be written according to the chemical notation:

$$l_{a_1,\rho}\, a_1 + l_{a_2,\rho}\, a_2 + \cdots \longrightarrow\ r_{a_1,\rho}\, a_1 + r_{a_2,\rho}\, a_2 + \ldots \quad (1)$$

The **stoichiometric coefficients** $l_{a,\rho}$ and $r_{a,\rho}$ describe the amount of molecular species $a \in \mathcal{M}$ in reaction $\rho \in \mathcal{R}$ on the lefthand and righthand side, respectively. Together, the stoichiometric coefficients define the **stoichiometric matrix**

$$\mathbf{S} = (s_{a,\rho}) = (r_{a,\rho} - l_{a,\rho}). \quad (2)$$

An entry $s_{a,\rho}$ of the stoichiometric matrix denotes the net amount of molecules of type a produced in reaction ρ. We also define mappings $\mathrm{LHS}(\rho) \equiv \{a \in \mathcal{M} : l_{a,\rho} > 0\}$ and $\mathrm{RHS}(\rho) \equiv \{a \in \mathcal{M} : r_{a,\rho} > 0\}$, returning the species with a positive coefficient on the lefthand and righthand side, respectively. Reaction ρ can take place in $A \subseteq \mathcal{M}$ only when $\mathrm{LHS}(\rho) \subseteq A$.

Given a reaction network $\langle \mathcal{M}, \mathcal{R} \rangle$ with $m = |\mathcal{M}|$ species and $r = |\mathcal{R}|$ reactions, the organisational structure is derived with respect to the following two criteria: closure and self-maintenance. A set of species $A \subseteq \mathcal{M}$ is **closed**, if for all reactions ρ with $\mathrm{LHS}(\rho) \subseteq A$, the products are also contained in A, that is, $\mathrm{RHS}(\rho) \subseteq A$. This closure property ensures that there exists no reaction in A producing new species not yet present in the organisation using only species of that organisation. The other property is a theoretical capability of an organisation to maintain all of its members. Since the maintenance possibly involves complex reaction pathways, the stoichiometry of the whole reaction network must be considered in general. A set of molecules $C \subseteq \mathcal{M}$ is **self-maintaining**, if there exists a flux vector $\mathbf{v} \in \mathbb{R}^r$ such that the following three conditions apply: (1) for all reactions ρ that can take place in C (i.e., $\mathrm{LHS}(\rho) \subseteq C$) the flux $v_\rho > 0$; (2) for all remaining reactions ρ (i.e., $\mathrm{LHS}(\rho) \nsubseteq C$), the flux $v_\rho = 0$; and (3) for all molecules $a \in C$, the production rate $(\mathbf{Sv})_a \geq 0$. v_ρ denotes the element of $\mathbf{v}$ describing the flux (i.e. rate) of reaction ρ. $(\mathbf{Sv})_a$ is the production rate of molecule a given flux vector $\mathbf{v}$.

We visualize the set of all organisations with a Hasse diagram, in which organisations are arranged vertically according to their size in terms of the number of their members (cf. Figure 6). Two organisations are connected by a line if the upper organisation contains all species of the lower organisation and there is no other organisation between them. The Hasse diagram represents the hierarchical **organisational structure** of the reaction network under study.

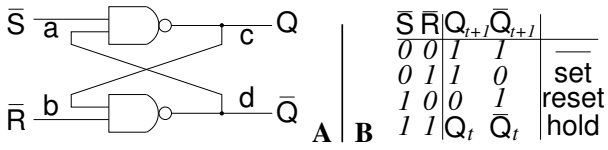


Figure 1: Circuit diagram and operation mode of flip-flop.

Method

We employ an evolutionary algorithm that instantiates a natural selection process on chemical reaction networks (Fernando and Rowe, 2007). The algorithm can mutate the reaction rules $\mathcal{R}$ of a reaction network with a fixed predefined set of molecular species $\mathcal{M} = \{a^0, a^1, b^0, b^1, c^0, c^1, d^0, d^1\}$. As mutational operators, the algorithm can add or delete a reaction, or replace a reaction with a different one, keeping as many of the previous participants as possible. To keep things simple, we employ a (1+1)-EA. That is, one parent generates one offspring, while the better of the two survives.

To enable neutral mutations and thus search space exploration, the offspring is kept if both have the same fitness. No parameter fitting is done, so that a change in parameters can only be realised through a replacement of a reaction with the same reaction, which has a different (randomly chosen) reaction constant. Only mass-action kinetics of first and second order are used in the evolution.

When speaking of a flip-flop logic gate in this work, we specifically mean an RS (Reset and Set) flip-flop, with a behaviour according to the truth table in Figure 1. To represent the four binary variables `a`, `b`, `c` and `d` making up this flip-flop in a chemical format, we employ two opposing species x^0 and x^1 for each binary variable `x`, where the presence of x^0 denotes the value `x = 0`, and x^1 denotes `x = 1` (cf. Matsumaru et al., 2007). To help maintain a valid state inside the system, we fix four destructive reactions $x^0 + x^1 \rightarrow \emptyset$ for all four species pairs $x^i = a^i, b^i, c^i, d^i$. These reactions cannot be changed or deleted by the evolutionary algorithm.

The ideal flip-flop that is the target of the artificial evolution works in the following way: The set operation $(\bar{S}, \bar{R}) = (0, 1)$ changes the state Q to 1, while the reset $(\bar{S}, \bar{Q}) = (1, 0)$ changes Q to 0. To hold the previous state, both inputs are set to 1. The forbidden input $(\bar{S}, \bar{Q}) = (0, 0)$ is not considered in the fitness function. In chemical form, the input $(\bar{S}, \bar{R}) = (0, 1)$ is represented by defining an inflow for a^0 and b^1, that is, $\{\emptyset \rightarrow a^0, \emptyset \rightarrow b^1\} \subseteq \mathcal{R}$; and the other two cases are treated similarly. The initial concentrations of c^i and d^i are set according to the previous state Q_t. Taking this together, we get six different test cases, coming from three different operations with two initial conditions each.

For each case, we specify either the presence or the absence of each species as desired, measured in steady state after simulating the reaction system for 1000 seconds. Numerical integration is done using the SBML ODE Solver Li-

brary (Machne et al., 2006). The classification as present or absent is decided by a concentration threshold of 10^{-9} (arbitrary units). For example, in the reset case, the following steady state concentrations are considered as correct: $a^1 = 1, a^0 = 0, b^1 = 0, b^0 = 1, c^1 = 0, c^0 = 1, d^1 = 1, d^0 = 0$. The fitness value is then calculated by counting the number of wrong presence / absence measurements, with 0 being the best possible fitness value. Once a fitness of 0 is reached, the evolution stops.

Results

To analyse the evolution of reaction networks acting as flip-flops, we performed 30 independent runs in order to evaluate properties of a "typical" run. Additionally, we also looked at one run in more detail.

Since the distinction between the three different input settings is realised by enabling or disabling inflow reactions for a^1, a^0, b^1 and b^0, we need to compute three lattices of organisations for each analysed candidate solution, one for each input setting.

Statistical analysis of many runs

The average fitness development (Figure 2) shows a stronger gain in fitness at the beginning, while the convergence towards zero is slower later on. Eventually, all runs reached a fitness of zero, i.e. the networks behaved as specified in the fitness function. Since a run stops exactly when the fitness of the current individual is 0, the number of generations usually differ between runs. In order to be able to average over these runs, we had to resample the data on fitness and number of organisations, such that a common number of measurements for each run is achieved. To this end, we constructed a timescale of "normalised evolutionary progress", defined by its endpoints 0.0 at the beginning of the evolution and 1.0 at the end when the final solution is found. The MATLAB function `resample`, which applies an anti-aliasing lowpass FIR filter during the resampling process, was used to create new data points at 1001 equally space points between 0.0 and 1.0.

Looking at the number of organisations for the three different input cases, we can see from Figure 3 that starting from around four to five organisations on average, the numbers diverge between the set/reset operations and the hold operation. While the number of organisations for the set/reset organisation converges between two and three, the hold operation yields around seven organisations on average.

By comparing the organisational structures between successive candidate solutions, we calculated that 90% of all fitness improvements are accompanied by a change in the organisational structure for at least one input case. In contrast, only 18% of organisational changes also come with a fitness improvement. When looking at the lineage of networks that led to the final solution, disregarding unsuccessful candidates, we find that 35% of all mutations changed the organisational structure for at least one input.

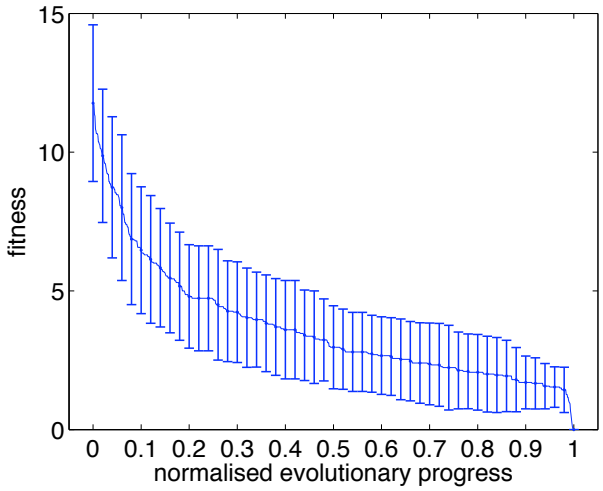


Figure 2: Average fitness value from beginning to end of evolutionary runs, from 30 independent repetitions. The x-axis denotes the normalised evolutionary progress from the random initial solution ($x = 0$) to finding a solution with fitness 0 ($x = 1$). For this, the different runs were resampled to 1000 samples, as described in the text. Errorbars indicate standard deviation.

Detailed analysis of one run

For an in-depth analysis, we pick the first evolutionary run that we performed for this problem. We will describe how the fitness improvements correlate with changes in the organisational structure, and give details on one specific mutational event and its consequences for the organisational structure of the network.

Comparing the average fitness development shown in Figure 2 with the single run analysed here (Figure 4 upper part), we can conclude that the fitness of the individual run progressed in a fairly standard way. This is especially true given that the behaviour of the 30 runs is quite diverse, as indicated by the large standard deviations. Also the length of this run (162 generations) is in the usual region, with an average run taking 221 generations with a standard deviation of 119. Also the number of organisations (Figure 4 lower part) is in agreement with the average number (Figure 3), even if the number of organisations for the set/reset operations are at the outer limits of the typical range (five and one, respectively).

Looking at the fitness increases in the course of the evolution (Figure 4 upper part) and the organisational structures of all networks that appear during the run (not shown), it can be observed that all but one of the eight fitness jumps are accompanied by changes in the organisational structure for at least one of the three input cases. However, taking the number of organisations at any point in the evolution into

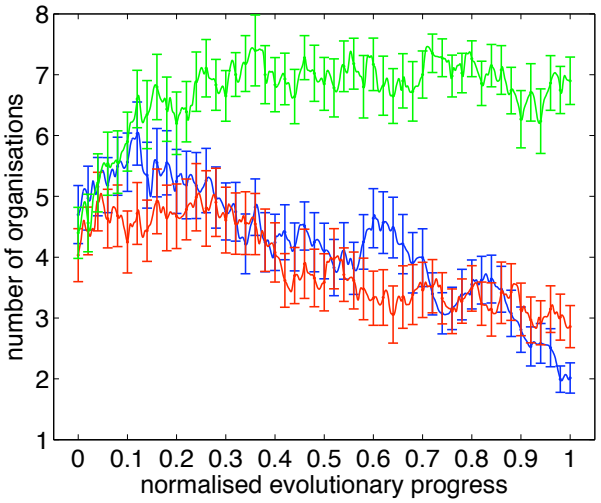


Figure 3: Average number of organisations from 30 independent runs of the evolution. Colors denote the (a_0, b_1) input (blue), the (a_1, b_0) input (red), and the (a_1, b_1) input (green). Errorbars indicate the standard error. Unit of x-axis as in figure 2.

account, one can also see that not every change in organisational structure leads to a fitness change, in fact, most do not.

We now relate the fitness change in one successful mutation to the change in organisational structure incurred by that mutation. As an example, we pick the fitness jump from generation 112 to 113, which improved the fitness from seven to four wrong presence/absence values. Looking at the reaction networks before and after the mutation (Figure 5), we see that the mutation added one reaction, which converts b^0 into d^1.

This additional reaction does not change the organisational structure for input cases (a^0, b^1) and (a^1, b^1) (set and hold, not shown), but reduces the lattice of organisations for input case (a^1, b^0) (reset) from five organisations to two (Figure 6). Looking at the behaviour of both networks for all input cases and initial configurations (i.e. all six test-cases), one can observe that the change occurs only in input case (a^1, b^0) with an initial configuration in which c^1 and d^0 are present (data not shown). For this case, the steady-state before the mutation has a^1, b^0, and c^1 present, and d^0 is still present after 1000 seconds even though its concentration is still decreasing at that time. This yields four wrong presence/absence values, since c^0 and d^1 should be present and c^1 and d^0 should be absent, but the total opposite is the case. After the mutation, d^1 is present and c^1 and d^0 are absent, but also c^0 is absent, so there is one wrong value left.

On the organisational level, the mutation removes three organisations (Figure 6), among which is also the organisation (a^1, b^0, c^1) responsible for the wrong behaviour of the

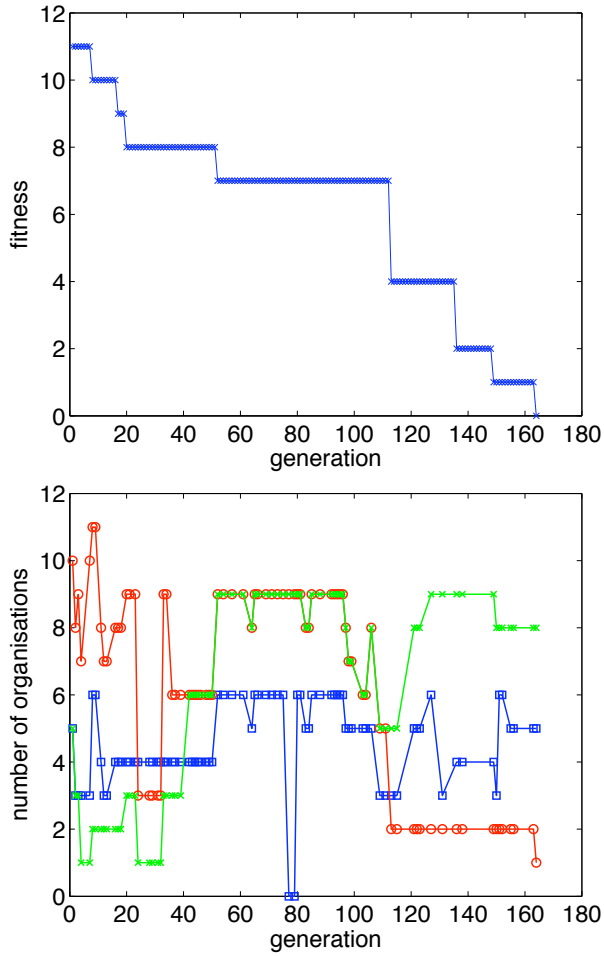


Figure 4: One exemplary run. Given are fitness (upper plot) and number of organisations for all three input cases (lower plot). In the lower plot, the three input cases are shown in blue ((a^0, b^1) input), red ((a^1, b^0) input), and green ((a^1, b^1)). Each mark (square, cross or circle) denotes a new network structure in the evolutionary trajectory.

network. After the mutation, the dynamics take the steady-state into organisation (a^1, b^0, d^1), resulting in a better behaviour. However, it is interesting to note that both organisational structures also contain organisation (a^1, b^0, c^0, d^1), which is the "correct" one that is also used in the final solution. Even though this organisation is present, the dynamics of both reaction systems are such that the steady-state does not lie inside it. We had to wait for another 49 generations for this to happen.

An evolved chemical flip-flop

An outcome of the evolutionary process described above is analysed. The reaction network considered here has a fitness value of 0, i.e. solves the given task. The network structure is shown in Figure 7.

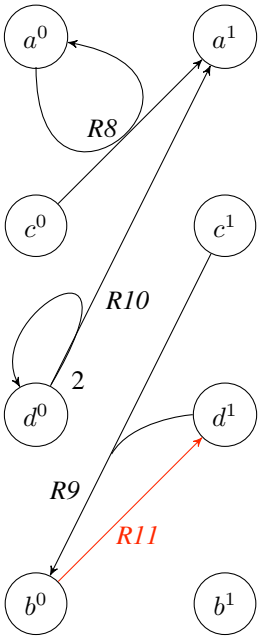


Figure 5: The reaction network of the candidate solution analysed in the text, after the mutation adding reaction *R11*. The added reaction is shown in red.

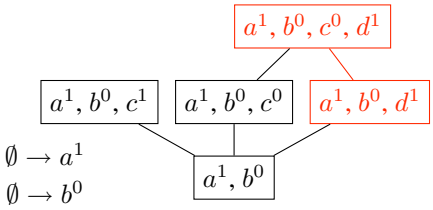


Figure 6: Organisational structure of the networks from Figure 5 for input (a^1, b^0), before (whole structure) and after addition of reaction *R11* (only red part).

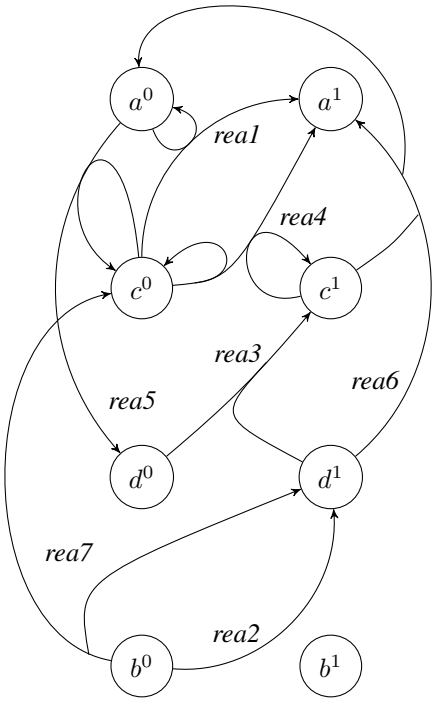


Figure 7: Chemical reaction network implementing flip-flop circuits, designed through an evolutionary process. Cooperative decay reactions ($a^1+a^0 \rightarrow \emptyset$, $b^1+b^0 \rightarrow \emptyset$, $c^1+c^0 \rightarrow \emptyset$, $d^1 + d^0 \rightarrow \emptyset$) are omitted.

There are seven reactions, labeled as *rea1* to *rea7* in the figure, in addition to four reactions of cooperative decay (not shown in the figure), $a^1+a^0 \rightarrow \emptyset$, $b^1+b^0 \rightarrow \emptyset$, $c^1+c^0 \rightarrow \emptyset$, $d^1 + d^0 \rightarrow \emptyset$. This base reaction network is extended to include inflow reactions, representing the inputs to the flip-flop circuit, depending on the operations. Organisational structures of the reaction system for each operational mode are shown in Figure 8.

Analysing the organisational structure of the reaction network, it becomes evident that the reaction system based on this reaction network is surely usable for the flip-flop computation. Including the two inflows $\emptyset \rightarrow a^1$ and $\emptyset \rightarrow b^0$ in the reaction network, as shown in Figure 8 A, only one set of species $\{a^1, b^0, c^0, d^1\}$ satisfies the conditions to be the organisation. It implies that only this species combination can be found in the dynamical reaction system in equilibrium states. Therefore, the reset operation can be realized in the evolved reaction system. The network with the inflows of $\emptyset \rightarrow a^0$ and $\emptyset \rightarrow b^1$ contains five organisations as shown in Figure 8 B, and one of those $\{a^0, b^1, c^1, d^0\}$ corresponds to the set operation.

Changing inflow reactions to $\emptyset \rightarrow a^1$ and $\emptyset \rightarrow b^1$ achieves the hold operation. In terms of the organisations, as shown in Figure 8 C, the two organisations *orgHR*=

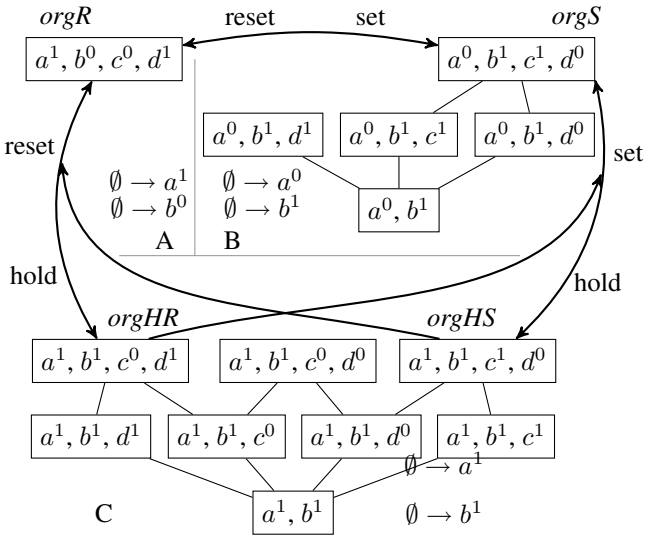


Figure 8: Organisational structure in the reaction network shown in Figure 7.

$\{a^1, b^1, c^0, d^1\}$ and *orgHS*= $\{a^1, b^1, c^1, d^0\}$ in the reaction network with those inflows reflect the bistability of the flip-flop circuit. Depending on the state at the previous time step, the hold operation results in a different state, namely the previous one. When the reaction system has been in the state after the set operation, (*i.e.*, *orgS*), the hold operation brings the system to the state of *orgHS*, keeping the output species unchanged as c^1 and d^0. Holding the information that the system has been reset can be achieved by moving the system state from *orgR* to *orgHR*.

The last operation of setting both inputs to be zero ($\mathtt{a} = \mathtt{b} = 0$) is forbidden for the flip-flop circuit. If adding two inflows of $\emptyset \to a^0$ and $\emptyset \to b^0$ to the base reaction network, one set of species becomes the organisation: $\{a^1, a^0, b^0, c^1, c^0, d^1, d^0\}$. Only b^1 is not involved to form the organisational structure.

If no inflow reaction is present, there are 42 organisations in the base reaction network. The smallest organisation is the empty set $\emptyset$. The sets containing four species forms the largest organisations, and there are four organisations of that size. The organisations with the size of four in Figure 8 are also found to be the organisation without inflows, except the organisation labeled as *orgR*. In fact, all organisations in Figure 8 except *orgR* are also organisations without inflows.

Dynamical Behaviour

To validate the organisational analysis of the reaction network, a dynamical reaction system is constructed and simulated with *Copasi* (Hoops et al., 2006), a biochemical reaction system simulator. Agreeing to the fitness calculation of the evolutionary design process, mass action kinetics is assumed for every reaction, if applicable. The ordinary differential equations (ODEs) for the input species read:

$$\begin{aligned}
[\dot{a^1}] &= k_1[a^0][c^0] + k_4[c^1][c^0] + k_6[c^1][d^1] \\
&\quad -d_a[a^1][a^0] + I_{a^1}(1 - [a^1]) \qquad (3)\\
[\dot{a^0}] &= -k_5[a^0][c^0] + k_6[c^1][d^1] \\
&\quad -d_a[a^1][a^0] + I_{a^0}(1 - [a^0]) \qquad (4)\\
[\dot{b^1}] &= -d_b[b^1][b^0] + I_{b^1}(1 - [b^1]) \qquad (5)\\
[\dot{b^0}] &= -k_2[b^0] - k_7[b^0] \\
&\quad -d_b[b^1][b^0] + I_{b^0}(1 - [b^0]) \qquad (6)
\end{aligned}$$

where a kinetic parameter for a reaction rea_id is denoted as k_{rea_id}. Kinetic parameters for the cooperative decay reactions are represented by d, and the subscript specifies the pair. For example, the decay rate of the cooperative decay reaction $a^0 + a^1 \to \emptyset$ is denoted as d_a.

Inflow reactions representing the operation of reset, set, and hold are controlled by the four parameters: I_{a^1}, I_{a^0}, I_{b^1}, and I_{b^0}. These parameters are binary variables, accepting only 0 or 1. For example, when the chemical flip-flop is set, I_{a^0} and I_{b^1} are set to one and the other pair of parameters $I_{a^1} = I_{b^0} = 0$ is set to zero. Inflows are assumed to be constant fluxes. Furthermore, the inflows are linked to normal decay reactions such as $a^1 \to \emptyset$ in order to avoid endless increase of the input species concentration. The resulting term of the ODE is $I_{a^1}(1 - [a^1])$, for example.

The ODEs for the output species read:

$$\begin{aligned}
[\dot{c^1}] &= k_3[d^1][d^0] - k_6[d^1][c^1] \\
&\quad -d_c[c^1][c^0] - I_{b^0}[c^1] \qquad (7)\\
[\dot{c^0}] &= -k_1[a^0][c^0] + k_7[b^0] \\
&\quad -d_c[c^1][c^0] - I_{b^0}[c^0] \qquad (8)\\
[\dot{d^1}] &= k_2[b^0] - k_3[d^1][d^0] - k_6[d^1][c^1] + k_7[b^0] \\
&\quad -d_d[d^1][d^0] - I_{b^0}[d^1] \qquad (9)\\
[\dot{d^0}] &= -k_3[d^1][d^0] + k_5[a^0][c^0] \\
&\quad -d_d[d^1][d^0] - I_{b^0}[d^0] \qquad (10)
\end{aligned}$$

Kinetic parameter values are also provided as the outcome of the evolutionary design, but we manually adjusted the values so that the operations can be continuously repeated. When the fitness of the reaction system was calculated during the evolution process, three of the operations were evaluated separately and the reaction system was reinitialized for each case. This re-initialization step between operations is prevented so that the end state of the previous operation becomes the initial state of the next operation. For that purpose, the outflows of the input species are added as described above in order to restrict the increase of the concentration. For the output species, the outflows are also added as shown above, activated only when the inflow of b^0 is present. This modification is also to restrict the increase of the concentrations of the output species, specially, when the system is reset.

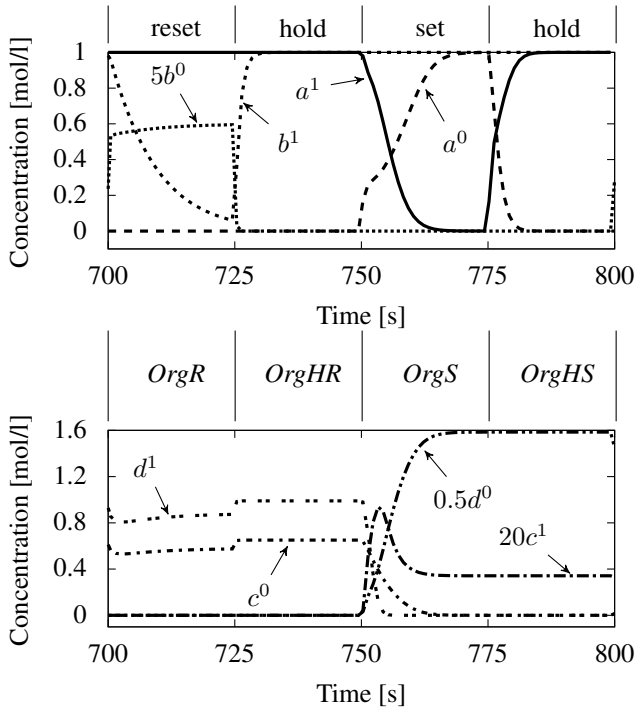


Figure 9: Dynamical simulation of chemical flip-flop designed by evolution. Parameters are set as follows: $d_a = d_b = 0.1$, $k_4 = 2.33941$, $k_6 = 2.83745$, $k_1 = 4.44231$, $k_5 = 3.62963$, $k_7 = 4.82838$, $k_3 = 1.0$, $k_2 = 0.1$, $d_c = 0.001$, $d_d = 1.0$. Additionally, for each operation of reset, set, and hold, inflow reactions are activated. For the set operation, the parameters are set such that $I_{a^0} = I_{b^1} = 0$ and $I_{a^1} = I_{b^0} = 1$ to activate inflows of a^1 and b^0 species and to deactivate the others. The reset operation is initiated by setting $I_{a^0} = I_{b^1} = 1$ and $I_{a^1} = I_{b^0} = 0$. The hold operation is achieved with the parameter settings of $I_{a^1} = I_{b^1} = 1$ and $I_{a^0} = I_{b^0} = 0$.

The last modification is the kinetic parameter of the reaction *rea1*, k_1, from 4.44231 to 0.5. The rational of this adjustment is: under the input condition "set", the system is observed to converge to the organisation of $\{a^0, b^1, d^1\}$, instead of *orgS*. This behaviour results from the fast extinction of species c^0 so that the generation of d^0 by *rea5* is insufficient. Slowing down the reaction speed of *rea1*, species c^0 stays in the system longer and produces d^0 enough to neutralize d^1.

Conclusion

We found that most fitness improvements come together with change in organisational structure (90%), showing that organisation analysis indeed yields insight into the evolutionary process. On the other hand, most organisational changes are fitness-neutral (82%), indicating that a lot of the information given in the lattice of organisations does not directly relate to the measured function of the networks. We have also seen mutations where the replacement of a reaction with the same type of reaction led to a fitness increase caused purely by the changing of a kinetic parameter, as well as changes of network structure not reflected in the organisations (but improving fitness). All this implies that while organisational analysis can give us many indications regarding the function of a reaction network, sometimes it does not tell the whole story of the network's dynamics.

We have also seen that the number of organisations for the set and reset states is substantially smaller than the number for the hold state, in analogy to the hand-constructed flip-flop by Matsumaru et al. (2007). In comparison to their solution, the evolved networks show a larger number of organisations for each input case. To realize the flip-flop behaviour in the reaction system, the minimum number of organisations in the reaction network is one for the set and reset operation and three for the hold operation. The hand-designed flip-flop implementation shown by Matsumaru et al. (2007) has two organisations for set and reset, respectively, and three for hold. In comparison, the evolved networks have more organisations, on average between two and three each for set and reset, and seven for hold. This implies that even though the function of the flip-flop networks is reflected in their organisational structure, this structure contains more information than only the operational modes specified in the fitness function.

As an interesting extension to this work, one could use organisational analysis to direct the evolution of reaction networks. By first designing the perfect organisational structure and then evolving networks with this structure, it would be possible to study whether these networks have the desired functionality. A key step in this direction is certainly the design of an appropriate fitness function based on a network's lattice of organisations.

In an additional investigation on top of the results shown here, one should look at the effect of different mutational operators on network structure, fitness and organisational structure. This will lead to helpful insights on how the mutations affect the lattice of organisations, and also on how specific organisational changes are related to changes in the fitness function.

In our opinion, the most important lesson to be learned from this work is that the evolutionary process investigated here produces reaction networks with an organisational structure that reflects their flip-flop functionality. Even though our choice of representation format of the binary information in chemical form may favour this, we believe that this phenomenon is mainly caused by the structure of the fitness function, i.e. by the task that is required of the networks. It will be very interesting in future to investigate this with other representation formats.

Acknowledgements

We acknowledge financial supports by the European Union, NEST-project ESIGNET no. 12789 and by the German Research Foundation (DFG) Grant DI 852/4.

References

Alberts, B., Johnson, A., and Lewis, J. (2003). *Essential Cell Biology*. Garland Publishing.

Cooper, B., Schonbrunner, N., and Krauss, G. (2001). *Biochemistry of signal transduction and regulation*. Wiley-VCH.

Deckard, A. and Sauro, H. (2004). Preliminary studies on the in silico evolution of biochemical networks. *ChemBioChem*, 5:1423–1431.

Dittrich, P. and Speroni di Fenizio, P. (2007). Chemical organisation theory. *Bull Math Biol*, 69(4):1199–1231.

Dwight Kuo, P., Banzhaf, W., and Leier, A. (2006). Network topology and the evolution of dynamics in an artificial genetic regulatory network model created by whole genome duplication and divergence. *BioSystems*, 85:177–200.

Fernando, C. and Rowe, J. (2007). Natural selection in chemical evolution. *Journal of Theoretical Biology*, 247(1):152–167.

Fontana, W. and Buss, L. W. (1994). 'The arrival of the fittest': Toward a theory of biological organization. *Bull Math Biol*, 56:1–64.

François, P. and Hakim, V. (2004). Design of genetic networks with specified functions by evolution in silico. *PNAS*, 101:580–585.

Guido, N. J., Wang, X., Adalsteinsson, D., McMillen, D., Hasty, J., Cantor, C. R., Elston, T. C., and Collins, J. J. (2006). A bottom-up approach to gene regulation. *Nature*, 439(7078):856–860.

Hoops, S., Sahle, S., Gauges, R., Lee, C., Pahle, J., Simus, N., Singhal, M., Xu, L., Mendes, P., and Kummer, U. (2006). COPASI - a COmplex PAthway SImulator. *Bioinformatics*, 22:3067–3074.

Koza, J., Mydlowec, W., Lanza, G., Yu, J., and Keane, M. (2001). Automatic synthesis of both the topology and sizing of metabolic pathways using genetic programming. In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO-2001)*, pages 57–65. Morgan Kaufmann.

Lenser, T., Hinze, T., Ibrahim, B., and Dittrich, P. (2007). Towards evolutionary network reconstruction tools for systems biology. In E. Marchiori, J.H. Moore, J. R. E., editor, *Proceedings of the Fifth European Conference on Evolutionary Computation, Machine Learning and Data Mining in Bioinformatics (EvoBIO)*, volume 4447 of *LNCS*.

Machne, R., Finney, A., Muller, S., Lu, J., Widder, S., and Flamm, C. (2006). The SBML ODE Solver Library: a native API for symbolic and fast numerical analysis of reaction networks. *Bioinformatics*, 22(11):1406–7.

Matsumaru, N., Centler, F., Speroni di Fenizio, P., and Dittrich, P. (2007). Chemical organization theory as a theoretical base for chemical computing. *International Journal of Unconventional Computing*, 3(4):285–309.

Matsumaru, N., Speroni di Fenizio, P., Centler, F., and Dittrich, P. (2006). On the evolution of chemical organizations. In Artmann, S. and Dittrich, P., editors, *Explorations in the complexity of possible life: abstracting and synthesizing the principles of living systems, Proceedings of the 7th German Workshop of Artificial Life*, pages 135–146. Aka, Berlin.

Paladugu, S., Chickarmane, V., Deckard, A., Frumkin, J., McCormack, M., and Sauro, H. (2006). In silico evolution of functional modules in biochemical networks. *IEE Proceedings-Systems Biology*, 153(4):223–235.

Soyer, O., Pfeiffer, T., and Bonhoeffer, S. (2006). Simulating the evolution of signal transduction pathways. *Journal of Theoretical Biology*, 241:223–232.

Speroni di Fenizio, P., Dittrich, P., and Banzhaf, W. (2001). Spontanous formation of cells in a universal artificial chemistry on a planar graph. In Kelemen, J. and Sosik, P., editors, *Advances in Artificial Life. Proc. 6th European Conference on Artificial Life (ECAL 2001), Prague, Czech Republic, September 10 - 14, 2001, LNCS 2159*, pages 206–215. Springer, Berlin.

Simulated Evolution of Mass Conserving Reaction Networks

Anthony M.L. Liekens, Huub M.M. ten Eikelder, Marvin N. Steijaert, Peter A.J. Hilbers

Faculty of Biomedical Technology, Technische Universiteit Eindhoven, the Netherlands
anthony@liekens.net

Abstract

With the rise of systems biology, the systematic analysis and construction of behavioral mechanisms in both natural and artificial biochemical networks has become a vital part of understanding and predicting the inner workings of intracellular signaling networks. As a modeling platform, artificial chemistries are commonly adopted to study and construct artificial reaction network motifs that exhibit complex computational behaviors. Here, we present a genetic algorithm to evolve networks that can compute elementary mathematical functions by transforming initial *input* molecules into the steady state concentrations of *output molecules*. More specifically, the proposed algorithm implicitly guarantees mass conservation through an atom based description of the molecules and reaction networks. We discuss the adopted approach for the artificial evolution of these chemical networks, evolve networks to compute the square root function. Finally, we provide an extensive deterministic and stochastic analysis of a core square root network motif present in these resulting networks, confirming that the motif is indeed capable of computing the square root function.

Introduction

In biological organisms, networks of chemical reactions control the processing of information in a cell. A general approach to study the behavior of these networks is to analyze modules that are frequently observed in natural systems. Numerous network motifs that perform computational tasks have been discovered in biochemical reaction networks. Reaction networks are able to compute Boolean operations and implement simple binary computers (Arkin and Ross, 1994; Sauro and Kholodenko, 2004; Steijaert et al., 2007). Cell signaling networks are known to exhibit parallelism, the integration and amplification of signals, bistable behavior and hysteresis through feedback and memory (Bray, 1990; Bhalla and Iyengar, 1999; Bray, 1995; Tyson et al., 2003; Steijaert et al., 2008). Many engineering metaphors have been put forward as analogies to signaling networks, such as neural networks and analog electronic circuits (Bray, 1995; ten Eikelder et al., 2007). As an example, elementary operations such as addition, multiplication, integration and amplification can be found as modules of the MAPK pathway (Bhalla, 2003).

A more recent approach to study computations in biochemical networks is to construct artificial networks or abstractions of real life systems to study the available space of computations that can be performed in cellular reaction networks. *In vitro* molecular computations have been performed with gene expression networks (Benenson et al., 2004). Through the adoption of artificial chemistries (Dittrich et al., 2001; Dittrich, 2001) to implement chemical networks, it has been shown that algebraic functions can be constructed using a bottom-up approach based on motifs that implement elementary mathematical operations (Buisman et al., 2008). Related research employs *in silico* evolutionary algorithms for the discovery of conceptual networks that perform basic computations (Deckard and Sauro, 2004; Paladugu et al., 2006; Lenser et al., 2007).

In the current study, we have developed a comparable genetic algorithm that allows for the evolutionary design of reaction networks with a desired function. For given input molecules and their initial amounts, the desired reaction network needs to process these input molecules and generate an output pool of products whose concentrations correspond to a desired function of the amounts of inputs. In contrast with related work, our approach guarantees networks that respect the law of conservation of mass explicitly. Molecular species in our reaction networks are considered to be strings consisting of imaginary atoms and by satisfying the condition that the total set atoms in reactants and products in a reaction must be equal for all reactions in the network, we can guarantee the conservation of mass in our reaction network. By enforcing this condition upon the construction of reaction networks and within the genetic operators in our genetic algorithm, it is guaranteed that the evolved networks do not violate the law of mass conservation. Other approaches either test for mass conservation at each fitness evaluation, e.g., Lenser et al. or ignore the law of mass conservation completely, e.g., Paladugu et al.

First, we give an overview of the implementation of the reaction networks within our artificial chemistries framework and how they are evaluated with respect to a desired input-output function. Together with a set of genetic oper-

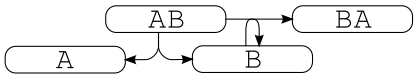


Figure 1: A small reaction network with two reactions.

ators that act on the networks, we can implement a genetic algorithm to artificially evolve such networks. Next, we review some networks that have been evolved with the software implementation, where the target function is the square root, i.e., for an amount X input molecules of a specific type, the reaction network needs to generate an output molecule with desired amount $\sqrt{X}$ at steady state. We review some evolved networks and discuss small network motifs that act as *kernels* in these square root networks. We analytically study one elementary network motif that computes a square root-like function and furthermore examine its behavior with a stochastic approach to model instantiations of the system with small molecular counts.

Methods

Molecules and reaction networks

In the artificial chemistries deployed by our genetic algorithm, we let an alphabet Σ denote the available *atoms* and strings s with variable length over Σ are possible *molecular species*. The number and the order of atoms in the string defines the uniqueness of a species. We define the *molecular mass* of a species or string s as a vector $\mathbf{m}$ where m_l is the number of characters $l \in \Sigma$ in s.

In order to abstract from natural biochemical networks, we say that an individual – which represents a reaction network – comprises of a number of reactions that take specific reactants from a pool of molecules, execute a specific transformation on these reactants at a predefined rate and return their products to the molecular pool. Each reaction has at most two reactants and two products. This limitation is inspired by nature, where enzymatic reactions with more than two substrates are rare. In order to guarantee mass conservation within the chemical networks, it is required that the total mass of reactants in a reaction equals the mass of the products. All of the reactions in this study are assumed to follow the law of mass-action kinetics according to a rate $k > 0$, i.e., a reaction occurs with a propensity that is proportional to k and the concentrations of available reactant molecules.

As an example, in a setup with alphabet or atom set $\Sigma = \{A, B\}$, the reaction network of an individual as depicted in Figure 1 contains the valid reactions $r_1 = AB \rightarrow A + B$ and $r_2 = AB + B \rightarrow BA + B$, with their respective reaction rates k_1 and k_2. Clearly the total mass, i.e., the number of atoms A and B, is conserved in these reaction.

Network evaluation and fitness

An individual or network is evaluated by providing the network with an input pool of reactants and observing the amounts of participating molecules over time. We let an ordinary differential equation (ODE) model of the reaction network compute the transient behavior of the network and if the system reaches a steady state, we compare the resulting pool of products with a desired output of the network, as a function of the input. As the output concentration of a molecule in the pool is closer to the desired output, for a set of inputs, the fitness of the individual is higher.

In a first step to compute the fitness of an individual, an ODE representation of its reaction network is constructed. Since we have assumed mass-action kinetics, this step is fairly straightforward and results in the following ODE system for our example network with reactions r_1 and r_2 as in Figure 1:

$$\begin{array}{rcl} & d[AB]/dt &= -k_1[AB] - k_2[AB][B] \\ d[A]/dt = & d[B]/dt &= k_1[AB] \\ & d[BA]/dt &= k_2[AB][B] \end{array}$$

where $[s]$ denotes the dimensionless amount or concentration of molecular species s.

In our genetic algorithm, reaction networks are stored as System Biology Markup Language (SBML) objects (Hucka et al., 2003). A network in this format is passed on to the SBML ODE Solver Library (Machne et al., 2006) which constructs an ODE model of the network and solves it with numerical integration. The ODE solver reports the steady state behavior of the network back to the fitness function, when a steady state is detected. If the ODE solver cannot find a steady state within a reasonable amount of time – a network may show, for example, stable oscillatory behavior, keeping it from reaching a steady state – the individual is eliminated from the population.

In order to compute the fitness of an individual in relation to an input-output function that needs to be approximated by the evolutionary algorithm, we iteratively run the ODE model for a set of input and output pairs. At time 0 of the ODE model, we set the initial amount of molecules for specifically designated input molecules. All other molecules in the system are initialized with concentration 0. With these initial values, the steady state of the ODE system is computed with the numerical ODE solver. For each molecule in the system, we compute the squared difference between the desired output and the steady state concentration of that molecule. The molecule with the smallest mean squared error for varying inputs is considered to be the output molecule and the fitness of the individual is inversely proportional to its mean squared error. Consequently, as the steady state concentration of a molecule is closer to the desired output, the fitness of the individual is higher. If an individual is detected not to reach its steady state for at least one input setting, its fitness is set to 0. It should be noted that we select specific molecules to act as input molecules of our system, but we do not select a specific molecule to act as the output molecule of the reaction network. As the evolutionary algo-

rithm is free to let all molecules act as output molecules, it is expected that the genetic algorithm is better able to find approximations of the desired function.

Genetic algorithm

The population of the genetic algorithm is seeded with randomly generated reaction networks, with a fixed number of reactions in each individual. In a random reaction, a random set of atoms is distributed over the two reactants and products, such that the total mass of reactants and products is equal. For all of the experiments in this paper, it sufficed to initialize the reactions with 2 to 7 random atoms. Mutations allow for larger molecules in the reaction networks if these would be required by the evolutionary process, as discussed later. Finally, a reaction rate is assigned to each reaction, uniformly chosen between 0 and 10.

To generate a new individual in the next generation's population, we select two parent individuals from the current population, proportionally to their fitness as defined above. We apply a uniform crossover operator and iterate over the resulting set of reactions with a mutation operator to generate the offspring individual which is evaluated and appended to the new population of the next generation.

With uniform crossover, we iterate over the ordered lists of reactions in both parents, where both reactions have an equal probability of ending up in the offspring's reaction network. Consequently, as the population is initialized with reaction networks with a fixed number of reactions, this number of reactions is maintained throughout the evolution of the population. This allows the user to enforce a specific number of reactions in the evolved reaction network and to prevent network bloat.

For the mutation operator, we iterate over the reactions in the offspring reaction network and mutate each reaction with a mutation parameter μ (usually, $\mu = 0.1$). The mutation operator for a reaction consists of two steps, one changing the reactants and products, where the second step mutates the reaction rate of the reactions. Firstly, in order to change the constituent products and reactants of a reaction, a random atom from alphabet Σ is inserted at a random position of a random reactant and a random product with probability $\mu/2$. Similarly, a randomly chosen atom is removed from both a reactant and a product in the reaction, also with probability $\mu/2$. Complete reactions are replaced with new, randomly generated reactions with a probability μ. Through these first mutation operators, the topology of the reaction network changes. Secondly, we multiply the reaction rate with a random number from a Gaussian distribution with mean 1 and standard deviation μ. This latter mutation operator does not change the topology of the reaction network, but it is involved in the parameter optimization of the network.

Typical runs of the genetic algorithm have been seeded with 100 individuals with just a few reactions in each individual. Our primary goal here is to find small networks – with up to 10 reactions – for elementary mathematical operations. However, with a limited reaction network size, the evolutionary algorithm may not be able to find exact implementations of the desired function – not all input-output functions can exactly be represented as a reaction network with a finite number of reactions – such that an approximation of the behavior evolves within the available space of network behaviors.

Parameter optimization

In addition to the above genetic algorithm that evolves both a network topology and reaction rate parameters, we have also adopted a limited version of the genetic algorithm for the optimization of reaction rates in a fixed network topology. In this genetic algorithm, the initial set of individuals is populated with networks of the same topology, but with random reaction rates (uniformly distributed between 0 and 10). The mutation operator in this algorithm is only allowed to mutate the reaction rate parameters of the constituent reactions, according to a normal distribution as described above. With uniform crossover, the reaction rates of the corresponding reactions are exchanged.

Networks that have been found by the main genetic algorithm are further optimized by this genetic algorithm, to obtain a network that behaves optimally for a given topology. Additionally, we have adopted this parameter optimizer to optimize user-defined networks and their corresponding desired behavior. Typical runs of the parameter optimizer assumed populations of 100 individuals for 100 generations.

Results

We have adopted the genetic algorithm to evolve networks that compute elementary mathematical operations. Some of the networks to compute these operations are straightforward. For example, a network that computes the difference $[A] - [B]$ of input molecules A and B, with $[A] > [B]$ can be as simple as the single reaction $A + B \rightarrow AB$ with rate $k > 0$. Each molecule A binds with a B molecule, such that $[A]$ can act as the output in the chemical equilibrium of the system, which is equal to the initial amount of A minus the initial amount of B. A network that is not as straightforward to construct a network that computes the square root of an amount of input molecule.

Square root networks

We have used our genetic algorithm to construct networks that compute the square root of the initial amount of input molecule ABC, with alphabet $\Sigma = \{A, B, C\}$. The ODE model of a candidate network is evaluated by setting the initial amount of molecule ABC to $X = 1, 4, 9, 16, \ldots, 100$ in consecutive runs. The molecule whose amount at steady state is nearest to the desired output $\sqrt{X}$ is designated as the

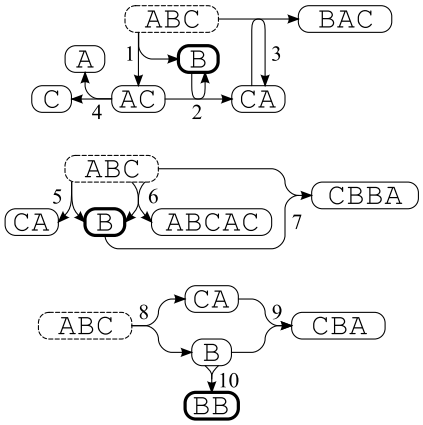


Figure 2: Three evolved networks that compute the square root function. The input molecule is denoted by the dashed outline, where the output is in bold.

output molecule of the network and its mean squared error is reported back to the fitness function.

Three networks that have been evolved with our genetic algorithm are shown in Figure 2. The first network, with mean squared error 0.119 for the 10 desired outputs, whose input is molecule ABC outputs a molecule B and consists of the following 4 reactions

$$\begin{aligned} ABC &\xrightarrow{k_1} B + AC && \text{with } k_1 = 1.159 \\ AC + B &\xrightarrow{k_2} CA + B && \text{with } k_2 = 3.716 \\ ABC + CA &\xrightarrow{k_3} BAC + CA && \text{with } k_3 = 5.590 \\ AC &\xrightarrow{k_4} A + C && \text{with } k_4 = 7.881. \end{aligned}$$

A second network that outputs an amount of molecules B that approximates the square root of molecules ABC (mean squared error equals 0.226) is given by the reaction network

$$\begin{aligned} ABC &\xrightarrow{k_5} B + CA && \text{with } k_5 = 7.359 \\ ABC + ABC &\xrightarrow{k_6} B + ABCAC && \text{with } k_6 = 1.502 \\ ABC + B &\xrightarrow{k_7} CBBA && \text{with } k_7 = 1.468. \end{aligned}$$

The third evolved reaction network with input ABC and output BB is given by

$$\begin{aligned} ABC &\xrightarrow{k_8} B + CA && \text{with } k_8 = 5.56937 \\ CA + B &\xrightarrow{k_9} CBA && \text{with } k_9 = 6.62825 \\ B + B &\xrightarrow{k_{10}} BB && \text{with } k_{10} = 8.34484 \end{aligned}$$

where the mean squared error between the counts of output molecule BB and the desired output is 0.335.

It should be pointed out that the evolved networks have been cleaned up manually to only show the reactions that are essential for the networks' square root behavior. Duplicate reactions have been merged and reactions that further process waste particles have been removed without affecting the output generating behavior of the network. E.g., in the second evolved network, waste molecule $CBBA$ is further processed into smaller waste molecules BB and CA, but this reaction does not interfere with the production of output molecule B from input ABC.

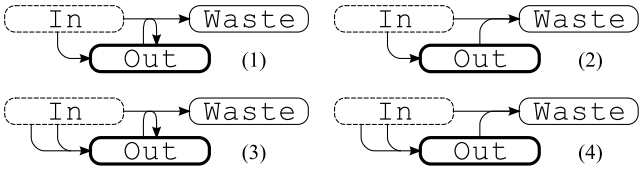


Figure 3: Four network motifs act as the kernels of evolved square root networks.

The relatively small errors show that all three networks provide good approximations of the desired square root behavior, within the range of inputs. Because the fitness is only evaluated within this input range, the networks do not guarantee a generally good approximation for other inputs.

Square root kernels In these and other networks that have been evolved to compute the square root of an input, a common behavioral subnetwork can be observed. In this common motif, a first set of reactions generates output molecules from the input molecules. The output molecule then takes part in reactions that remove remaining inputs from the pool. This behavior is at the core of the first and second evolved square root networks, as in Figure 2.

Figure 3 shows 4 elementary network motifs that are common to most of our evolved square root networks. All of these elementary kernels can be implemented such that they abide to the law of mass conservation. We have constructed these networks and optimized their parameters to study whether the behavior of these network motifs can be related to the square root. Table 1 gives the reaction rates for these networks such that the error for inputs $\{1, 4, 9, ..., 100\}$ is approximately minimal. Networks (3) and (4) behave worst, showing a steady state amount of output molecules that is linearly related to the input. Network (1) provides a good approximation of the square root network, where the performance of network (2) is mediocre. It should be pointed out that network (1) can be further optimized and has a mean squared error of 0.049 when $k_{out}/k_{waste} = 0.579$.

Analysis of a square root kernel

Since network motif (1) in Figure 3 provides the best approximation of the square root, we study the system analytically, in order to understand how the elementary network is capable of computing a good approximation of the square root function. We let x, y and w denote the amounts of input, output and waste molecules, and k_1 and k_2 the reaction rates of the output and waste producing reactions, respec-

kernel	k_{out}	k_{waste}	MSE
(l)	0.931	1.454	0.136
(2)	4.111	0.659	0.608
(3)	0.663	19.157	2.142
(4)	3.039	3.544	2.142

Table 1: Optimized parameters k_{out} and k_{waste} of the corresponding output and waste producing reactions for the square root kernels and their respective mean squared errors

tively. The network can be modeled by the following system of differential equations

$$\begin{aligned} \frac{dx(t)}{dt} &= -k_1 x(t) - k_2 x(t) y(t), \\ \frac{dy(t)}{dt} &= k_1 x(t), \\ \frac{dw(t)}{dt} &= k_2 x(t) y(t). \end{aligned}$$

For the computation of the square root of a number X, the initial concentrations for this system are $x(0) = X$, $y(0) = 0$ and $w(0) = 0$. The value of $y(t)$ for large t then hopefully approaches $\sqrt{X}$. Note that the differential equations are nonlinear, which makes it difficult to obtain analytical results.

Limiting values We first compute the behavior of the system for $t \to \infty$. Trivially, the limiting value of the input concentration $x(t)$ is given by $\lim_{t\to\infty} x(t) = 0$. Define the limiting values of output and waste by

$$\hat{y} = \lim_{t\to\infty} y(t), \qquad \hat{w} = \lim_{t\to\infty} w(t) \, .$$

We try to compute the value of $\hat{y}$. For all $t \geq 0$ the sum of the concentrations $x(t) + y(t) + w(t)$ is constant. In view of the initial condition this means that

$$x(t) + y(t) + w(t) = X \, . \tag{1}$$

Since $y(t)$ is supposed to approach the square root of X, it is obvious to consider $y^2(t)$. A simple computation gives

$$\frac{dy^2(t)}{dt} = 2k_1 y(t) x(t) = \frac{2k_1}{k_2} \frac{dw(t)}{dt} \, .$$

Using the initial conditions this implies that

$$y^2(t) - \frac{2k_1}{k_2} w(t) = 0 \, . \tag{2}$$

Since (1) and (2) hold for all values of t, we conclude that

$$\hat{y} + \hat{w} = X, \qquad \hat{y}^2 - \frac{2k_1}{k_2}\hat{w} = 0 \, .$$

Elimination of $\hat{w}$ from these equations leads to the quadratic equation

$$\hat{y}^2 - \frac{2k_1}{k_2}(X - \hat{y}) = 0 \, .$$

The non-negative solution of this equation is given by

$$\hat{y} = -\frac{k_1}{k_2} + \sqrt{\tfrac{2k_1}{k_2} X + \tfrac{k_1^2}{k_2^2}} \, .$$

By selecting $k_2 = 2k_1$ we obtain

$$\hat{y} = -\frac{1}{2} + \sqrt{X + \tfrac{1}{4}} \, .$$

Hence for X not too small, the chemical reaction network computes indeed an approximation of $\sqrt{X}$ if $k_2 = 2k_1$. Although all results can also be obtained for the general case, we shall assume in the sequel that $k_2 = 2k_1$. Note that the GA does not find this relation, as it attempts to compensate for the extra $-1/2$ in the steady state relation of our network.

Analytical solution In fact, in this case it is even possible to compute the analytical solution of the system. Using the relations (1) and (2) we can eliminate $x(t)$ and $w(t)$ from the system of differential equations. The resulting equation for $y(t)$ is then

$$\frac{dy(t)}{dt} = -k_1 y^2(t) - k_1 y(t) + k_1 X$$

Since a single first order differential equation can be solved by integration, even if it is nonlinear, we can integrate this equation. This results in

$$y(t) = a \tanh(k_1 a t + C) - \frac{1}{2} \, .$$

where $a = \sqrt{X + \frac{1}{4}}$ and $C = \operatorname{arctanh}(\frac{1}{2a})$. In Figure 4 we give the transient output concentration for the case $X = 400$ and $k_1 = 1/2, 1$ and 2. All three solutions approach $\sqrt{X}$ (more precisely, they approach $-\frac{1}{2} + \sqrt{X + \frac{1}{4}}$), but the speed of convergence increases with increasing k_1.

Stochastic model of a square root kernel

Modelling a chemical reaction system using differential equations that describe the time evolution of concentrations is limited to situations where smoothly varying concentrations exist. If the number of molecules is limited, this assumption does not hold anymore. In that case a discrete stochastic model can be used.

Probability distribution We now describe a simple Markov-like stochastic approach to the square root network. Suppose initially there are X input molecules, and no output and waste molecules. In each reaction of the system one input molecule transforms to either an output molecule or a

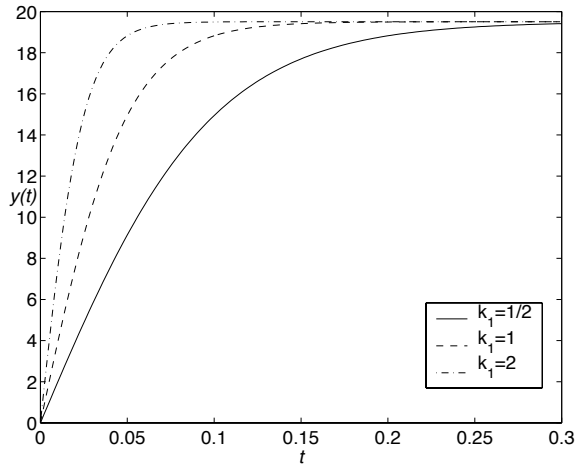


Figure 4: exact solution for $X = 400$ and $k_1 = 1/2, 1$ and 2

waste molecule. Since the total number of molecules is always equal to X, the state of the system can be described by a tuple (q, r) where q is the number of input molecules and r is the number of output molecules. The corresponding number of molecules of the waste compound is then trivially given by $X - q - r$. In general, in a state (q, r) there are two possible state transitions:

1. One of the input molecules transforms into an output molecule. This step happens with propensity $k_1 q$.
2. The other possibility is that an input molecule transforms into a waste molecule. This transition requires an output molecule as catalyst and has a propensity $k_2 q r$.

Since the probabilities of these two possible reactions are nothing but normalised propensities, the actual probabilities of the first and second possible reaction are $\frac{k_1}{k_1+k_2 r} = \frac{1}{1+2r}$ and $\frac{k_2 r}{k_1+k_2 r} = \frac{2r}{1+2r}$ respectively, with $k_2 = 2k_1$. Note that these transition probabilities do not depend on the number of input molecules q. Hence the transition probabilities depend only on the number of output molecules r. This means we can describe the system with the transition graph shown in Figure 5. A step to the right in this transition graph corresponds with an output producing reaction. In state r, i.e., with r output molecules, this reaction has probability $p_r = \frac{1}{1+2r}$. A step from state r to state r in the transition graph corresponds with an input to waste reaction. This reaction has probability $1 - p_r = \frac{2r}{1+2r}$.

Initially the system starts with X input molecules and no output and waste molecules. In terms of the transition graph the system starts in $r = 0$. After X steps all input molecules are used and the system can be in any of the states $r = 1, \ldots, X$ Note that, since $p_0 = 1$, the system cannot produce waste particles and is forced to move to state 1.

Let f_s be the distribution of the number of output molecules after s steps. So $f_s(r)$ is the probability that the

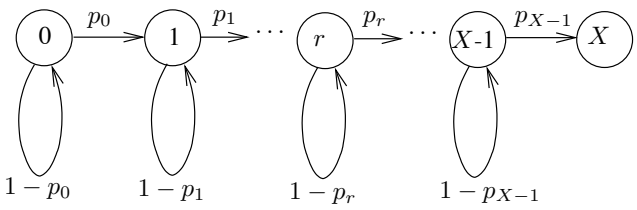


Figure 5: System described by number of output molecules r

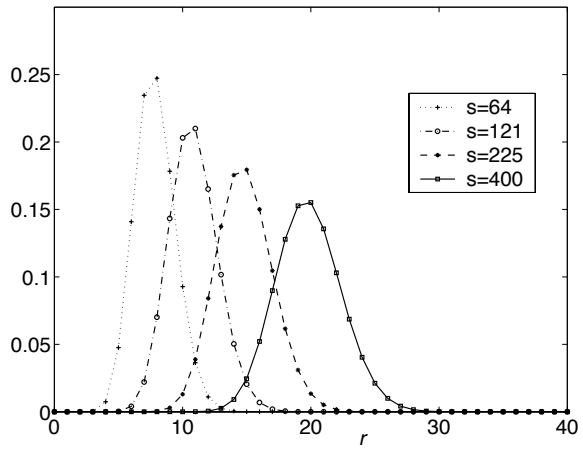


Figure 6: Distribution $f_s(r)$ for $s = 64, 121, 225$ and 400

system is in state r after s steps. Initially $f_0(0) = 1$ and $f_0(r) = 0$ for $r = 1, \ldots, X$. The successive distributions f_s can easily be computed recursively. It is easily seen from Figure 5 that

$$f_{s+1}(r) = \begin{cases} (1 - p_r)\, f_s(r) & \text{if} \quad r = 0 \\ (1 - p_r)\, f_s(r) + p_{r-1}\, f_s(r-1) & \text{if} \quad r \geq 1. \end{cases} \tag{3}$$

With this formula the distributions f_s can easily be computed. In Figure 6 the results are shown for $s = 64, 121, 225$ and 400. As can be seen from Figure 6 the various distributions f_s have their maximum value in $\sqrt{s}$. This means that it is most likely that the stochastic system, started with X input molecules, ends after $s = X$ steps with $\sqrt{X}$ output molecules. However, as Figure 6 shows, other final numbers of output molecules are very well possible.

Mean of the probability distribution The results of the previous subsection suggest that the probability distribution of the number of output molecules after s steps in centered around $\sqrt{s}$. We now try to give a mathematical basis for this observation. Let the mean of probability distribution f_s be given by

$$M_s = \sum_{r=0}^{X} r f_s(r) \; .$$

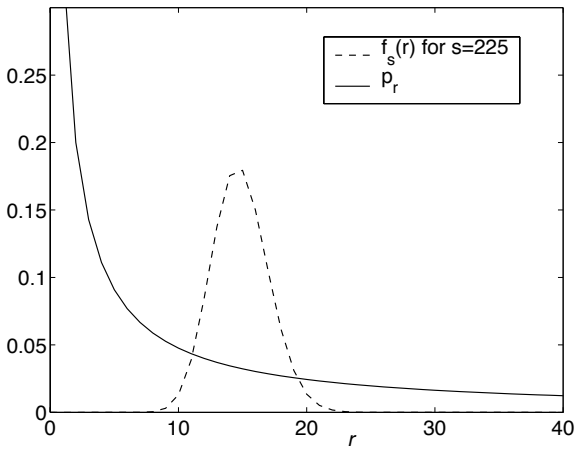


Figure 7: p_r and $f_s(r)$ as function of r for $s = 225$

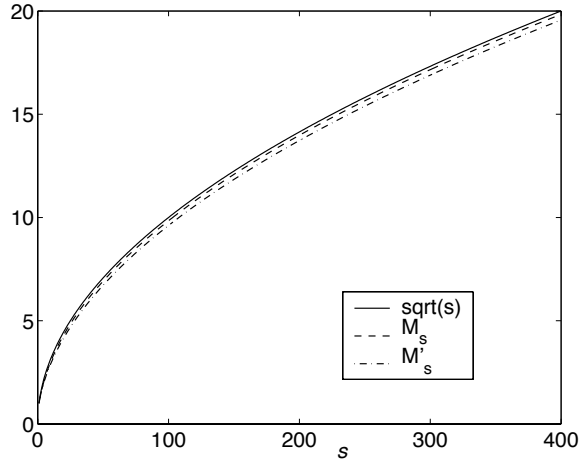


Figure 8: $\sqrt{s}$, M_s and the approximation M'_s

We try to compute M_s. From (3) we obtain for $s < X$.

$$\begin{aligned} M_{s+1} &= \sum_{r=0}^{X} r f_{s+1}(r) \\ &= \sum_{r=0}^{X} r(1-p_r)\, f_s(r) + \sum_{r=1}^{X} r p_{r-1}\, f_s(r-1) \\ &= \sum_{r=0}^{X} r(1-p_r)\, f_s(r) + \sum_{r=0}^{X} (r+1) p_r\, f_s(r) \\ &= \sum_{r=0}^{X} r\, f_s(r) + \sum_{r=0}^{X} p_r\, f_s(r) \\ &= M_s + \sum_{r=0}^{X} p_r\, f_s(r)\ . \end{aligned} \tag{4}$$

The behavior of p_r and $f_s(r)$ as function of r are shown in Figure 7 for the case $s = 225$. This figure shows that the largest contribution to the summation in (4) comes from the r values between 10 and 20. Hence we can approximate the summation in (4) by replacing p_r by a constant value p_{r_0} that gives a good approximation of p_r in the interesting region. In the situation of Figure 7 we could use $r_0 = 15$, thus approximating p_r by the constant p_{15}.

For the general case it is tempting to use $r_0 = \sqrt{s}$, since we conjecture that the maximum of the distribution $f_s(r)$ occurs at $r = \sqrt{s}$. However, since this the goal of this analysis is to compute the mean M_s, it would not be correct to use this conjecture at this point. An alternative is to use the value of M_s for r_0. Since M_s is the mean of the "Gaussian like" distribution $f_s(r)$, the biggest contribution to the summation in (7) originate from r values close to M_s. Thus, approximating p_r in (7) by p_{r_0} with r_0 the (approximate) mean of $f_s(r)$, yields the recurrence relation

$$M'_{s+1} = M'_s + p_{r_0} \sum_{r=0}^{X} f_s(r) = M'_s + p_{r_0} = M'_s + \tfrac{1}{1+2M'_s}\ . \tag{5}$$

It is easily verified that this recurrence relation converges to $\sqrt{s} - \frac{1}{2}$. In fact, it can be shown analytically that the difference $M'_s - (\sqrt{s} - \frac{1}{2})$ is of the order $\mathcal{O}(\frac{\log s}{\sqrt{s}})$. In Figure 8 the exact mean M_s of the distribution, the approximation M'_s and the function $\sqrt{s}$ are shown. Clearly the exact mean of the distribution M_s is close to the "goal" $\sqrt{s}$. So indeed in the stochastic model also the square root is computed. The approximation M'_s obtained from the recurrence relation (5) is a good approximation of the exact mean M_s.

Finally we mention that it is possible to compute the standard deviation of the output, leading to

$$\sigma_X = \mathcal{O}(X^{1/4})\ .$$

This implies that, although the standard deviation increases with increasing input, the coefficient of variation, i.e., the quotient σ_X / M_X, behaves like $\mathcal{O}(X^{-1/4})$ as $X \to \infty$. Consequently the system becomes more and more deterministic as X increases.

Conclusion and Discussion

We have developed a genetic algorithm that allows us to evolve artificial mass-conserving reaction networks that compute a function in terms of amounts of input and output molecules. We have evolved networks that compute an amount of output molecules, approximately equal to the square root of the initial amount of input molecules. Several square root kernels have been identified, resulting in one elementary network motif with two reactions that provides a

good approximation of the square root function. Deterministic and stochastic analyses confirm the desired behavior of this network motif.

The artificial chemistries adopted in the reaction networks in this approach provide a rather rudimentary abstraction of biochemical networks. One limitation of the resulting networks is that they are *single shot* networks. Once the system has reached its equilibrium state, it has to be reset – waste and output molecules need to removed from the system – before a new amount of input molecules can be introduced into the system. By only allowing the input molecule to serve as a catalyst in the reactions and by providing sufficient (constant) resource molecules as additional inputs of the systems, this problem can be overcome. In future work, the assumption of mass-action kinetics is to be expanded to Michaelis-Menten kinetics, which provides more realistic reaction dynamics for the enzymatic reactions envisioned by our approach, but prove harder to grasp analytically. The current implementation can also be adapted to evolve transient behaviors, instead of solely involving the limit behavior of the model as the target of the output function. As such, the genetic algorithm can be adopted to evolve for example oscillatory networks or networks with specific transient responses to temporal inputs.

References

Arkin, A. and Ross, J. (1994). Computational Functions in Biochemical Reaction Networks. *Biophysical Journal*, 67(2):560–78.

Benenson, Y., Gil, B., Ben-Dor, U., and Adar, R. (2004). An Autonomous Molecular Computer for Logical Control of Gene Expression. *Nature*, 429:423–9.

Bhalla, U. (2003). Understanding Complex Signaling Networks through Models and Metaphors. *Progress in Biophysics and Molecular Biology*, 81(1):45–65.

Bhalla, U. and Iyengar, R. (1999). Emergent Properties of Networks of Biological Signaling Pathways. *Science*, 283:381–387.

Bray, D. (1990). Intracellular Signalling as a Parallel Distributed Process. *Journal of Theoretical Biology*, 143(2):215–31.

Bray, D. (1995). Protein Molecules as Computational Elements in Living Cells. *Nature*, 376(6538):307–12.

Buisman, H., ten Eikelder, H., Hilbers, P., and Liekens, A. (2008). Computing Algebraic Functions with Biochemical reaction Networks. *Artificial Life Journal, Special Issue on Artificial Chemistries*.

Deckard, A. and Sauro, H. (2004). Preliminary Studies on the In Silico Evolution of Biochemical Networks. *ChemBioChem*, 5(10):1423–31.

Dittrich, P. (2001). *On Artificial Chemistries*. University of Dortmund.

Dittrich, P., Ziegler, J., and Banzhaf, W. (2001). Artificial Chemistries, A Review. *Artificial Life*, 7(3):225–275.

Hucka, M., Finney, A., Sauro, H., Bolouri, H., Doyle, J., Kitano, H., Arkin, A., Bornstein, B., Bray, D., Cornish-Bowden, A., et al. (2003). The systems biology markup language (SBML): a medium for representation and exchange of biochemical network models.

Lenser, T., Hinze, T., Ibrahim, B., and Dittrich, P. (2007). Towards Evolutionary Network Reconstruction Tools for Systems Biology. *Proceedings of the Fifth European Conference on Evolutionary Computation, Machine Learning and Data Mining in Bioinformatics (EvoBIO), LNCS 4447*, pages 132–142.

Machne, R., Finney, A., Muller, S., Lu, J., Widder, S., and Flamm, C. (2006). The SBML ODE Solver Library: a native API for symbolic and fast numerical analysis of reaction networks.

Paladugu, S., Chickarmane, V., Deckard, A., Frumkin, J., McCormack, M., and Sauro, H. (2006). In silico Evolution of Functional Modules in Biochemical Networks. *IEE Proceedings System Biology*, 153:223.

Sauro, H. and Kholodenko, B. (2004). Quantitative Analysis of Signaling Networks. *Progress in Biophysics and Molecular Biology*, 86:5–43.

Steijaert, M., Liekens, A., ten Eikelder, H., and Hilbers, P. (2007). Multiple Functionalities of Biochemical Reaction Networks. *International Conference on Systems Biology (ICSB 2007)*.

Steijaert, M., ten Eikelder, H., Liekens, A., Bosnacki, D., and Hilbers, P. (2008). Stochastic Switching Behavior of a Bistable Auto-phosphorylation Network. *12th Annual International Conference on Research in Computational Molecular Biology (RECOMB 2008)*.

ten Eikelder, H., Crijns, S., Steijaert, M., Liekens, A., and Hilbers, P. (2007). Computing with Feedforward Networks of Artificial Biochemical Neurons. *2nd International Workshop on Natural Computation (IWNC 2007)*.

Tyson, J., Chen, K., and Novak, B. (2003). Sniffers, Buzzers, Toggles and Blinkers: Dynamics of Regulatory and Signaling Pathways in the Cell. *Current Opinion in Cell Biology*, 15:221–231.

Distributed Gradient Optimization with Embodied Approximation

Yaroslav Litus, Richard T. Vaughan

Autonomy Lab, Simon Fraser University, Canada
ylitus@cs.sfu.ca

Abstract

We present an informal description of a general approach for developing decentralized distributed gradient descent optimization algorithms for teams of embodied agents that need to rearrange their configuration over space and/or time, into some optimal and initially unknown configuration. Our approach relies on using embodiment and spatial embeddedness as a surrogate for computational resources, permitting the reduction or elimination of communication or shared memory for conventional parallel computation. Intermediate stages of the gradient descent process are manifested by the locations of the robots, instead of being represented symbolically. Each point in the space-time evolution of the system can be considered an approximation of the solution, which is refined by the agents' motion in response to sensor measurements. For each agent, motion is approximately in the direction of the local antigradient of the global cost function. We illustrate this approach by giving solutions to two non-trivial realistic optimization tasks from the robotics domain. We suggest that embodied approximations can be used by living distributed systems to find affordable solutions to the optimization tasks they face.

Introduction

We know by now that The World Is Its Own Best Model (Brooks (1990)). Brooks' recommendation to avoid internalizing the world - a costly and error-prone process - but rather to sense and react to it directly whenever possible, has become part of the robotics canon. This paper makes explicit an inversion of this approach, in which where the world is used to externalize a computational process.

In particular, we present an informal description of a general approach of developing decentralized distributed gradient descent optimization algorithms for teams of embodied agents that need to rearrange their configuration over space and time into some optimal and initially unknown configuration. This class of problems includes many classical mobile agent problems such as formation control, facility location, rendezvous, navigation and interference reduction. Given the intractability of finding optimal solutions to these large, high-dimensional joint state-space planning problems, gradient descent methods are commonly used to find approximate solutions.

The proposed approach is to use embodiment and spatial embeddedness as a surrogate for computational resources, permitting the reduction or elimination of communication or shared memory for conventional parallel computation. Intermediate stages of the gradient descent process are manifested by the locations of the robots, instead of being represented symbolically. Each point in the space-time evolution of the system can be considered an approximation of the solution, which is refined by the agents' motion in response to sensor measurements. For each agent, motion is approximately in the direction of the local antigradient of the global cost function.

Gradient-based formation control and navigation have been widely used in robotics (Zelek, 1999; Tanner and Kumar, 2005). The idea of embodied computation was recently presented in Hamann and Wörn (2007), however contrary to the authors' claim we believe that some globally defined algorithms can be implemented by embodied computation. Finally, Loizou and Kumar (2007) have shown biologically plausible navigation and tracking algorithms which involve using other agents locations to calculate a local gradient. To our knowledge, this paper is the first to explicitly present embodied approximation as a method to implement parallel gradient optimization algorithms.

We illustrate this approach by giving solutions to two non-trivial realistic optimization tasks from the robotics domain. This work is in the context of our interest in large-scale distributed systems such as animal colonies and multi-robot systems, which work together to solve complex tasks. We are interested in identifying mechanisms that exploit the characteristics of the embodied multi-agent domain to solve complex computational problems. We are particularly interested in solving practical resource allocation problems in multi-robot systems, with *energy autonomy* as the key motivating problem. This is the motivation for our choice of example problems: two different versions of energy-efficient robot-robot rendezvous, useful for for recharging or refueling, or as a component of various other tasks. While looking for ways to find meeting places which minimize the traveling costs for groups of robots, we developed fully decentral-

ized heuristic methods which require no range information to converge to a good approximation of the optimal solution. These heuristics afford a very simple implementation.

The key insight that underlies our approach is that the spatial configurations of the agents themselves could be considered as an approximate solution to the entire problem. An individual agent can move itself, thus refining its local component of the current solution approximation. No representation of the problem, or the current solution, needs to be held by any robot: they manifest the solution by their physical configuration. This is an example of what Payton has called "world-embedded computation" (Payton et al., 2001) and exploits the property of "strong embodiment" identified by Brooks (1990), extended to a multi-agent system.

Distributed gradient optimization with embodied approximation

Assume a system of n spatially embedded agents is described by a spatial configuration $s \in H$ where $H = H_1 \times H_2 \times \ldots \times H_n$ and H_i is a vector space of possible configurations of agent i. We will denote the i-th component of s as s_i. Every agent has control over the first-order dynamics of its own configuration which allows it to continuously change its configuration possibly with some restrictions on the speed of this change. We also assume that every agent i can sense the spatial configuration of a set of other agents ψ_i. This set can change as the system evolves. Given some cost function $J : H \rightarrow R$ we want to rearrange the system into an optimal configuration $s^* = \arg\min_s J(s)$. Note that the cost function may be parametrized by initial agent configuration s_0.

If the function J has certain mathematical properties (e.g. is continuously differentiable) then a good approximation to an optimal configuration can be found in a centralized way by using one of many gradient optimization algorithms. It will start with some initial approximation and iteratively change it in constant or decreasing steps in the direction of (approximated) antigradient. Details of these particular algorithms are irrelevant for present discussion. This centralized solution will need a central processor with amount of memory of the order of the size of the state vector s as well as means to communicate s^* to all agents once an acceptable approximation is found. Once all agents know their component of s^* they can proceed directly towards it. If J is parametrized by initial configuration of agents S_0 then the means of communicating or sensing this configuration by the central processor are required.

Since the system of n agents in question has n processors that can work in parallel it is natural to try to use this resource to get a parallel solution. Distributed implementation of iterative algorithms are well-studied (Baudet, 1978; Bertsekas, 1982; Kung, 1976). It is known that if processors synchronously compute and communicate their partial results to other processors then the resulting distributed procedure is equivalent to a single-processor implementation thus preserving its convergence properties. Further, under certain realistic conditions (like small iteration steps and bounded communication delays) the synchrony assumption can be relaxed and processors can be allowed to compute at different speeds and communicate sporadically while still giving the same convergence properties as original serial algorithm (Tsitsiklis et al., 1986). Therefore, we can assign every agent i the task of iterative optimization of its own component s_i of the global state vector s by changing s_i in the direction of the corresponding component of the antigradient vector ∇J_i and periodically communicate the current value of s_i to other agents as well as receive updates of the current approximations of $s_j, j \neq i$ from other agents. Note, that the amount of information from other processors needed to compute ∇J_i depends on the particular problem. If ∇J_i depends only on s_i then no additional information is necessary and s_i can be computed independently of other processors. If ∇J_i depends on some other components of the global state s then the current values of these components should be received from the processors which compute them. We denote the set of agents that compute components necessary to calculate ∇J_i by ξ_i. If the assumptions of Tsitsiklis et al. (1986) are met, then eventually every agent will know the approximation s_i^* and will be able to proceed towards it.

However, a physical multi-agent system is much more than simply n parallel processors. Physical embodiment implies spatial embeddedness, and for some problems these remarkable dual properties allow us to drastically reduce or totally eliminate the need to communicate intermediate results of calculations, or indeed any *description* of the problems or solutions, substituting instead direct physical observations. In addition, instead of waiting for an iterated algorithm to converge agents can perform reconfiguration *during* the optimization thus giving the resulting algorithm an attractive anytime property. The high-level description of the approach is given in Algorithm 1.

Algorithm 1 Gradient optimization with embodied approximation

1: **for all** agents i **do**
2: sense current configurations s_j^{ψ} for $j \in \psi_i \cap \xi_i$
3: update using communication current configurations s_j^{ψ} for $j \in (\{1, 2, \ldots, i-1, i+1, \ldots, n\} \backslash \psi_i) \cap \xi_i$
4: $D_i \leftarrow \nabla J_i(s_i, s_{j|j \in \xi_i})$
5: **if** ∇J_i exists and component can be improved **then**
6: move in direction $-D_i$
7: **else**
8: stay still
9: **end if**
10: communicate new configuration to other agents
11: **end for**

The key idea is to use agent configurations directly as approximations to corresponding components of the global goal configuration. Thus all agents move in the (approximated) antigradient direction which is calculated at Step 4 using their current configuration as the current approximation to the solution. The communication at Step 3 is necessary only if not all of the information from the relevant agents belonging to set ξ_i can be acquired by direct sensing at Step 2. In the best case no communication is required as all necessary information is available via sensing, so $\psi_i \supseteq \xi_i$. If some components nevertheless have to be acquired by communication, this could be done in an asynchronous sporadic manner as was argued above.

This approach makes agents themselves serve as a shared "memory" for the parallel computation they perform where the information about current approximation is embodied in physical configuration of the agents. Once the values of necessary components are available, gradient direction ∇J_i or an approximation to it can be calculated and agent can move in antigradient direction thus improving approximation of component s_i and global state s. In a certain sense, agent relocation becomes a part of a computation process, creating a parallel computer comprised by agent's processors and physical bodies of agents. The algorithm continues until convergence is reached which is detected in a way specific for the particular problem and algorithm employed.

Applications

We illustrate the idea of optimization with embodied approximation by two problems of energy-efficient multi-robot coordination which we have studied in previous work. The first problem is to assemble a team of robots at an initially unknown point which minimizes the total energy spent on relocation (Zebrowski et al., 2007). The second, more difficult problem is an instance of the multi-facility location problem which asks to plan a route for a team of robots to rendezvous a single dedicated service robot, possibly each at a *different* point (Litus et al., 2008, 2007).

Energy-efficient single-point rendezvous

Assume n robots are located at positions r_i, $i = 1 \ldots n$. When a robot moves, it expends energy proportional to the length of its trajectory. Robots have individual energy costs c_i per unit of traveled distance, thus if robot i moves from a to b, it spends $c_i||a-b||$ units of energy. Now the task is to find a point p^* which minimizes the total energy spent by all robots for meeting at that point.

Definition 1 (Energy-efficient rendezvous problem)**.** *Given robot locations $r_i \in R^d, i = 1, \ldots, n$ find rendezvous point*

$$p^* = arg\min_p J(p) = \sum_{i=1}^{n} c_i||p - r_i|| \qquad (1)$$

Historically, this problem has a variety of names including the Fermat-Steiner problem, Weber problem, single facility location problem, and the generalized Fermat-Torricelli problem. Though it is not possible to find a closed-form solution for this problem, the properties of its solutions are well known (see Gueron and Tessler (2002) for the case $n = 3$ and Kupitz and Martini (1997) for the general case). Effective numerical algorithms exist (Rosen and Xue, 1991). Interestingly, it is also possible to describe the solution using a mechanical interpretation as a system of idealized strings, pulleys and weights (Polya, 1968).

The antigradient of the cost function is

$$-\nabla J(x) = \sum_{i=1}^{n} u(x, r_i), u(a, b) = \frac{b-a}{||a-b||} \qquad (2)$$

That is, the antigradient at some point is simply a sum of unit vectors pointing in the directions of original robot locations. Note that antigradient is not defined at the locations of the robots themselves while one of these locations can be a solution. If points r_i are not collinear (not lying upon a straight line), the goal function in (1) is strictly convex, which ensures the uniqueness of p^* if $n \geq 3$. In this case solution is characterized by the following theorem (Kupitz and Martini, 1997).

Theorem 1. *If r_i are not collinear and for each point r_i*

$$||\sum_{j=1}^{n} c_j \frac{r_j - r_i}{||r_j - r_i||}|| > r_i, i \neq j \qquad (3)$$

then $p^* \neq r_i$ for any i and $\nabla J(p^*) = 0$ (the floating case). If (3) does not hold for some r_i then $p^* = r_i$ (the absorbed case).

A simple embodied approximation method gives a useful solution to this problem. Let all robots compute the solution point approximation simultaneously by checking the absorbed case condition in Theorem 1 and moving along antigradient direction if the condition is not met. This produces the distributed Algorithm 2 which runs until robots converge to a single point. Note that since robots are embodied they can not occupy the same point, thus we consider that two robots met whenever the distance between them is closer than some meeting range ϵ.

In order to calculate the antigradient at its current location, each robot needs to know the direction towards the original robot locations r_i. This could be achieved either by means of global localization and memorizing r_i or by setting static beacons at points r_i (hence the name of the algorithm). Both localization and static beacons are expensive solutions, but embodied approximation can be used once again to get rid of the necessity to calculate directions to r_i.

Once some of the robots move from their original locations r_i a new instance of the rendezvous problem is created. In this new instance all robots are located at the new

Algorithm 2 Static algorithm for energy-efficient rendezvous

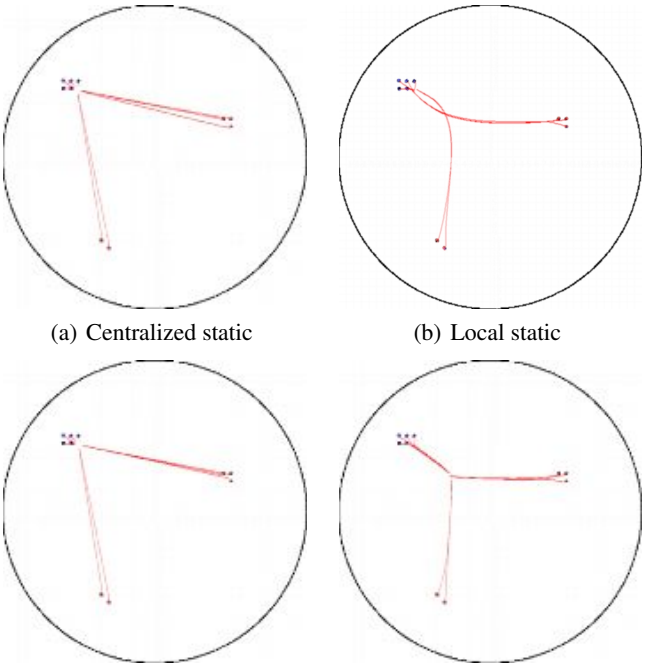
for all robots i **do**
  Update current location of self, x_i
  $D_i \leftarrow \sum_{i|x \neq r_i} c_i u(x, r_i), u(a, b) = \frac{a-b}{||a-b||}$
  if $x = r_i$ for some i **then**
    $c \leftarrow c_i$
  else
    $c \leftarrow 0$
  end if
  if $||D_i|| < c$ **then**
    stop
  else
    move in direction D_i
  end if
end for

"original" locations r_i thus sensing the instantaneous directions to the robots is enough to approximate the antigradient value. Hence memorizing r_i or using static beacons is replaced by using the robots as dynamic beacons resulting in the distributed Algorithm 3.

Algorithm 3 Dynamic algorithm for energy-efficient rendezvous

for all robots j **do**
  $A_j \leftarrow \{i|||r_i - r_j|| \leq \epsilon\}$, meaning the set of robots which are closer to r_j than meeting threshold ϵ. Note that $j \in A$, thus A has at least one element.
  $D_j \leftarrow \sum_{i \notin A} c_i \vec{d}(r_j, r_i)$.
  $c \leftarrow \sum_{i \in A} c_i$
  if $||D_j|| < c$ **then**
    stop
  else
    move in direction D_i
  end if
end for

A set of experiments was conducted to compare the performance of these algorithms with the performance of a centralized optimization algorithm (see Zebrowski et al. (2007) for details). Some typical results are shown in Table 1. The centralized static algorithm calculates the optimal meeting point p^* exactly once and commands all robots to proceed directly to that point. The centralized dynamic algorithm periodically recomputes the meeting point incorporating new positions of the robots while they move towards their rendezvous. It could be seen that distributed methods achieve the quality of solution comparable to centralized methods. The difference is explained by the fact that the trajectory along the antigradient is not straight and thus longer then the direct path to the optimal meeting locations (see Fig. 1). However, in applications that require scalability distributed



(a) Centralized static (b) Local static

(c) Centralized dynamic (d) Local dynamic

Figure 1: Typical paths taken to single point rendezvous (Map 2)

Table 1: Mean Total Energy Used For Rendezvous. All Standard Deviations $< 5\%$

	Dynamic		Static	
Map	Centralized	Distributed	Centralized	Distributed
1	149.99	151.56	149.92	152.79
2	105.21	115.21	105.25	119.00
3	77.56	80.49	77.28	81.48
4	477.89	503.18	476.97	530.11

algorithms with low demands on sensing and communication will be preferable even if they produce slightly worse results than centralized solutions. As the number of robots grows so does the running time of a centralized algorithm. Often this growth is fast enough to render the centralized solution impractical for a fairly small number of robots. At the same time scalable distributed solutions work with any number of robots, sometimes at a price of giving worse results.

Ordered Frugal Feeding Problem

Consider a team of worker robots that can recharge by docking with a dedicated refueling (or equivalently, recharging) robot called a *tanker*, as described in (Zebrowski and Vaughan, 2005). The tanker robot could remain at a fixed location, acting as a conventional charging station, or it could move to rendezvous with worker robots. Simultaneously, worker robots can wait for the tanker to come to them, or

they can move to meet the tanker.

By analogy to a mother animal attending her offspring we call this problem the "Frugal Feeding Problem". If we impose a total order in which worker robots must be met and charged, (perhaps based on urgency or some other priority scheme) we obtain the "Ordered Frugal Feeding Problem" considered here. As in a single point rendezvous problem we model locomotion costs as the weighted Euclidean distance between the origin and destination.

The Ordered Frugal Feeding Problem can be stated formally as follows:

Definition 2 (Ordered Frugal Feeding Problem)**.** *Given tanker location $p_0 \in \mathbb{R}^d$, worker locations $r_i \in \mathbb{R}^d, i = 1, \ldots, k$ find*

$$\min_{p_1,p_2,\ldots,p_k} J(p_1, p_2, \ldots, p_k) = \sum_{i=1}^{k} \big(w_0||p_i - p_{i-1}|| + w_i||r_i - p_i||\big); \quad (4)$$

Here $w_0||p_i - p_{i-1}||$ gives the cost of tanker relocation between points p_{i-1} and p_i, $w_i||r_i - p_i||$ gives the cost of moving worker i from its original position to meeting point p_i.

Definition (2) could be amended to require the tanker to return to its original location after attending all workers, perhaps to refuel itself, without affecting the presented results.

We denote solution points as p_i^*. The possibility of several robots being attended in one place is permitted, and captured by the possible coincidence of some meeting points. We define a *meeting* as the event when robots come within distance s of each other.

Unlike the single-point rendezvous case this problem involves several relocations of the system (tanker needs to go between all rendezvous points in turn). However, the embodied approximation approach is applicable and we have found a distributed algorithm which produces good approximations to the optimal solutions.

We start with considering the antigradient of the cost function J. Using Eq (2) we can express the i-th component of the antigradient as

$$-\nabla J_i(p_i) = w_0 u(p_i, p_{i-1}) + w_0 u(p_i, p_{i+1}) + w_i u(p_i, r_i), \quad (5)$$

for $i = 1..k-1$. The antigradient component for the last robot k includes one less term. Let the current position of every worker robot i represent its current approximation to the solution point p_i^*. Let the position of the tanker also represent its current approximation of the meeting point for the first robot to be charged. That is, point p_1 is simultaneously represented by tanker and worker 1. The system will perform parallel gradient descent with all robots moving along the corresponding approximation to the antigradient component given by Eq (5) calculated at their own position. In other words, robots 0 (the tanker) and 1 (the next robot to be charged) move along the approximated antigradient of the part of cost function $g(p_1) = w_0||p_0 - p_1|| + w_0||p_1 - p_2|| + w_1||p_1 - r_1||$ using the current position of robot 2 as an approximation to the unknown solution point p_2. The rest of the robots $j, j = 2, ..., k$ move along the approximated antigradient of the cost functions $f_j(p_j) = w_0||p_j - p_{j-1}|| + w_0||p_j - p_{j+1}|| + w_j||r_j - p_j||$ using r_{j-1}, r_{j+1} as approximations for the unknown solution points p_{j-1}, p_{j+1}. Algorithm 4 describes the procedure formally. Convergence of this algorithm is guaranteed, as analyzed in Litus et al. (2008).

Algorithm 4 Distributed Algorithm for Ordered Frugal Feeding Problem

```
1: i ← 1
2: define d⃗(x, y) = (y − x)/||x − y||
3: let r_0 be the current tanker position, r_i be the position
   of worker i
4: while i ≤ n do
5:   if r_0 is close to r_i then
6:     tanker charges worker i; i ← i + 1
7:   else
8:     if i = n (only one robot in queue) then
9:       the lighter of tanker and worker goes towards the
         other
10:    else
11:      r_{n+1} = r_n
12:      D⃗_0 ← w_1 d⃗(r_0, r_1) + w_0 d⃗(r_0, r_2).
13:      D⃗_i ← w_0 d⃗(r_i, r_0) + w_0 d⃗(r_i, r_{i+1})
14:      for all i < j ≤ n do
15:        D⃗_j ← w_0 d⃗(r_j, r_{j−1}) + w_0 d⃗(r_j, r_{j+1})
16:      end for
17:      for all j ∈ {0, i, i + 1, ..., n} do
18:        if ||D⃗_j|| < w_j then
19:          robot j stops
20:        else
21:          robot j proceeds in the direction D⃗_j
22:        end if
23:      end for
24:    end if
25:  end if
26: end while
```

Parallel computation of movement directions in steps 11-14 and simultaneous movement of all robots in steps 17-20 provide the scalability of this algorithm. Importantly, the per-robot, per-timestep cost is constant, so the method works for arbitrary population sizes. Unlike the single-point algorithms described in previous section, here every robot needs to know the direction to at most two other robots to calculate its movement direction.

More specifically, this algorithm requires every worker to know its own weight and the tanker weight, whether or not it is the head of the current charging queue, and the direc-

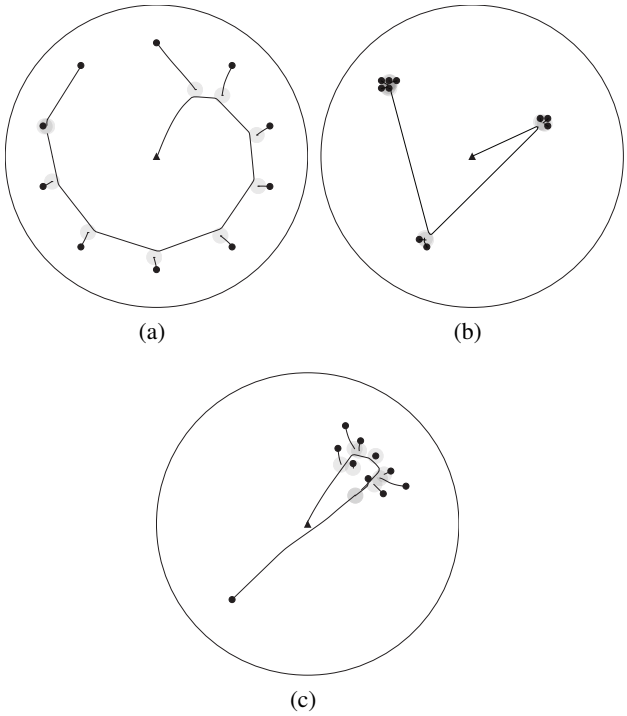


Figure 2: Some trajectories produced by distributed algorithm for Ordered Frugal Feeding Problem. Triangle represents tanker, small disks represent workers, large disks show the meeting ranges at the final point of each worker trajectory.

tion towards the next worker in the queue and the previous worker (or tanker if the robot is at the head of the queue). The tanker needs to know its own weight, the weights of the first two robots in the queue and the directions towards them. As a robot is met and charged, it is removed from the queue, and this update is broadcast to all robots.

As in the single point rendezvous case here instead of operating with the model of the world and searching for the complete solution, each robot uses the position of itself, its queue predecessor and successor as the current embodied approximation to the solution points. Every robot tries to improve the global solution quality by moving in the direction which decreases the part of the total cost function that concerns itself and its neighbors. If the robot finds itself located at the minimizing point for the current local configuration of robots, the robot stops. Fig 2 shows some trajectories produced by the algorithm.

To evaluate the performance of the algorithm we performed 3000 simulations running distributed algorithm for Ordered Frugal Feeding Problem on randomly generated instances of the problem, with initial robot locations drawn from the same uniform distribution, and 3 different uniform distributions of weights, with 1000 experiments for each weight distribution. In each experiment, 10 randomly weighted worker robots and a tanker were placed at random locations in an square arena with 20m sides. A meeting range of a 0.1m and a movement step length of 0.01m were set. The path traveled by each robot was recorded, and it's length was multiplied by the robot's weight, then summed for the population to give the total energy spent on performing the rendezvous. For comparison we computed optimal solutions of each instance using discrete search (Litus et al., 2007) on a regular grid with 40 ticks per dimension covering the embedding hypercube of original robot locations. The regularity of the grid allows us compute a lower bound of the optimal cost based on the result of discrete search by subtracting the maximum possible error due to discretization (no such quality bounds are readily available for numerical approximation methods to the continuous problem, and no closed form solution is known).

We calculated the upper bound of approximation factor for every problem instance by dividing distributed algorithm for Ordered Frugal Feeding Problem results by the lower bound of optimal cost. Table 2 reports the statistics of these approximation factor bounds for each distribution.

Table 2: Statistics of approximation factor bounds

	$w_i = 1$	$w_i \sim U[1,3)$	$w_i \sim U[1,100]$
Mean	1.19	1.22	1.31
Median	1.19	1.20	1.25
St. dev.	0.03	0.08	0.23
Skewness	0.67	1.18	4.27
Kurtosis	0.65	1.65	31.82

The results show increasing variation of the approximation factor bounds with increasing variation in robot weights. Indeed, this distributed algorithm for Ordered Frugal Feeding Problem can cause the tanker to move suboptimally in the direction of two light workers and then return to the third heavy worker. Hence, this distributed algorithm for Ordered Frugal Feeding Problem does not have a constant approximation factor. However, the average quality of solutions for uniformly distributed problem appears very good. Thus, the method could be used where the guaranteed quality of results obtained in non-scalable centralized manner should be sacrificed in favor of simple decentralized scalable solution with good average performance.

Conclusion

A general approach for development of decentralized distributed gradient descent optimization algorithms for teams of embodied agents is presented. This approach uses embodiment and spatial embedding as valuable computational resources which allow to reduce or eliminate communication or shared memory requirements for parallel computation. Spatial configuration of agents serves as an embodied representation of the current approximation to the global so-

lution which is accessed by means of sensing and refined by agents moving along local antigradient directions. Two non-trivial optimization tasks of energy-efficient path planning for mutli-robot teams were used to illustrate the embodied approximation approach. The resulting distributed algorithms show good results in experimental evaluation. These algorithms use very simple mechanisms that result in energy efficient group behavior. In both cases, computationally complex optimization tasks are solved using biologically affordable machinery of bearing-only sensing. Thus, it seems promising to look for examples of optimization by means of embodied approximation in groups of animals. Such animal strategies could and should be used to increase the efficiency, and eventually autonomy, of robot teams.

Admittedly the extent to which embodied approximation can substitute communication or shared memory is limited by the properties of sensors and environment. Limited sensor range and occlusions will limit the number of state vector components that could be accessed by sensing. However, some optimization problems by their nature need only local and limited exchange of information. For other problems embodied approximation may still significantly reduce the need for inter-agent communication and should be considered as one of the available resources.

Future work includes application of embodied approximation approach to other optimization problems emerging in multi-agent teams. A search for biological examples of parallel gradient optimization with embodied approximation is another interesting direction. Finally, accurate mathematical models of spatially embedded multi-agent systems with certain sensing capabilities could be developed and the computation power of such systems can be theoretically studied taking into account embodied approximation as a computational resource.

References

Baudet, G. M. (1978). Asynchronous iterative methods for multiprocessors. *J. ACM*, 25(2):226–244.

Bertsekas, D. P. (1982). Distributed dynamic programming. *IEEE Trans. Automat. Contr.*, 27(3):610–616.

Brooks, R. A. (1990). Elephants don't play chess. *Robotics and Autonomous Systems*, 6(1&2):3–15.

Gueron, S. and Tessler, R. (2002). The fermat-steiner problem. *The American Mathematical Monthly*, 109(5):55–129.

Hamann, H. and Wörn, H. (2007). Embodied computation. *Parallel Processing Letters*, 17(3):287 – 298.

Kung, H. (1976). Synchronized and asynchronous parallel algorithms for multiprocessors. In *Algorithms and Complexity*, pages 153–200. New York:Academic.

Kupitz, Y. and Martini, H. (1997). Geometric aspects of the generalized Fermat-Torricelli problem. *Bolyai Society Mathematical Studies*, 6:55–129.

Litus, Y., Vaughan, R., and Zebrowski, P. (2008). A distributed heuristic for energy-efficient multi-robot multi-place rendezvous. Submitted for publication.

Litus, Y., Vaughan, R. T., and Zebrowski, P. (2007). The frugal feeding problem: Energy-efficient, multi-robot, multi-place rendezvous. In *Proceedings of the IEEE International Conference on Robotics and Automation*.

Loizou, S. and Kumar, V. (2007). Biologically inspired bearing-only navigation and tracking. In *Proceedings of 46th IEEE Conference on Decision and Control*, New Orleans, USA.

Payton, D., Daily, M., Estowski, R., Howard, M., and Lee, C. (2001). Pheromone robotics. *Auton. Robots*, 11(3):319–324.

Polya, G. (1968). *Mathematics and plausible reasoning, 2 vols*. Princeton, 2 edition. vol 1. Induction and analogy in mathematics; vol 2. Patterns of plausible inference.

Rosen, J. and Xue, G.-L. (1991). Computational comparison of two algorithms for the euclidean single facility location problem. *ORSA Journal on Computing*, 3(3).

Tanner, H. G. and Kumar, A. (2005). Formation stabilization of multiple agents using decentralized navigation functions. In *Proceedings of Robotics: Science and Systems*, Cambridge, USA.

Tsitsiklis, J. N., Bertsekas, D. P., and Athans, M. (1986). Distributed asynchronous deterministic and stochastic gradient optimization algorithms. *IEEE Trans. Automat. Contr.*, 31(9):803–812.

Zebrowski, P., Litus, Y., and Vaughan, R. T. (2007). Energy efficient robot rendezvous. In *Proceedings of the Fourth Canadian Conference on Computer and Robot Vision*, pages 139–148.

Zebrowski, P. and Vaughan, R. (2005). Recharging robot teams: A tanker approach. In *Proceedings of the International Conference on Advanced Robotics (ICAR)*, pages 803–810, Seattle, Washington.

Zelek, J. (1999). Dynamic discovery and path planning for a mobile robot at a cocktail party. *Robot Motion and Control, 1999. RoMoCo '99. Proceedings of the First Workshop on*, pages 285–290.

Emergence of Glider-like Structures in a Modular Robotic System

Joseph T. Lizier[1,2], Mikhail Prokopenko[1], Ivan Tanev[3] and Albert Y. Zomaya[2]

[1]CSIRO Information and Communications Technology Centre, Locked Bag 17, North Ryde, NSW 1670, Australia
[2]School of Information Technologies, The University of Sydney, NSW 2006, Australia
[3]Department of Information Systems Design, Doshisha University, 1-3 Miyakodani, Tatara, Kyotanabe, Kyoto 610-0321, Japan
jlizier@it.usyd.edu.au

Abstract

Information-driven evolutionary design has been proposed as an efficient method for designing self-organized multi-agent systems. *Information transfer* is known to be an important component of distributed computation in many complex systems, and indeed it has been suggested that maximization of information transfer can give rise to interesting behavior and induce necessary structure in a system. In this paper, we report the first known application of a direct measure of information transfer, *transfer entropy*, as a fitness function to evolve a self-organized multi-agent system. The system evolved here is a simulated snake-like modular robot. In the most fit snakebot in the final generation, we observe coherent traveling information transfer structures. These are analogous to gliders in cellular automata, which have been demonstrated to represent the coherent transfer of information across space and time, and play an important role in facilitating distributed computation. These observations provide evidence that using information transfer to drive evolutionary design can produce useful structure in the underlying system.

Introduction

The principle of self-organization is well known to offer the advantages of flexibility, robustness and scalability over centralized system designs (Prokopenko et al., 2006a). Most self-organized solutions are currently designed using a genetic algorithm of some form, with fitness functions measuring achievement of the task required of the system (*task-based evolution*). Several authors have recently been investigating the potential for *information-driven evolutionary design* to push the advantages of self-organization even further, e.g. (Prokopenko et al., 2006a; Polani et al., 2007; Klyubin et al., 2005; Sporns and Lungarella, 2006). This concept proposes the use of information-theoretical measures of the information processing carried out by the system as generic fitness functions in evolutionary design. From an engineering perspective, template-based evolution for generic information processing skills could be simpler and afford a framework based approach to such design of self-organized systems. It also provides to us the potential to better understand the evolved solutions, and more importantly the opportunity to study and understand the emergence rather than engineering of intelligence (Polani et al., 2007).

We believe information-driven self-organization is best facilitated using measures of the information dynamics of distributed computation (Lizier et al., 2007). Any task we wish to evolve the system to solve involves a distributed computation, so evolving for the fundamental building blocks of the computation is a direct way to allow that computation to emerge. We could evolve directly for a particular computational property (e.g. information storage as opposed to transfer), or for a mix of those properties.

Information transfer has been suggested to be a particularly important fitness function here. It has been conjectured that information transfer can give rise to interesting behavior and induce necessary structure in a multi-agent system (Prokopenko et al., 2006a). One inspiration of this viewpoint is the concept of *empowerment* (Klyubin et al., 2005), which refers to an agent's self-perception of its influence over its environment. Alluding to (but not directly measuring) information transfer, it is quantified as the channel capacity between an agent's actuators and sensors through the environment. Maximization of empowerment has been suggested to be an intrinsic selection pressure[1]. With or without the presence of explicit actuator-sensor channels, we expect information transfer to be a useful fitness function because of its important role in distributed computation.

Here, we present the first experiment of the use of a direct measure of information transfer, transfer entropy (Schreiber, 2000), as the sole fitness function in an evolutionary design task. An initial aim of the experiment is to check whether information transfer underpins co-ordinated motion, as was suggested in previous work (Prokopenko et al., 2006a). More importantly, we aim to investigate what type of behavior emerges when a system is evolved to maximize information transfer. Much previous work on information-driven evolution has sought to confirm whether it can approximate direct evolution for a given task. Here, we simply seek to investigate what type of solution or computation

[1]The justification or otherwise of the suggestion that *natural* evolution is driven by the intrinsic forces of information processing is irrelevant to whether information-driven evolutionary design can be used as a successful tool for *artificial* systems.

is generated by evolution for information transfer, and hypothesize that it will induce useful computation in the system. Our findings will help us to understand the role that information transfer can play in a unified framework for information-driven evolutionary design, focusing on the information dynamics of distributed computation.

We use a snake-like modular robot (the *snakebot*) for experimentation: information structure has been observed to emerge previously with a fitness function for fastest motion (Prokopenko et al., 2006b), and conversely fast motion has emerged from evolution with a measure of co-ordination as the fitness function (Prokopenko et al., 2006a). We measure information transfer using the transfer entropy (Schreiber, 2000) between neighboring modules of the snakebot, and evolve the snakebot to maximize this quantity. Information transfer in this fashion could be utilized by the snake in leading to co-ordinated motion between the modules, communicating information about obstacles, or driving new behaviors in a given direction along the snake.

We report that *coherent* traveling information transfer structures were observed to emerge (using local transfer entropy (Lizier et al., 2008a)) in the evolved snakebot. We say "emerged" because while high information transfer was selected for, local coherent structures were not part of the specification. This is an important finding, because these structures are analogous to *glider* structures in cellular automata (CAs). Gliders are known to be the information transfer agents in CAs, providing for long-range correlations across space and time and playing a fundamental role in the distributed computation carried out in the CA (Lizier et al., 2008a). As such, we have provided evidence that using a direct measure of information transfer as a fitness function in information-driven evolutionary design can indeed produce useful structure in the system.

Information-driven evolution

Task-based evolution, the incumbent method of designing self-organized systems, can be impractical. Hand-crafting fitness functions for every task can be time-consuming and tedious, and requires specialized human understanding of the task. It has the potential to under-specify the problem (thereby solving a different task) or perhaps over-specify it (leading to an inflexible design). Also, the intelligent designer may not be completely sure of how to measure performance of the required task, or this may be difficult (e.g. measuring speed may require extra sensors). Furthermore, if the initial task-based fitness landscape is flat and features no gradients, task-based evolution has no foothold around which to begin designing a solution. Finally, evolution often delivers intricate solutions for which (human) system managers cannot understand the inner workings: this is particularly undesirable for critical systems where maintenance or prediction of behavior is required.

As an alternative, information-driven evolutionary design proposes the use of information-theoretic measures to design the required information processing structure in self-organized systems. This has been prompted by observations of complexity to grow or necessary information-theoretic structure to emerge during task-based evolution. Growth of complexity during evolution has been observed by Adami (2002) (measuring "physical complexity" in the Avida simulation system) and Yaeger and Sporns (2006) (measuring neural complexity of evolved agents in the PolyWorld simulation system). Looking at evolution for particular tasks, Prokopenko et al. (2006b) observed co-ordination (measured as excess entropy (Crutchfield and Feldman, 2003)) to increase in snakebots evolved for maximum velocity, and Baldassare et al. (2004) observed a decrease in entropy in a swarm evolved for co-ordinated motion.

These observations suggest that such information-theoretic metrics could be used themselves in information-driven evolutionary design. This idea is fundamentally based on the theory that information structure is vital to the emergence of self-organized intelligence (Polani et al., 2007). The concept could provide a consistent framework for the evolutionary design of self-organized systems, using template-based evolution for required computational tasks. This framework would be able to produce useful structure where task-based evolution faces initially flat task-based fitness landscapes, perhaps serving as a platform from which to launch better-equipped task-based evolution. Furthermore, it may provide solutions which are simpler for humans to understand in terms of the underlying information dynamics. Perhaps most important is the potential for this approach to provide insight into the emergence rather than engineering of intelligence (Polani et al., 2007), and thereby facilitate unsupervised learning.

Several examples of successful information-driven evolutionary design exist in the literature. Maximization of empowerment has been shown to induce a necessary structure in agent's behavior by Klyubin et al. (2005). Sporns and Lungarella (2006) have evolved hand-eye co-ordination to grab a moving object using maximization of neural complexity, and demonstrated that this solution contained more intrinsic diversity than solutions from task-driven evolution; the increased diversity may afford greater flexibility to the system. Prokopenko et al. (2006a) were able to evolve fast-moving snakebots using maximization of an information-theoretic measure of co-ordination. Also, Sperati et al. (2007) have observed interesting periodic behavior and complex structure in groups of robots which were evolved to maximize their mutual information.

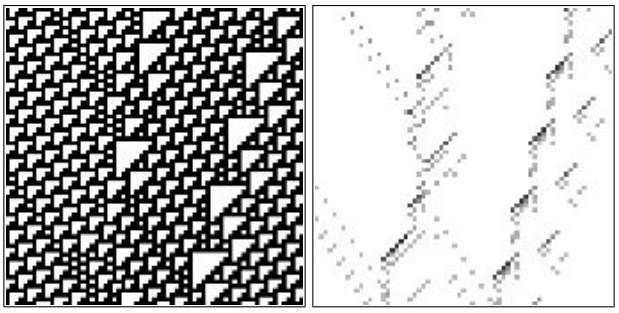
We suggest that the information dynamics of distributed computation (Lizier et al., 2007, 2008a) provide the most intuitive basis for information-driven evolution. These information dynamics are the primitive functions of Turing universal computation, i.e. *information storage*, *transfer* and *modification*. Any task we wish the system to achieve in-

volves some form of computation. As such, using a framework for distributed computation allows us to target the evolution toward the computational requirements of the task at hand, i.e. selecting either the most relevant computational function as the fitness function, or balancing the functions in a more complex manner. Importantly, using such a framework provides a basis through which to understand the computation carried out by the evolved solution. Also, guiding a system toward the building blocks of distributed computation is perhaps the most intuitive way to facilitate the emergence of collective intelligence.

Information transfer is an important candidate fitness function here. It has been observed to be a critical part of the dynamics of many complex systems, for example being manifested in dipole-dipole interactions in microtubules which give rise to self-organization there (Brown and Tuszynski, 1999). Another important example are particles or gliders in CAs (e.g. see Fig. 1), which are coherent traveling information transfer structures in those systems (Lizier et al., 2008c). Much importance has been placed on the role of gliders in CA dynamics; in fact, they have been demonstrated to transport information for the distributed computation carried out in CAs (Lizier et al., 2008c). For example in a density-classification task, gliders appear to transport information about the density in the region of the CA where they originated, with glider collisions processing this information to make a decision about the overall density (Mitchell et al., 1994). Information transfer is also related to the concept of empowerment (Klyubin et al., 2005), with much importance placed on the maximization of the capacity of the information channel between an agent's actuators and sensors here. Importantly also, it has long been conjectured that information transfer is maximized in the vicinity of an order-chaos phase transition (Langton, 1990), where critical dynamics are said to facilitate the emergence of complex computation. Several authors have since inferred this conclusion from related measures (Solé and Valverde, 2001), however evidence from a directed, dynamic measure of information transfer has only recently been provided (Lizier et al., 2008b). In the following section, we describe this measure of information transfer.

Information transfer

Our measure of information transfer is of course found in the domain of information theory (MacKay, 2003), which is proving to be a useful framework for the analysis and design of complex systems, e.g. (Prokopenko et al., 2006a). The fundamental quantity in this domain is the (Shannon) *entropy*, which represents the uncertainty in a sample x of a random variable X: $H_X = -\sum_x p(x) \log_2 p(x)$ (all with units in bits). The *joint entropy* of two random variables *X* and *Y* is a generalization to quantify the uncertainty of their joint distribution: $H_{X,Y} = -\sum_{x,y} p(x,y) \log_2 p(x,y)$. The *conditional entropy* of *X* given *Y* is the average uncertainty that remains about *x* when *y* is known: $H_{X|Y} = -\sum_{x,y} p(x,y) \log_2 p(x|y)$. The *mutual information* between *X* and *Y* measures the average reduction in uncertainty about *x* that results from learning the value of *y*, or vice versa: $I_{X;Y} = H_X - H_{X|Y}$. The *conditional mutual information* between *X* and *Y* given *Z* is the mutual information between *X* and *Y* when *Z* is known: $I_{X;Y|Z} = H_{X|Z} - H_{X|Y,Z}$.



(a) (b)

Figure 1: Elementary CA rule 110. (a) Raw states. (b) Local transfer entropy with $k = 16$ (for transfer one step to the left per time step) highlights glider structures.

The mutual information has previously been used as a de facto measure for information transfer (e.g. by Solé and Valverde (2001)), however this approach is criticized by Schreiber (2000) as a symmetric measure of statically shared information. To address these concerns, Schreiber introduced the transfer entropy to quantify the *information transfer* between a source and a destination agent as the average information provided by the source about the destination's next state that was not contained in the past of the destination. This formulation provides a properly directional and dynamic measure of information transfer. The transfer entropy is the average mutual information between the previous state of the source[2] y_n and the next state of the destination x_{n+1}, *conditioned* on the past of the destination $x_n^{(k)}$:

$$T_{Y \to X}(k) = \sum_{u_n} p(u_n) \log_2 \frac{p(x_{n+1}|x_n^{(k)}, y_n)}{p(x_{n+1}|x_n^{(k)})}. \quad (1)$$

This average is over all state transition tuples $u_n = (x_{n+1}, x_n^{(k)}, y_n)$. From another perspective, it is also an average over a *local transfer entropy* (Lizier et al., 2008c) at all observed time points:

$$t_{Y \to X}(n+1, k) = \log_2 \frac{p(x_{n+1}|x_n^{(k)}, y_n)}{p(x_{n+1}|x_n^{(k)})}, \quad (2)$$

$$T_{Y \to X}(k) = \langle t_{Y \to X}(n, k) \rangle \quad (3)$$

[2]The transfer entropy can be formulated using the l previous states of the source. However, where only the previous state is a causal information contributor, we set $l = 1$ to measure direct transfer only at step n.

In general, these measures are only completely accurate in the limit $k \to \infty$ (Lizier et al., 2008c), since this removes all information that was already in the history of the destination from being mistaken as transfered. This is computationally infeasible however, so we use as large a history k as is facilitated by our observation set.

The transfer entropy can also be formulated to condition on the states of all other *causal information contributors* to the destination, so as to completely account for the contribution of the source Y. This form is known as the *complete* transfer entropy (see Lizier et al. (2008c)). The formulation in Eq. (2) is then labeled the *apparent* transfer entropy (note: in this paper, we refer to this form unless otherwise stated).

The transfer entropy has been studied in a number of interesting applications, for example in characterizing information flow in sensorimotor networks by Lungarella and Sporns (2006). Bertschinger et al. (2006) used the transfer entropy to investigate the distinction of a system from its environment, and the autonomy of the system. Studies of the local transfer entropy in CAs provided the first quantitative evidence for the long-held conjecture that gliders are the information transfer agents therein (Lizier et al., 2008c) (see Fig. 1). Application to random boolean networks (RBNs) suggests that the apparent transfer entropy is maximized in the vicinity of a phase transition from ordered to chaotic behavior, while the complete transfer entropy continues increasing into the chaotic regime (Lizier et al., 2008b). Large apparent transfer entropy appears to indicate that the dynamics support coherent information transfer (in the form of gliders in CAs) as an important component of complex distributed computation (Lizier et al., 2008a).

To compute the transfer entropy for continuous variables, a simple approach is to discretize the continuous variables and apply Eq. (1), however with a slight increase in effort, one can remain in the continuous regime. In doing so, Schreiber (2000) recommends using the method of *kernel estimation* to estimate the required probabilities, rather than an approach based on correlation integrals. (The same technique is used under different guises in computing the "pattern entropy" by Dettmann and Cohen (2000) and the "approximate entropy" by Pincus and Singer (1996)). This method has been used, for example, to compute transfer entropy in signal transduction by calcium ions by Pahle et al. (2008). With the kernel estimation method, the joint probability of the state transition tuple $u_n = (x_{n+1}, x_n^{(k)}, y_n)$ for example is estimated by counting similar tuples:

$$\hat{p}_r(u_n) = \frac{1}{N} \sum_{n'} \Theta \left(\left| \begin{pmatrix} x_{n+1} - x_{n'+1} \\ x_n^{(k)} - x_{n'}^{(k)} \\ y_n - y_{n'} \end{pmatrix} \right| - r \right), \quad (4)$$

where by default Θ is the step kernel ($\Theta(x > 0) = 1$, $\Theta(x \leq 0) = 0$) using the precision r, and the norm $|\cdot|$ is the maximum distance, though other choices are possible. The average transfer entropy $T_{Y \to X}(k)$ is then computed as the average of local transfer entropies (see Eq. (3) and Eq. (2)), where each local transfer entropy uses these kernel estimations to compute the relevant probability distribution functions. That is, computation of the average transfer entropy for continuous variables is *necessarily* a computation over each local point in time rather than over all possible state transition tuples. Here, we will present the first use of the local transfer entropy values for continuous variables.

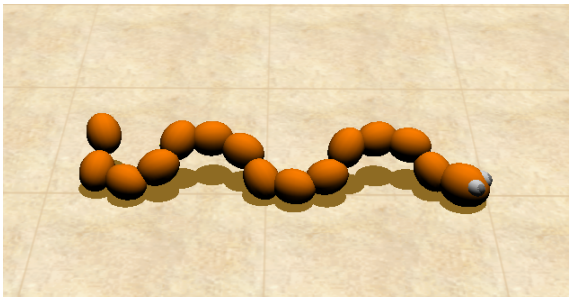


Figure 2: Snakebot

Evolving the snakebot for maximum information transfer

The snakebot is a snake-like modular robot, introduced in (Tanev et al., 2005), which is simulated in the Open Dynamics Engine (ODE). As shown in Fig. 2, it consists of a set of identical spherical morphological segments which are linked by universal joints. The joints each have two actuators for joint rotation, which are oriented vertically and horizontally in the initial standstill position of the snakebot, and all have identical angle limits. No anisotropic friction between the morphological segments and the surface is considered. The genome for the snakebot is an algebraic expression for the *desired* turning angles of its horizontal and vertical actuators as a function of time and actuator index. The periodic functions $\sin$ and $\cos$ are included in the function set, providing support for periodic gaits. The turning angles however are constrained by interactions between the segments and with the terrain; as such the *actual* actuator angles represent the emergent dynamics. Here, $\alpha_{i,n}$ and $\beta_{i,n}$ represent the actual horizontal and vertical turning angles respectively at time step n, where i is the actuator index (so $1 \leq i \leq S$ where $S = 14$ is the number of joints), and $1 \leq n \leq N$ for $N = 1800$ time steps in the simulation run.

Initial experiments to evolve fastest motion in any direction indicated that side-winding motion (i.e. locomotion predominantly perpendicular to the long axis of the snakebot) provided superior speed characteristics (Tanev et al., 2005). As previously mentioned, subsequent experiments observed the increase in co-ordination (as excess entropy) with this evolution (Prokopenko et al., 2006b), and then evolved similar fast moving side-winding locomotion using this mea-

sure of co-ordination as a fitness function (Prokopenko et al., 2006a). In capturing correlation across space and time, the (two-dimensional) excess entropy is something of an overall measure of distributed computation which balances the underlying components of information storage and transfer. Here, we evolve the snakebot using transfer entropy, in order to maximize the information transfer component of distributed computation. It was suggested in (Prokopenko et al., 2006a) that information transfer underpinned co-ordinated motion. An information transfer is certainly required in a transient sense to achieve co-ordinated motion, but the level of information transfer in this initial phase may not be very significant compared to the information transfer averaged over longer experimental periods for other behaviors. The evolution of the snakebot here will take place in a flat environment. We will observe what types of behavior emerge as a result of selecting for information transfer.

In evaluating the fitness of each snakebot after it is simulated for N time steps, we compute the average transfer entropy $T_{i+1\to i}(k)$ between each pair of consecutive modules $i+1$ and i, in the direction from the tail toward the head (i.e. decreasing module number i). The transfer entropy is computed using the time series of actual horizontal turning angles $\alpha_{i,n}$. Kernel estimation is used with these continuous values, with r set to one quarter of the standard deviation of the turning angles. Also, we use the default step kernel and maximum distance norm, ignoring matched pairs within 20 time steps *and* neighboring modules to avoid spurious dynamic correlations (as recommended by Schreiber (2000)). The direction of tail toward head is selected because each module only applies desired turning angles to the actuators in front of it (i.e. in the direction of the head), thereby giving preferential treatment to information traveling in this direction. Although it is possible for information to be transferred across more than one joint per time step, we consider only consecutive pairs since this is likely to be the dominant transfer mode. Also, as per footnote 2, we only consider transfer from a single previous state of the source variable, so as to consider information transferred directly at the given time step. We use a past history length $k = 30$ (as for the correlation entropy calculations in Prokopenko et al. (2006a)). This is large enough to eliminate information storage from the calculation (see Results), while allowing adequate sampling of the underlying distributions (because the presence of $\sin$ and $\cos$ functions mean that the emergent turning angle sequences are generally quasi-periodic and therefore much of the state space of $\alpha_{i,n}^{(k)}$ remains unexplored). Our *fitness function* is then the average of these transfer entropies over all $S-1$ consecutive module pairs for the given snakebot:

$$T_{tail\to head}(k) = \frac{1}{S-1}\sum_{i=1}^{S-1} T_{i+1\to i}(k). \qquad (5)$$

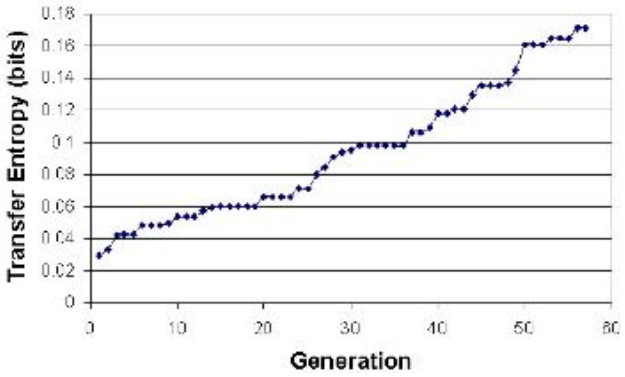


Figure 3: Snakebot fitness (average transfer entropy $T_{tail\to head}(k=30)$) per generation, plotted for the best performer in each generation.

The Genetic Programming (GP) techniques used for snakebot evolution are described by Tanev et al. (2005). The snakebots evolve within a population of 200 individuals, with the best performers selecting using the fitness function described above. No minimum limit is placed on how far the snakebot moves, since we are not evolving for fast locomotion. The selection is based on a binary tournament with selection ratio of 0.1 and reproduction ratio of 0.9. Random subtree mutation is used with a ratio of 0.01.

Results and discussion

First, we note that snakebots exhibiting a high degree of co-ordinated motion (as exemplified by most fit individual from (Prokopenko et al., 2006a)) were found to have significantly lower transfer entropy than individuals specifically evolved to maximize transfer entropy (e.g. 0.007 bits versus 0.175 bits for the most fit snakebot here). Highly co-ordinated snakebots exhibited very short transients before becoming co-ordinated, and minimal transfer entropy in their ongoing behavior. Co-ordinated motion is certainly more strongly associated with memory (in fact is a distributed memory (Lizier et al., 2008a)) than information transfer. When neighboring modules achieve perfect co-ordination, they have effectively reached a periodic attractor: their next states are completely predictable from their individual pasts, and so no additional information from the neighbor is measured as transfer entropy. It is possible that transfer entropy might be measured to be higher for snakebots attempting co-ordinated motion in a challenging environment, where information transfer in the longer and more significant transient toward co-ordination may play an important role in the dynamics.

In our evolution of snakebots for transfer entropy, the growth in the average transfer entropy $T_{tail\to head}(k=30)$ of the most fit snakebot in each generation is shown in Fig. 3.

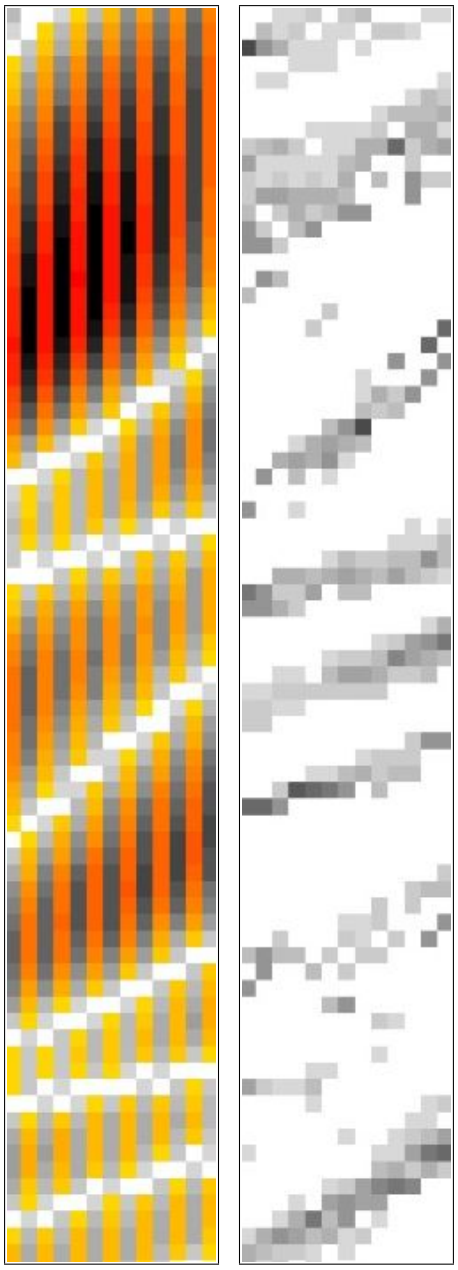
We will focus on the most fit individual in the final (57th)

generation as the result of this evolution, which had an average transfer entropy of 0.175 bits between neighboring modules toward the head per time step. This snakebot did not display a fast, well co-ordinated side-winding locomotion. Instead, it displayed a complex form of wriggling behavior, where thrashing of the tail appeared to drive new behavior along the body of the snake, achieving a slow movement to the side[3]. The dynamics of this behavior are clearer when examining the time-series of the actual horizontal turning angles $\alpha_{i,n}$, as displayed in Fig. 4(a). Here, we see that coherent waves of behavior are consistently traveling along the snakebot, from the tail toward the head. Each wave involves the modules turning in alternating directions along the snake (visible in color image online), reaching a maximum angle then coming back to a rest position. The modules then swap their turning angles in the next wave. Importantly, these waves are not completely periodic, allowing scope for information transfer effects.

Already, we note a fairly clear correspondence to emergent traveling structures in microtubules and gliders in CAs, however to confirm the information transfer properties, we examine the local transfer entropy profile in Fig. 4(b). The local transfer entropy profile here tells us much more about the snakebot dynamics than the average transfer entropy does (as was observed for CAs in (Lizier et al., 2008c)). As expected, we confirm that we have coherent traveling waves of information transfer moving along the snakebot from the tail toward the head, which coincide in direction and approximately in time with the time-series waves previously observed. As an example, note the images of the snakebot in Fig. 5 with modules colored to indicate local transfer entropy (also, videos with the modules of the snake highlighted according to their local transfer entropy are available online, see footnote 3). We can be confident that the information transfer measured is not misattributed information storage, because our use of $k = 30$ considers a longer past history than the length of the time-series waves here. Note that these coherent transfer structures were not observed in fully-coordinated or random snakebots.

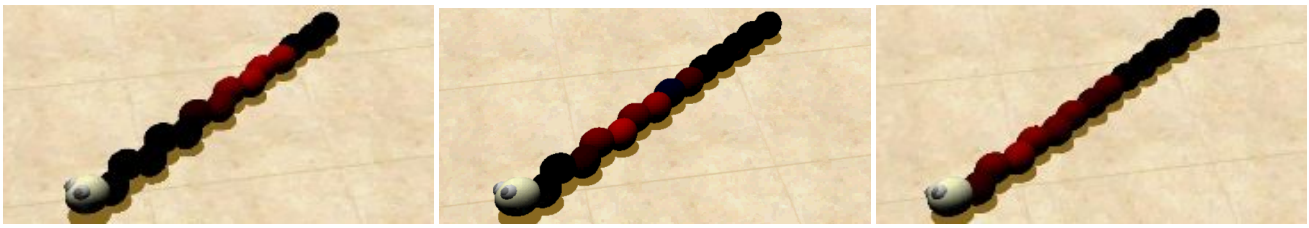
There is a wide variation in the types of such information transfer structures observed here: some move faster than others (indicated by a flatter structure), some are more highly localized in time (thinner structures), some contain higher local transfer entropies (darker coloring), and some do not coherently travel the whole way along the body of the snakebot. Importantly, none of these differences are detectable by superficial examination of the time-series of the actual actuator angles. Indeed, apart from their coincidence in direction and approximately in time, there is little correspondence between the time-series waves and the informa-

[3]Videos of the snakebot, showing raw motion and local transfer entropy are available at http://www.it.usyd.edu.au/~jlizier/publications/08ALifeSnakebotTe or http://www.prokopenko.net/modular_robotics.html



(a) (b)

Figure 4: Local apparent transfer entropy highlights "gliders" in the evolved snakebot. (a) Raw actuator turning angles for each of the 13 destination modules (head at left, tail at right) of the snakebot for 76 consecutive time steps (time increases down the page): grayscale represents a positive turning angle, yellow-red (color online) represents a negative turning angle; range is -50 to 50 degrees. (b) Local transfer entropy $t_{i+1\rightarrow i}(n, k = 30)$ into each of the 13 information destination modules of the snakebot, between consecutive modules in the tail $\rightarrow$ head direction: grayscale, range 0.0 bits (white) to 2.8 bits (black).



(a) (b) (c)

Figure 5: Snakebot modules colored to indicate incoming local transfer entropy (black is 0.0 bits, red is 2.8 bits; color online) from neighboring module toward the tail, for three consecutive time steps. The information transfer from the tail appears to communicate a straightening behavior here.

tion structure that is obvious to the observer. Certainly, there is no simple method of using the time-series waves to infer the location in time of the local information transfer structures: these are observed to begin and end at various time points within the time-series waves. Local transfer entropy reveals the *precise* space-time dynamics of the manner in which the tail drives new behavior in the snakebot in a way not possible by examining the time-series alone.

As coherent traveling local information transfer, these structures are clearly analogous to gliders in CAs (see Fig. 1). This finding is significant because of the important role that gliders play in CA dynamics, where they coherently transfer information relevant to the collective computation of the CA. We previously noted that the coincidence of gliders and coherent information transfer with a maximization of (apparent) transfer entropy (Lizier et al., 2008a). Here, we have demonstrated the emergence of glider-like structures when (apparent) transfer entropy is optimized, *without* explicitly selecting for such local coherence. This suggests that coherent glider-like structures are the most efficient mode of (apparent) information transfer. This has significant implications for glider-like structures observed in natural systems, e.g. dipole-dipole interactions in microtubules (Brown and Tuszynski, 1999), which could have evolved to exploit this efficient mode of information transfer where coherent communication or effect over some distance is beneficial.

The coherence of glider structures is of particular importance to the computation in CAs; without coherence of information transfer, complex computation does not appear to take place (Lizier et al., 2008a,b). A second requirement for such truly distributed computation though is *bidirectional* information transfer. Here, with strong information transfer encouraged in one direction only, although we have demonstrated the emergence of an important building block for non-trivial computation, we have evolved only a trivial type of computation. (This is effectively the reason that there are very few points of negative local transfer entropy measured in the snakebot here). In future work, we will build on our results here to evolve bidirectional information transfer for true distributed computation.

Conclusion

We have presented the first experiment of the use of transfer entropy as a generic fitness function for information-driven evolutionary design. We have demonstrated that maximizing information transfer in this manner can lead to the emergence of *coherent* transfer structures which, as manifested by gliders, are known to underpin distributed computation in CAs. Here, this useful generic skill was not fully capitalized on by the snakebot, but the important finding is that the use of information transfer as a fitness function led to the emergence of this computational capability. Also, our experiment implies that glider-like structures are the most efficient mode of coherent information transfer, which is itself significant insight into the nature of information transfer.

All agent-based systems compute; indeed it is their computation that makes them useful to us. Here, the snake computes where to move. While information transfer does not appear to be important for co-ordinated motion in flat environments, it could underpin computation for tasks such as successful navigation in challenging environments, where different parts of the body could sample many sections of the environment in parallel, and communicate information about the environment along the structure. Information transfer could be used to develop the required computational capability for tasks such as these in future work.

We intend to explore the use of information transfer in information-driven evolutionary design in other settings where bidirectional information transfer may be required for distributed computation. We also intend to investigate the use of the other information dynamics of computation (information storage and modification) (Lizier et al., 2007) in such design, and explore the circumstances under which each should be used and indeed how they can be used together.

Acknowledgements

JL thanks Jürgen Pahle for discussions on local transfer entropy for continuous-valued variables. IT was supported in part by the National Institute of Information and Communications Technology of Japan.

References

Adami, C. (2002). What is complexity? *Bioessays*, 24(12):1085–1094.

Baldassare, G., Parisi, D., and Nolfi, S. (2004). Measuring coordination as entropy decrease in groups on linked simulated robots. In *Proceedings of the International Conference on Complex Systems (ICCS2004), Boston*. To be published.

Bertschinger, N., Olbrich, E., Ay, N., and Jost, J. (2006). Information and closure in systems theory. In Artmann, S. and Dittrich, P., editors, *Proceedings of the 7th German Workshop on Artificial Life (GWAL-7), Jena, Germany*, Amsterdam. IOS Press.

Brown, J. A. and Tuszynski, J. A. (1999). A review of the ferroelectric model of microtubules. *Ferroelectrics*, 220:141156.

Crutchfield, J. P. and Feldman, D. P. (2003). Regularities unseen, randomness observed: Levels of entropy convergence. *Chaos*, 13(1):25–54.

Dettmann, C. P. and Cohen, E. G. D. (2000). Microscopic chaos and diffusion. *J. Stat. Phys.*, 101(3):775–817.

Klyubin, A. S., Polani, D., and Nehaniv, C. L. (2005). All else being equal be empowered. In Capcarrere, M. S., Freitas, A. A., Bentley, P. J., Johnson, C. G., and Timmis, J., editors, *Proceedings of the 8th European Conference on Artificial Life (ECAL 2005), Kent, UK*, volume 3630 of *Lecture Notes in Computer Science*, pages 744–753, Berlin / Heidelberg. Springer.

Langton, C. G. (1990). Computation at the edge of chaos: phase transitions and emergent computation. *Physica D*, 42(1-3):12–37.

Lizier, J. T., Prokopenko, M., and Zomaya, A. Y. (2007). Detecting non-trivial computation in complex dynamics. In Almeida e Costa, F., Rocha, L. M., Costa, E., Harvey, I., and Coutinho, A., editors, *Proceedings of the 9th European Conference on Artificial Life (ECAL 2007), Lisbon, Portugal*, volume 4648 of *Lecture Notes in Artificial Intelligence*, pages 895–904, Berlin / Heidelberg. Springer.

Lizier, J. T., Prokopenko, M., and Zomaya, A. Y. (2008a). A framework for the local information dynamics of distributed computation in complex systems. Submitted to Physica D.

Lizier, J. T., Prokopenko, M., and Zomaya, A. Y. (2008b). The information dynamics of phase transitions in random boolean networks. In *Proceedings of the Eleventh International Conference on the Simulation and Synthesis of Living Systems (ALife XI), Winchester, UK*. MIT Press. to appear.

Lizier, J. T., Prokopenko, M., and Zomaya, A. Y. (2008c). Local information transfer as a spatiotemporal filter for complex systems. *Physical Review E*, 77(2):026110.

Lungarella, M. and Sporns, O. (2006). Mapping information flow in sensorimotor networks. *PLoS Computational Biology*, 2(10):e144.

MacKay, D. J. (2003). *Information Theory, Inference, and Learning Algorithms*. Cambridge University Press, Cambridge.

Mitchell, M., Crutchfield, J. P., and Hraber, P. T. (1994). Evolving cellular automata to perform computations: Mechanisms and impediments. *Physica D*, 75:361–391.

Pahle, J., Green, A. K., Dixon, C. J., and Kummer, U. (2008). Information transfer in signaling pathways: a study using coupled simulated and experimental data. *BMC Bioinformatics*, 9:139.

Pincus, S. and Singer, B. H. (1996). Randomness and degrees of irregularity. *Proceedings of the National Academy of Sciences*, 93(5):2083–2088.

Polani, D., Sporns, O., and Lungarella, M. (2007). How information and embodiment shape intelligent information processing. In Lungarella, M., Iida, F., Bongard, J., and Pfeifer, R., editors, *Proceedings of the 50th Anniversary Summit of Artificial Intelligence, New York*, volume 4850 of *Lecture Notes in Computer Science*, pages 99–111, Berlin / Heidelberg. Springer.

Prokopenko, M., Gerasimov, V., and Tanev, I. (2006a). Evolving spatiotemporal coordination in a modular robotic system. In Nolfi, S., Baldassarre, G., Calabretta, R., Hallam, J., Marocco, D., Meyer, J.-A., and Parisi, D., editors, *Proceedings of the Ninth International Conference on the Simulation of Adaptive Behavior (SAB'06), Rome*, volume 4095 of *Lecture Notes in Artificial Intelligence*, pages 548–559. Springer Verlag.

Prokopenko, M., Gerasimov, V., and Tanev, I. (2006b). Measuring spatiotemporal coordination in a modular robotic system. In *Proceedings of the 10th International Conference on the Simulation and Synthesis of Living Systems (ALifeX), Bloomington, Indiana, USA*. MIT Press.

Schreiber, T. (2000). Measuring information transfer. *Physical Review Letters*, 85(2):461–464.

Solé, R. V. and Valverde, S. (2001). Information transfer and phase transitions in a model of internet traffic. *Physica A*, 289(3-4):595–605.

Sperati, V., Trianni, V., and Nolfi, S. (2007). Evolving coordinated group behaviors through maximization of mean mutual information. Submitted to Swarm Intelligence.

Sporns, O. and Lungarella, M. (2006). Evolving coordinated behavior by maximizing information structure. In Rocha, L. M., Yaeger, L. S., Bedau, M. A., Floeano, D., Goldstone, R. L., and Vespignani, A., editors, *Proceedings of the Tenth International Conference on Simulation and Synthesis of Living Systems (ALifeX), Bloomington, Indiana, USA*, pages 323–329. MIT Press.

Tanev, I., Ray, T., and Buller, A. (2005). Automated evolutionary design, robustness, and adaptation of sidewinding locomotion of a simulated snake-like robot. *IEEE Transactions on Robotics*, 21(4):632–645.

Yaeger, L. and Sporns, O. (2006). Evolution of neural structure and complexity in a computational ecology. In Rocha, L. M., Yaeger, L. S., Bedau, M. A., Floeano, D., Goldstone, R. L., and Vespignani, A., editors, *Proceedings of the Tenth International Conference on Simulation and Synthesis of Living Systems (ALifeX), Bloomington, Indiana, USA*, pages 330–336. MIT Press.

The Information Dynamics of Phase Transitions in Random Boolean Networks

Joseph T. Lizier[1,2], Mikhail Prokopenko[1] and Albert Y. Zomaya[2]

[1]CSIRO Information and Communications Technology Centre, Locked Bag 17, North Ryde, NSW 1670, Australia
[2]School of Information Technologies, The University of Sydney, NSW 2006, Australia
jlizier@it.usyd.edu.au

Abstract

Random Boolean Networks (RBNs) are discrete dynamical systems which have been used to model Gene Regulatory Networks. We investigate the well-known phase transition between ordered and chaotic behavior in RBNs from the perspective of the *distributed computation* conducted by their nodes. We use a recently published framework to characterize the distributed computation in terms of its underlying *information dynamics*: information *storage*, information *transfer* and information *modification*. We find maximizations in information storage and coherent information transfer on either side of the critical point, allowing us to explain the phase transition in RBNs in terms of the intrinsic distributed computations they are undertaking.

Introduction

The information dynamics of distributed computation has recently emerged as an important tool for studying complex systems, e.g. information transfer in cellular automata (Lizier et al., 2008b, 2007). We believe that information dynamics are particularly relevant to networked systems: while network's structure has attracted much attention (Aldana, 2003), their time-series dynamics are "much less well understood" (Mitchell, 2006). Although the time-series dynamics of state-space trajectories and damage spreading are established, Mitchell (2006) suggests that "the main challenge is understanding the dynamics of the propagation of information ... in networks, and how these networks process such information."

Several studies have investigated the propagation and the processing of information in networks, in particular reporting phase transitions of these properties between ordered and chaotic regimes. Solé and Valverde (2001) investigated the effect of varying the message generation rate in a model of computer networks, finding phase transitions maximizing the number of packets actually delivered and the mutual information in the status of random node pairs. They infer that information transfer is maximized at the critical state. Kinouchi and Copelli (2006) investigated varying the "branching ratio" (effectively an activity level) in a network of excitable elements, finding phase transitions maximizing the dynamic range of the element's output, and inferring a maximization of information processing at criticality.

We are particularly interested in investigating the information dynamics of Random Boolean Networks (RBNs) (Kauffman, 1993), in part because of the power in their generality as discrete dynamical network models with a large sample space available. Also, they have a well-known phase transition from ordered to chaotic dynamics, in terms of length of transients in phase space with respect to average connectivity or activity level. We are also motivated by their popularity as models of Gene Regulatory Networks (GRNs). Perhaps most importantly, there have been several recent attempts to study the computational properties of RBNs (in particular information transfer). Here, Ribeiro et al. (2008) measure mutual information in the states of random node pairs as a function of connectivity in the network, and Rämö et al. (2007) measure the uncertainty (entropy) in the size of perturbation avalanches as a function of an order parameter. Both find maximization near the critical point, claiming that their results imply maximization of information propagation in this regime.

While these results are interesting, they do not directly measure the information dynamics claimed, e.g. none of the purported measures of information transfer properly measure directed, dynamic flows of information. Measures of model or task specific properties (by Solé and Valverde (2001), Kinouchi and Copelli (2006) and Rämö et al. (2007)) are qualitatively appealing but give no insights into the underlying quantitative nature of the information dynamics, while mutual information between random pairs of nodes (by Ribeiro et al. (2008) and Solé and Valverde (2001)) measures dynamic correlation across the collective which may result from an information transfer but is not a measure of it. (A more generic measure of "information transfer" in networks is presented in (Solé and Valverde, 2004), however it is a static measure of structure rather than a directed, dynamic flow of information.)

In this paper, we examine the information dynamics of RBNs from the perspective of the distributed computation undertaken by the nodes of the network in computing their

attractor. We apply a recently published framework to characterize the information dynamics of the distributed computation in terms of the elements of Turing universal computation: information storage, information transfer and information modification (Lizier et al., 2007). Our perspective of computation in RBNs is an important one, underlined by the comments of Mitchell (2006) on information dynamics in networks, and by the general importance attributed to information processing in biological systems (Polani et al., 2007; Gershenson, 2004a). Importantly, the perspective of distributed computation is unique in quantitatively aligning with our understanding of information storage, transfer and modification (Lizier et al., 2007).

We begin with overviews of RBNs and our framework for the information dynamics of distributed computation, and subsequently discuss how the framework will be applied to RBNs. We then present the results of this application, demonstrating that information storage and transfer are both maximized in the vicinity of the phase transition between ordered and chaotic dynamics. Importantly, we demonstrate a shift from the dynamics being dominated by information storage in the ordered regime, to a balance of information storage and transfer around the critical point, and a further shift to the dominance of information transfer (in particular higher order interactions) in the chaotic regime. Near the critical point we observe maximum capability for coherent computation, with relatively few but high-impact non-trivial information modification events. It is likely that these insights on the nature of computation in the vicinity of order-chaos phase transitions will be applicable to other complex systems.

Random Boolean Networks

Random Boolean Networks are a class of generic discrete dynamical network models. They are particularly important in artificial life, since they were proposed as models of gene regulatory networks by Kauffman (1993). See also Gershenson (2004a) for another thorough introduction to RBNs.

An RBN consists of N nodes in a directed *network* structure. The nodes take *boolean* state values, and update their state values in time as a function of the state values of the nodes from which it has incoming links. The network topology (i.e. the adjacency matrix) is determined at *random*, subject to whether the in-degree for each node is constant or stochastically determined given an average in-degree $\overline{K}$ (giving a Poissonian distribution). It is also possible to bias the network structure, e.g. toward scale-free degree distribution (Aldana, 2003). Given the topology, the deterministic boolean function or lookup table by which each node computes its next state from its neighbors is also decided at *random* for each node, subject to a probability p of producing "1" outputs (p close to 1 or 0 gives low activity, close to 0.5 gives high activity). The nodes here are heterogeneous agents: there is no spatial pattern to the network structure (indeed there is no inherent concept of locality), nor do the nodes have the same update functions. (Though, of course either of these can arise at random). Importantly, the network structure and update functions for each node are held static in time ("quenched"). In classical RBNs (CRBNs), the nodes all update their states synchronously.[1]

The synchronous nature of CRBNs, their boolean states and deterministic update functions give rise to a global state space for the network as a whole with deterministic transient trajectories ultimately leading to either fixed or periodic attractors in finite-sized networks (Wuensche, 1997). Effectively, the transient is the period in which the network is *computing* its steady state attractor.

RBNs are known to exhibit three distinct phases of dynamics, depending on their parameters: ordered, chaotic and critical. At relatively low connectivity (i.e. low degree K) or activity (i.e. p close to 0 or 1), the network is in an ordered phase, characterized by high stability of states and strong convergence of similar macro states in state space. Alternatively, at relatively high connectivity and activity, the network is in a chaotic phase, characterized by low stability of states and divergence of similar macro states. In the critical phase (the *edge of chaos* (Langton, 1990)), there is percolation in nodes remaining static or updating their values, and uncertainty in the convergence or divergence of similar macro states. This phase transition is typically quantified using a measure of sensitivity to initial conditions, or damage spreading. Following Gershenson (2004c), we take a random initial state A of the network, invert the value of a single node to produce state B, then run both A and B for many time steps (enough to reach an attractor is most appropriate). We then use the Hamming distance:

$$D(A,B) = \frac{1}{N}\sum_{i=1}^{N} |a_i - b_i|, \tag{1}$$

between A and B at their initial and final states to obtain a convergence/divergence parameter δ:

$$\delta = D(A,B)_{t\to\infty} - D(A,B)_{t=0}. \tag{2}$$

(Note $D(A,B)_{t=0} = 1/N$). Finding $\delta < 0$, implies the convergence of similar initial states, while $\delta > 0$ implies their divergence. For fixed p, the critical value of $\overline{K}$ between the ordered and chaotic phases is (Derrida and Pomeau, 1986):

$$K_c = \frac{1}{2p(1-p)}. \tag{3}$$

[1] There has been some debate about the best updating scheme to model GRNs (Darabos et al., 2007), and variations on the synchronous CRBN model are known to produce different behaviors. However, the relevant phase transitions are known to exist in all updating schemes, and their properties depend more on the network size than on the updating scheme (Gershenson, 2004b). As such, the use of CRBNs is justified for ensemble studies such as ours (Gershenson, 2004c).

For $p = 0.5$, we have $K_c = 2.0$. The standard deviation of δ peaks slightly inside the chaotic regime for finite-sized networks, indicating the widest diversity of networks for those parameters (Gershenson, 2004b).

Much has been speculated on the possibility that gene regulatory and other biological networks function in (or evolve to) the critical regime (see Gershenson (2004a)). It has been suggested that computation occurs more naturally with the balance of order and chaos there (Langton, 1990), possibly with information storage, propagation and processing capabilities maximized (Kauffman, 1993). Here we seek to improve on previous attempts to measure these computational properties, with a thorough quantitative study of the information dynamics in RBNs.

Information dynamics

Information theory (MacKay, 2003) is the natural domain to look for a framework to describe the information dynamics in complex systems, and indeed information theory is proving to be a useful framework for the analysis and design of complex systems, e.g. (Klyubin et al., 2004). The fundamental quantity is the (Shannon) *entropy*, which represents the uncertainty in a sample x of a random variable X: $H_X = -\sum_x p(x) \log_2 p(x)$ (all with units in bits). The *joint entropy* of two random variables *X* and *Y* is a generalization to quantify the uncertainty of their joint distribution: $H_{X,Y} = -\sum_{x,y} p(x,y) \log_2 p(x,y)$. The *conditional entropy* of *X* given *Y* is the average uncertainty that remains about *x* when *y* is known: $H_{X|Y} = -\sum_{x,y} p(x,y) \log_2 p(x|y)$. The *mutual information* between *X* and *Y* measures the average reduction in uncertainty about *x* that results from learning the value of *y*, or vice versa: $I_{X;Y} = H_X - H_{X|Y}$. The *conditional mutual information* between *X* and *Y* given *Z* is the mutual information between *X* and *Y* when *Z* is known: $I_{X;Y|Z} = H_{X|Z} - H_{X|Y,Z}$. Finally, the *entropy rate* is the limiting value of the entropy of the next state x of *X* conditioned on the previous $k-1$ states $x^{(k-1)}$ of *X*: $H_{\mu X} = \lim_{k\to\infty} H\left[x|x^{(k-1)}\right] = \lim_{k\to\infty} H_{\mu X}(k)$.

We have previously proposed a framework for the local information dynamics of distributed computation in (Lizier et al., 2007). The framework describes computation in terms of information storage, transfer and modification at each spatiotemporal point in a complex system.

The *information storage* of an agent in the system is the amount of information in its past that is relevant to predicting its future. The *excess entropy* is the total information stored by the agent (Feldman and Crutchfield, 2003), while the *active information storage* is the stored information that is currently in use in computing the next state of the agent (Lizier et al., 2007). We focus on the active information since it yields an immediate contrast in the relative contributions of storage and transfer to each computation. As shown in Fig. 1, the *local active information storage* for agent X is defined as the local (or unaveraged) mutual information

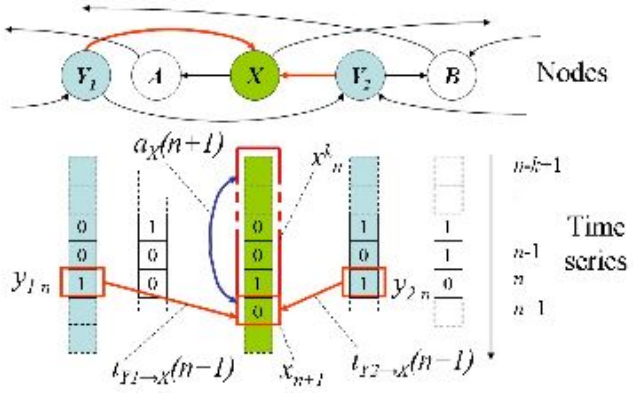


Figure 1: Information dynamics in a distributed network. For node X, this figure displays the local active information $a_X(n+1,k)$ and the local transfer entropies $t_{Y_1\to X}(n+1)$ and $t_{Y_2\to X}(n+1)$ from each of the causal information sources $V_X \in \{Y_1, Y_2\}$ at time $n+1$.

between its semi-infinite past $x_n^{(k)}$ (as $k \to \infty$) and its next state x_{n+1} at time step $n+1$:

$$a_X(n+1) = \lim_{k\to\infty} \log_2 \frac{p(x_n^{(k)}, x_{n+1})}{p(x_n^{(k)})p(x_{n+1})}, \qquad (4)$$

with $a_X(n,k)$ representing an approximation with finite history length k. The *active information* is the average over time (or equivalently weighted by the distribution of $(x_n^{(k)}, x_{n+1})$): $A_X(k) = \langle a_X(n,k)\rangle$. From our computational perspective, an agent can store information regardless of whether it is causally connected with itself; i.e. for RBNs, this means whether or not the node has a self-link. This is because information storage can be facilitated in a distributed fashion via one's neighbors, which amounts to the use of stigmergy (e.g. see Klyubin et al. (2004)) to communicate with oneself (Lizier et al., 2008a). Finally, the local entropy for any agent is the sum of the local active information and the local entropy rate $h_{\mu X}(n,k)$ (for any k):

$$h_X(n) = a_X(n,k) + h_{\mu X}(n,k), \qquad (5)$$

with their averages also related in this way. In a deterministic system, the entropy rate represents the joint contribution from the causal information sources to the destination (Lizier et al., 2008a), though it does not specify the information transfered from any particular one of those sources.

The *information transfer* between a source and a destination agent is defined as the information provided by the source about the destination's next state that was not contained in the past of the destination. The information transfer is formulated in the *transfer entropy*, introduced by Schreiber (2000) to address concerns that the mutual information (as a de facto measure of information transfer) was

a symmetric measure of statically shared information. The *local transfer entropy* (Lizier et al., 2008b) from a source agent Y to a destination agent X is the local mutual information between the previous state of the source[2] y_n and the next state of the destination x_{n+1}, *conditioned* on the semi-infinite past of the destination $x_n^{(k)}$ (as $k \to \infty$):

$$t_{Y \to X}(n+1) = \lim_{k \to \infty} \log_2 \frac{p(x_{n+1}|x_n^{(k)}, y_n)}{p(x_{n+1}|x_n^{(k)})}. \quad (6)$$

Again, $t_{Y \to X}(n,k)$ represents finite-k approximation, and the *transfer entropy* is the (time or distribution) average: $T_{Y \to X}(k) = \langle t_{Y \to X}(n,k) \rangle$. The local transfer entropy is shown in Fig. 1. The transfer entropy can also be formulated to condition on the states $v_{x,y,n}$ of all *causal information contributors* to the destination (the set V_X) except the source Y, so as to completely account for the contribution of Y. This formulation is known as the *complete* transfer entropy (Lizier et al., 2008b), with average and local values defined as:

$$T^c_{Y \to X}(n+1) = \langle t^c_{Y \to X}(n+1) \rangle , \quad (7)$$

$$t^c_{Y \to X}(n+1) = \lim_{k \to \infty} \log_2 \frac{p(x_{n+1}|x_n^{(k)}, y_n, v_{x,y,n})}{p(x_{n+1}|x_n^{(k)}, v_{x,y,n})}, \quad (8)$$

$$v_{x,y,n} = \{ z_n | \forall Z \in V, Z \neq Y \} . \quad (9)$$

The formulation in Eq. (6) is then labeled the *apparent* transfer entropy. Importantly, the transfer entropy properly measures a directed, dynamic flow of information, unlike mutual information measures used by Ribeiro et al. (2008) and Solé and Valverde (2001) which measure correlations only.

Information modification has been described as interactions between transmitted and/or stored information which result in a modification of one or the other (Langton, 1990). In (Lizier et al., 2007), we observed that *negative* values of $a_X(n)$ and $t_{Y \to X}(n)$ indicated misinformation or surprise regarding a given local outcome. We hypothesized that the sum of the local active information storage and apparent transfer entropy from each causal information contributor would be *negative* in a local information modification event, where no information source contained enough predictive power to overcome the misinformation generated by the other sources in the information "collision". This sum is known as the local *separable information*:

$$s_X(n) = a_X(n) + \sum_{Y \in V, Y \neq X} t_{Y \to X}(n). \quad (10)$$

Again, $s_X(n,k)$ represents finite-k approximation, and the *separable information* is the average $S_X(k) = \langle s_X(n,k) \rangle$.

[2]The transfer entropy can be formulated using the l previous states of the source. However, where only the previous state is a causal information contributor (as for RBNs), it is sensible to set $l = 1$ to measure direct transfer only at step n.

In Fig. 1, we have $s_X(n,k) = a_X(n,k) + t_{Y_1 \to X}(n,k) + t_{Y_2 \to X}(n,k)$. Positive local values of $s_X(n,k)$ indicate trivial information modification events, while negative local values of $s_X(n,k)$ indicate non-trivial information modifications events where the information sources interact in a non-trivial manner.

This framework was applied to cellular automata (CAs), which are effectively an ordered lattice-style sub-class of RBNs (Wuensche, 1997), in (Lizier et al., 2007). The framework quantified blinkers and regular domains as the dominant information storage elements, particles (gliders and domain walls) as the dominant information transfer agents, and particle collisions as the dominant (non-trivial) information modification events. These results align with existing conjecture on the nature of distributed computation in CAs, providing significant impetus for the use of this framework to analyze computation in other complex systems.

Information dynamics of RBNs

In this study, we seek to measure the *average* information dynamics of RBNs as a function of average in-degree or connectivity $\overline{K}$. For the RBNs simulated here, we use $N = 250$, Poissonian distributed in-degree for each node based on average in-degree $\overline{K}$, $p = 0.5$ (no bias in rules), and CRBNs with synchronous updating. Also, we do not bias the network structure, allowing comparison with the majority of existing RBN publications. The RBNs are modeled using enhancements to Gershenson's RBNLab software (http://rbn.sourceforge.net).

We measure the average entropy, entropy rate, and active information for each node in a given RBN (e.g. $A_X(k)$), then average these over each node in the RBN (to get e.g. $\langle A_X(k) \rangle$), then average these network averages over many networks generated for each $\overline{K}$ (at least 250) to determine the average values as a function of $\overline{K}$ (denoting this, e.g., as $A_X(k, \overline{K})$). Similarly, the average apparent and complete transfer entropies are measured for (at least 50) sample pairs of causally linked nodes (unlike the mutual information measurements by Ribeiro et al. (2008) and Solé and Valverde (2001) for *random* node pairs), averaged once to obtain network averages, and again over many networks to obtain averages as a function of $\overline{K}$.

While the *local* information dynamics are known to provide significantly greater insights into the distributed computation than their averaged counterparts (Lizier et al., 2007, 2008b), the averages will provide sufficient summaries regarding the ensemble properties with respect to $\overline{K}$. A hybrid approach is taken for the separable information; the average $S(k, \overline{K})$ is computed in a similar manner to the other metrics, however we also record the balance between its positive and negative local values (trivial and non-trivial information modifications respectively) $S^+_X(k, \overline{K})$ and $S^-_X(k, \overline{K})$ in contributing to the average. For a given node, we have for ex-

ample $S_X^+(k) = \left\langle s_X^+(n,k) \right\rangle$, where:

$$s_X^+(n,k) = \begin{cases} s_X(n,k) & \text{if } s_X(n,k) \geq 0 \\ 0 & \text{if } s_X(n,k) < 0 \end{cases} . \quad (11)$$

We seek to approximate an infinitely-sized network, and so avoid running the RBN for too many time steps because the computation is completed once the network reaches a periodic or fixed attractor (inevitable for finite-sized RBNs). For each simulation from an initial randomized state, we ignore a short initial transient of 30 steps to allow the network to settle into the main phase of the computation, then allow evolution over 400 time steps. Importantly, since the nodes in each RBN are heterogeneous agents, the probability distribution functions for each measure must be computed for each node individually rather than combining observations across all nodes (as could be done for the homogeneous agents in CAs (Lizier et al., 2007)). In order to properly sample the dynamics of each node in each RBN and generate enough data for the information theoretic calculations, many repeat runs from random initial states are required for each network (at least 4480 are used). For these calculations, one should use as large a history length k as facilitated by the number of observations (Lizier et al., 2008b); here we find $k \approx 13$ provides reasonable convergence for a reasonable number of repeat runs.

It has been hypothesized that RBNs close to the critical state possess a maximal information transfer capability (e.g. (Rämö et al., 2007)), which is generalized in the "edge of chaos" hypothesis (Langton, 1990): that systems exhibiting critical dynamics in the vicinity of a phase transition maximize their computational properties (see Kauffman (1993) regarding RBNs in particular). More specifically, Langton (1990) suggests that an intermediate level of information transfer and storage gives rise to complex computation in critical dynamics, with too much of either decaying the computational capability. This is at odds with suggestions of the maximization of information transfer in this regime, e.g. (Rämö et al., 2007; Solé and Valverde, 2001).

Our experiments aim to provide insight here. It is simple to foresee the average active information and apparent transfer entropy being zero in the extreme ordered regime (with fast freezing at point attractors) and in the extreme chaotic regime (where the high level of interactions overwhelm information storage and obscure the apparent contribution of each information source). It seems reasonable that both would be maximized, on average, in the interim near the critical region, where the dynamics support long correlations across space and time. On the other hand, we predict that the complete transfer entropy (which has been suggested to reveal higher information contributions as the level of interactions increases (Lizier et al., 2008a)) will continue to increase with the connectivity into the chaotic regime. Indeed, we observed that relatively high values of the apparent transfer entropy indicated the capacity for coherent local information transfer structures (i.e. gliders in CAs). We hypothesized there that an increasing of the complete transfer entropy in the chaotic regime indicated a higher level of interactions in conjunction with the loss of this coherence.

Important caveats are provided by criticisms of the edge of chaos hypothesis, e.g. see Mitchell et al. (1993). In examining *average* computational properties as a function of RBN parameters, we emphasize that there is in general a very large range of network realizations and consequently of behaviors possible for each parameter set. The local information dynamics of computation will provide much more detailed insights *for a given* RBN (as for CAs in (Lizier et al., 2008b)) than averages over nodes, networks and network sets discussed here. That being said, these averages can provide important insights into the computational properties as a function of RBN parameters, so long as we remember that the average results are akin to likelihoods rather than certainties, albeit likelihoods that are much stronger in the limit of infinite system size.

Results and discussion

Fig. 2 shows that the the average single node entropy $H_X(\overline{K})$ simply increases as a function of $\overline{K}$, as expected since the level of activity in the network is increasing with this parameter. More importantly, Fig. 2 also plots the average active information $A_X(k = 14, \overline{K})$ and entropy rate $H_{\mu X}(k = 14, \overline{K})$, showing that the active information rises then reaches a maximum near to the critical phase ($\overline{K} = 2$) before falling away, while the entropy rate only begins to rise near the critical phase then continues to rise and approach the entropy in the chaotic phase. Since the entropy is the sum of the active information and entropy rate (Eq. (5)), we can now begin to describe the phase transition in terms of computation: the ordered phase is dominated by information storage (information contained in the past of the node about its next state), the chaotic phase is dominated by information transfer (information from incoming links about the next state which was not contained in the node's past), while there appears to be something of a balance between the two near the critical phase.

We then examine the constituency of the information contributed from incoming links, the total of which is the entropy rate. Fig. 2 also plots the average *apparent* transfer entropy $T_{Y \to X}(k = 14, \overline{K})$ for each link, demonstrating that this quantity too rises to a maximum value close to the critical phase, then falls away. In contrast, Fig. 2 additionally plots the average *complete* transfer entropy $T^c_{Y \to X}(k = 13, \overline{K})$ for each link, which also begins to rise close to the critical phase but continues to increase into the chaotic phase. We see therefore that in the first stage of the shift toward the dominance of information transfer, the sources can be observed to have a significant influence on the destination (in the context of the destination's history) without considering the effect of the other causal sources (i.e.

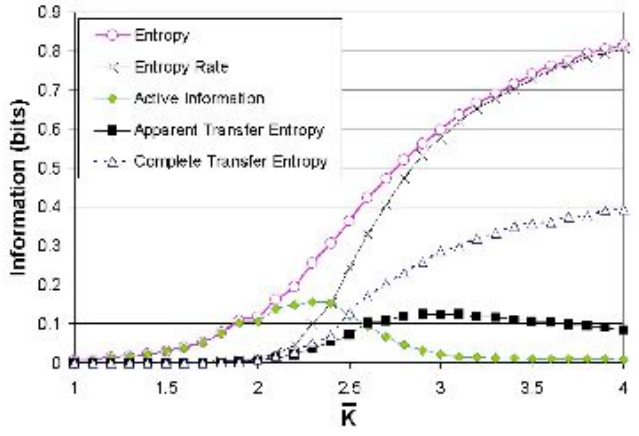


Figure 2: Average information dynamics versus average connectivity $\overline{K}$ for networks of size $N = 250$. Plotted here are the average entropy $H_X(\overline{K})$, entropy rate $H_{\mu X}(k = 14, \overline{K})$, active information $A_X(k = 14, \overline{K})$, apparent transfer entropy $T_{Y\to X}(k = 14, \overline{K})$ and complete transfer entropy $T^c_{Y\to X}(k = 13, \overline{K})$. The information required to predict the next state of each node is dominated by information storage at low $\overline{K}$ and by information transfer at higher $\overline{K}$ (first by coherent then interaction effects). Error bars (omitted) are on the scale of the data points for all plots.

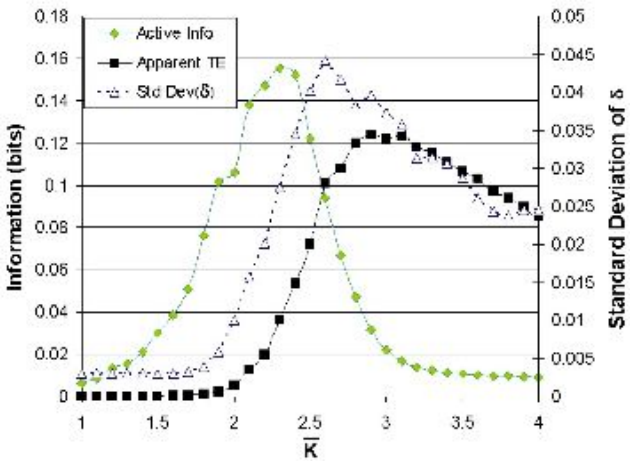


Figure 3: Maximizations in active information $A_X(k = 14, \overline{K})$ and apparent transfer entropy $T_{Y\to X}(k = 14, \overline{K})$ as a function of average connectivity $\overline{K}$ for $N = 250$, shown with respect to the standard deviation of the convergence/divergence parameter δ. This indicates that information storage peaks just on the ordered side of the phase transition, while (coherent) information transfer peaks just on the chaotic side of the phase transition.

$T_{Y\to X}(k = 14, \overline{K})$ is relatively high). In this regime, there is greater potential for *coherent* information transfer structures to propagate. However, as the activity level in the RBNs continues to rise with the average connectivity $\overline{K}$, the apparent effect of each source is swamped by the activity of the other causal sources, leading $T_{Y\to X}(k = 14, \overline{K})$ to fall away. Considering also the increase in $T^c_{Y\to X}(k = 13, \overline{K})$ (which does account for the other sources), we see that the level of interaction is increasing with the connectivity of the network. In the chaotic regime, the influence of any one information source can only be properly identified by taking all of the other sources into account also. These complimentary measures of information transfer provide different but useful insights, and give impetus to our hypothesis in (Lizier et al., 2008a) regarding the relative values of the apparent and complete components of information transfer in order-chaos phase transitions.

Next, we compare these maximizations to the phase transition as measured using the standard deviation of the convergence/divergence parameter δ (from Eq. (2)).[3] In Fig. 3 we see that the information storage peaks slightly within the ordered phase from the critical region, while the information transfer peaks slightly within the chaotic phase. Importantly, it is the apparent transfer entropy that peaks here (indicating the capability for coherent information transfer), as distinct from the complete transfer entropy which continues to increase into the chaotic phase. As per footnote 3, we expect the relative positions of these maximizations to be maintained around the critical phase as $N \to \infty$, with both likely to become closer to the critical point in this limit (as for the measure of correlation by Ribeiro et al. (2008)). The relative positions of the maximizations are quite interesting, because they align with existing conjecture on the nature of computation around phase transitions which typically associates information storage with the ordered phase and information transfer with the chaotic phase (e.g. (Langton, 1990)). Both the information storage and transfer appear to be driving the dynamics toward the critical phase, but from different sides of the phase transition.

We can also add quantitative evidence to the conflicting conjecture around whether information transfer is found at an intermediate (Langton, 1990) or maximum level (Solé and Valverde, 2001) at criticality. For RBNs, it is maximized close to criticality where one measures the apparent influence of a source in isolation, but equally it is at an intermediate level where the measurement considers the other

[3] δ was confirmed to change sign close to $\overline{K} = 2$ here (as per (Gershenson, 2004b)), with a subsequent slow increase after $\overline{K} = 2$ (known to be a finite-N effect). The standard deviation of δ is maximized during this increase in the chaotic regime (Gershenson, 2004b). Certain other measures suggested to indicate the critical phase are known to be shifted into the chaotic regime for finite-N, e.g. (Ribeiro et al., 2008). Given impetus as an indicator of the critical phase by the related measure of Rämö et al. (2007), we use the standard deviation of δ as guide to the relative regions of dynamics in finite-N networks.

causal information sources also. If these findings apply to such phase transitions in general, then both sources of conjecture appear to be well-founded, being resolved in these two different methods of measuring information transfer.

Indeed, we previously conjectured the capacity for the coherence of information transfer (provided by relatively large apparent transfer entropy) to be an important feature of complex dynamics in (Lizier et al., 2008a). Further insight into the coherent nature of the computation in the RBN is provided by the separable information $S_X(k, \overline{K})$. Fig. 4 shows that $S_X(k, \overline{K})$ is maximized for approximately the same values of $\overline{K}$ as the apparent transfer entropy (though it is slightly more spread out). This can be explained with reference to its positive and negative components, $S_X^+(k, \overline{K})$ and $S_X^-(k, \overline{K})$. We see from Fig. 4 that the early rise in the separable information is driven by $S_X^+(k, \overline{K})$ (trivial information modifications), with a peak occurring before $S_X^-(k, \overline{K})$ (non-trivial information modification events) rises and consequently reduce the total. As the connectivity $\overline{K}$ is further increased, $S_X^+(k, \overline{K})$ begins to fall whereas $S_X^-(k, \overline{K})$ continues to rise. Near the critical phase, at the peak of the separable information, note that there is in fact a relatively low incidence of non-trivial information modification events (i.e. $S_X^-(k, \overline{K})$ is low). This is interesting because of the importance placed on these events in computation, e.g. they are manifested as particle collisions in CAs. It appears that if the amount of non-trivial information modification events or information collisions is too large, the capacity of the system for complex computation is reduced. It is likely that this is due to a large amount of collisions eroding the coherent nature of the information storage and transfer within the system, disturbing the computation and reducing their own impact. A maximization of separable information, should perhaps be interpreted as maximizing the bandwidth for coherent information storage and modification, while allowing a smaller number of *high-impact* non-trivial information modification events in the coherent computation.

Finally, we note that all of the information dynamics described here experience maximum standard deviation in the vicinity of the critical region (not shown). This indicates maximal diversity in the information dynamics throughout the RBNs in this regime, as observed for other measures (e.g. (Gershenson, 2004b)).

Conclusion

We have described results which quantify the fundamental nature of computation around the critical phase in RBNs. The dynamics of RBNs are dominated by information storage in the ordered phase, with the level of information storage increasing with connectivity in the network. The increasing connectivity facilitates increasing activity, giving rise to an increasing level of information transfered from linked nodes. These two operations of universal computation appear to be in balance around the critical point. After

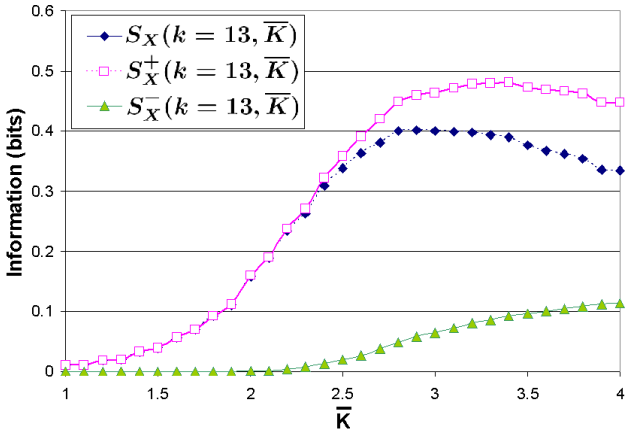


Figure 4: Separable information $S_X(k = 13, \overline{K})$ and its positive and negative components, $S_X^+(k = 13, \overline{K})$ and $S_X^-(k = 13, \overline{K})$ respectively, versus average connectivity $\overline{K}$ for $N = 250$. Trivial information modification (high $S_X^+(k)$) dominates the dynamics at low $\overline{K}$, while the amount of non-trivial information modification rises with $\overline{K}$.

this, information transfer continues to increase with connectivity, reducing the capacity for information storage. Near the critical point, there is a large amount of trivial information modifications, providing the capability for coherent information transfer and storage to flourish and indeed maximize, and allowing the small number of non-trivial information modifications to have a large impact on the coherent computation. As connectivity continues to increase, the information transferred from any single node observed in isolation initially appears strong, peaking slightly into the chaotic regime. With further increases however, the interaction between the nodes begins to dominate and erodes the capacity for coherent computation.

This new understanding of the information dynamics in RBNs near the critical phase is important, because there is evidence that the gene regulatory networks they model operate in this critical regime (Rämö et al., 2006). The implication here is that GRNs have evolved to a form facilitating maximum coherent computational capability. Furthermore, this study of RBNs represents the first exploration of an order-chaos phase transition using this framework for information dynamics: the results here are likely to be pertinent to order-chaos phase transitions in other systems.

We intend to continue our investigation of the information dynamics in RBNs, e.g. the effect of varying network size. Given the fundamental nature of the computational properties here, we expect to be able to describe the manner in which these information dynamics underpin other measures of the phase transition in RBNs, e.g. high interactivity (measured by complete transfer entropy and negative component of separable information) leads to large perturbation

avalanche sizes. We expect that the choice of RBN updating scheme will have little effect on the fundamentals of the phase transitions reported here, though this should be investigated. Furthermore, we intend to explore the effect of different topologies, in particular scale-free topologies (since most biological networks are scale-free with an exponent putting them near the critical point (Aldana, 2003)). Finally, we intend to investigate whether the information dynamics here can be used to drive evolution or self-tuning adaptation of RBNs to produce critical networks. Such an experiment could provide evidence that an underlying capacity for computation may have been a driver in GRN evolution.

References

Aldana, M. (2003). Boolean dynamics of networks with scale-free topology. *Physica D*, 185(1):45–66.

Darabos, C., Giacobini, M., and Tomassini, M. (2007). Semi-synchronous activation in scale-free boolean networks. In Almeida e Costa, F., Rocha, L. M., Costa, E., Harvey, I., and Coutinho, A., editors, *Proceedings of the 9th European Conference on Artificial Life (ECAL 2007), Lisbon, Portugal*, volume 4648 of *Lecture Notes in Artificial Intelligence*, pages 976–985, Berlin / Heidelberg. Springer.

Derrida, B. and Pomeau, Y. (1986). Random networks of automata: a simple annealed approximation. *Europhysics Letters*, 1(2):45–49.

Feldman, D. P. and Crutchfield, J. P. (2003). Structural information in two-dimensional patterns: Entropy convergence and excess entropy. *Physical Review E*, 67(5):051104.

Gershenson, C. (2004a). Introduction to random boolean networks. In Bedau, M., Husbands, P., Hutton, T., Kumar, S., and Suzuki, H., editors, *Proceedings of the Workshops and Tutorials of the Ninth International Conference on the Simulation and Synthesis of Living Systems (ALife IX), Boston, USA*, pages 160–173.

Gershenson, C. (2004b). Phase transitions in random boolean networks with different updating schemes. arXiv:nlin/0311008.

Gershenson, C. (2004c). Updating schemes in random boolean networks: Do they really matter? In Pollack, J., Bedau, M., Husbands, P., Ikegami, T., and Watson, R. A., editors, *Proceedings of the Ninth International Conference on the Simulation and Synthesis of Living Systems (ALife IX), Boston, USA*, pages 238–243, Cambridge, USA. MIT Press.

Kauffman, S. A. (1993). *The Origins of Order: Self-Organization and Selection in Evolution*. Oxford University Press, New York.

Kinouchi, O. and Copelli, M. (2006). Optimal dynamical range of excitable networks at criticality. *Nature Physics*, 2(5):348–351.

Klyubin, A. S., Polani, D., and Nehaniv, C. L. (2004). Tracking information flow through the environment: Simple cases of stigmergy. In Pollack, J., Bedau, M., Husbands, P., Ikegami, T., and Watson, R. A., editors, *Proceedings of the Ninth International Conference on the Simulation and Synthesis of Living Systems (ALife IX), Boston, USA*, pages 563–568. MIT Press.

Langton, C. G. (1990). Computation at the edge of chaos: phase transitions and emergent computation. *Physica D*, 42(1-3):12–37.

Lizier, J. T., Prokopenko, M., and Zomaya, A. Y. (2007). Detecting non-trivial computation in complex dynamics. In Almeida e Costa, F., Rocha, L. M., Costa, E., Harvey, I., and Coutinho, A., editors, *Proceedings of the 9th European Conference on Artificial Life (ECAL 2007), Lisbon, Portugal*, volume 4648 of *Lecture Notes in Artificial Intelligence*, pages 895–904, Berlin / Heidelberg. Springer.

Lizier, J. T., Prokopenko, M., and Zomaya, A. Y. (2008a). A framework for the local information dynamics of distributed computation in complex systems. Submitted to Physica D.

Lizier, J. T., Prokopenko, M., and Zomaya, A. Y. (2008b). Local information transfer as a spatiotemporal filter for complex systems. *Physical Review E*, 77(2):026110.

MacKay, D. J. (2003). *Information Theory, Inference, and Learning Algorithms*. Cambridge University Press, Cambridge.

Mitchell, M. (2006). Complex systems: Network thinking. *Artificial Intelligence*, 170(18):1194–1212.

Mitchell, M., Hraber, P. T., and Crutchfield, J. P. (1993). Revisiting the edge of chaos:evolving cellular automata to perform computations. *Complex Systems*, 7:89–130.

Polani, D., Sporns, O., and Lungarella, M. (2007). How information and embodiment shape intelligent information processing. In Lungarella, M., Iida, F., Bongard, J., and Pfeifer, R., editors, *Proceedings of the 50th Anniversary Summit of Artificial Intelligence, New York*, volume 4850 of *Lecture Notes in Computer Science*, pages 99–111, Berlin / Heidelberg. Springer.

Rämö, P., Kauffman, S., Kesseli, J., and Yli-Harja, O. (2007). Measures for information propagation in boolean networks. *Physica D*, 227(1):100–104.

Rämö, P., Kesseli, J., and Yli-Harja, O. (2006). Perturbation avalanches and criticality in gene regulatory networks. *Journal of Theoretical Biology*, 242(1):164–170.

Ribeiro, A. S., Kauffman, S. A., Lloyd-Price, J., Samuelsson, B., and Socolar, J. E. S. (2008). Mutual information in random boolean models of regulatory networks. *Physical Review E*, 77(1):011901–10.

Schreiber, T. (2000). Measuring information transfer. *Physical Review Letters*, 85(2):461–464.

Solé, R. V. and Valverde, S. (2001). Information transfer and phase transitions in a model of internet traffic. *Physica A*, 289(3-4):595–605.

Solé, R. V. and Valverde, S. (2004). Information theory of complex networks: Onevolution and architectural constraints. In Ben-Naim, E., Frauenfelder, H., and Toroczkai, Z., editors, *Complex Networks*, volume 650 of *Lecture Notes in Physics*, pages 189–207. Springer, Berlin / Heidelberg.

Wuensche, A. (1997). *Attractor Basins of Discrete Networks*. PhD thesis, The University of Sussex.

‘Psychoanalysis’ of a Minimal Agent

Santosh Manicka and Inman Harvey

Center for Computational Neurosciences and Robotics,
School of Cognitive and Computing Sciences
University of Sussex, Brighton BN1 9QH, United Kingdom
{S.Manicka, inmanh}@sussex.ac.uk

Abstract

The Secretary problem is studied with minimal cognitive agents, being a problem that needs memory and judgment. A sequence of values, drawn from an unknown range, is presented; the agent has only one chance to pick a single value as they are presented, and should try to maximize the value chosen. In extension of previous work (Tuci et al. 2002), Continuous Time Recurrent Neural Networks (CTRNN) are evolved to solve the problem, and then their strategies are analyzed by relating mechanisms to behavior. Strategies similar to the known optimal strategy are observed, and it is noted that significantly different strategies can be generated by very different mechanisms that perform equally well.

Introduction

This study is in the tradition of using Evolutionary Robotics techniques (Cliff et al. 1993; Harvey et al. 2005) to evolve artificial minimal agents with a genetically specified ‘nervous system’ so as to perform tasks of interest (Beer, 1996; Beer, 2003; Goldenberg et al. 2004). The interest of the Secretary Problem (described in the next section) is that it requires memory and judgment, and the provably optimal strategy requires a strategy of some sophistication.

Tuci et al (2002) evolved CTRNNs with 4 nodes to perform well at this task, and performed a preliminary analysis of the strategies seen. Here we largely replicate their methodology, and go on to look at their activation level patterns from their performance in various scenarios and then interpret them in terms of their behavior with the objective of uncovering any underlying strategy. Whereas Tuci et al had done an overall performance analysis, our strategy was to observe the behavior of the evolved mechanisms in the smallest units of the problem and do it over a strategically chosen range of problems so that we would be able to sensibly describe the observed behavior as a strategy. To put it simply, we analyze the strategies by relating the neural mechanisms to the behavior, in what could be metaphorically called a form of ‘psychoanalysis’. The significant results are:

1. A network is found to have evolved a strategy similar to the actual optimal strategy.
2. Two networks with nearly equal fitness values are found to have evolved significantly different strategies.

The Secretary problem can be considered as one in a larger class of problems in probabilistic decision making using a single criterion (maximize rank). While it has been shown, through this work, that evolution ‘thinks’ like a mathematician in a simple problem of a larger class of decision making problems, it could be interesting to investigate its influence in more complicated problems. One such interesting problem could be the game of poker. CTRNNs could be evolved and their strategies be compared with the game theoretic strategies of poker in search of interesting implications from a cognitive point of view. An even more complicated application could be problems involving multiple criteria like the ‘Experts case’ problem (Czogala and Roubens, 1989). If successful CTRNNs could be evolved in such problems, a behavioral analysis of their strategies as adopted in this work might help reveal interesting cognitive insights as this approach is fundamentally different from the usual analytical approaches.

Background

The Secretary problem is a problem of choice from among a temporal sequence of random possibilities so that the expected payoff from the choice is maximized or the expected cost of the choice is minimized. A very simple form of the secretary problem version has been described by Ferguson (1989) as follows:

1. There is one secretarial position to be filled
2. The total number of applicants is known
3. The applicants are interviewed sequentially in random order. An order has the same chance of occurrence as any other order.
4. An applicant should either be accepted or rejected at the end of the interview of the applicant and the decision should be made solely on the relative rank of the applicant
5. An applicant once rejected cannot later be accepted
6. The interviewer will not be satisfied unless the chosen applicant is the best in the group (i.e., the payoff is either 1 or 0)
7. If no applicant is accepted before the last applicant, the last applicant should be accepted.

The solution to the problem is quite simple: for a specific integer $r \geq 1$, reject the first r-1 applicants and choose the next applicant who is the best among all the applicants seen until then. Mathematically stated, the probability of choosing the best applicant is $1/n$ if $r=1$; if $r>1$ then (Ferguson, 1989)

$$\Phi_n(r) = \sum_{j=r}^{n} P \quad (j^{\text{th}} \text{ applicant is best and you select it})$$

$$= \sum_{j=r}^{n} \left(\frac{1}{n}\right)\left(\frac{r-1}{j-1}\right) = \left(\frac{r-1}{n}\right)\sum_{j=r}^{n} \frac{1}{j-1}$$

For a very large value of n, the value of r is calculated as $1/e$. This translates as "Reject the first ~37% of the interviewees and then pick the first best". Implementation of this analytic solution is quite straight forward; it hardly takes a few lines of code in a computer program. Given the proof arrived at by mathematicians, an interesting question, from an evolutionary point of view, would be to ask "How would evolution shape the cognition of an agent powered by a dynamical system to solve this problem?" This would be interesting in the sense that the agent here is not expected to 'know' mathematics. An experiment performed with such an agent could lead to insights into the mechanisms of cognitive behavior. Such experiments have been conducted in the past by evolving continuous time recurrent neural networks (CTRNNs) (Beer, 1996). Tuci et al (2002) have successfully evolved a CTRNN that could solve the Secretary problem (that maximizes the expected payoff rather than look for the single best item). In this paper we go further to analyze in depth successful CTRNNs.

Methods

As the emphasis of this work is analysis, we used a proven method of evolution of a CTRNN to solve the Secretary problem; we used a very similar approach as used by Tuci et al (2002). In the experimental set up by Tuci et al, the logical sequence of values (worthiness of the interviewee) is presented in the form of a temporal sequence of inputs to the CTRNN; for each one the binary input is switched to value 1 for a length of time proportional to the current value (referred to as 'exposure time'), and then cleared to zero. A sequence (also referred to as a 'trial') contains 20 unique items (integers), from which the network is expected to choose one. The single thresholded output is then tested for a binary accept/reject decision, before moving on to the next value (item) in the sequence. Overall, the network is expected to maximize its payoff by choosing an item with a relative rank in the sequence as large as possible. The presentation of a trial is terminated as soon as the network accepts an item or after all the 20 items are presented (in which case the last item is considered accepted). Between consecutive presentations of items, the network is cleared by setting the input to zero for 2 (simulated) seconds. Before the start of each trial, the network is reset by setting the output values of the nodes to zero. The input is always fed to node 1 and is time-based (Tuci et al. 2002) i.e., the external input will remain at the value '1' for a particular number of iterations (value of an item in the trial/time-step size). In our experiment, the time-step size value was 0.2. So, if an item's value is 27, the external input will be '1' for 135 iterations of network-update.

The network is run through a set of 60 trials, each of length 20 during its evolution. Each trial is defined as follows (Tuci et al. 2002)

$$c = 1,\ldots..,60 \quad \left\{\begin{array}{l} l_c = c; \\ h_c = l_c + 29 \end{array}\right\}$$

Where, l_c is the lowest possible value of an item of trial c and h_c is the highest such possible value.
Each neuron of the 4-neuron CTRNN we evolved uses the following state equation (Tuci et al. 2002):

$$\tau_i \dot{y}_i = -y_i + \sum_{j=1}^{k} w_{ji} z_j + g I_i$$

with

$$z_j = \frac{1}{1+\exp\left[-\left(y_j + \beta_j\right)\right]}, i = 1,...,4$$

y_i = cell potential
τ_i = decay constant
w_{ji} = strength of connection from neuron j to i
z_j = firing rate of neuron j
I_j = external input to neuron i
g = sensory gain factor
β_j = bias

When the activation of the output neuron exceeds 0.5 at the end of presentation of an item, the item is considered chosen.
The initial strength of the population, the fitness function and the evolutionary parameters that we used were the same as what were used in (Tuci et al. 2002) except for a few changes (as we could not replicate the experiment with the original parameters): the decay constants were mapped to $[10^0, 10^{1.8}]$ instead of $[10^0, 10^{2.8}]$. We used mutation probability 0.2 instead of 0.3 with explicit elitism. The cut-off values for the elitism varied between top 5% and 8%. The number of generations was varied between 5000 and 25000 in the runs.
We evolved 2 fairly-well performing CTRNNs using the above mentioned parameters from 7 evolutionary runs. We consider a network to perform fairly-well when its fitness value is comparable to that of the best network evolved by Tuci et al i.e., a fitness value of 0.85. Henceforth, we will refer to these networks as N1 and N2. Below, we describe their morphologies and their various performance measures.

Results of evolution

Morphology and performance of network N1

We have presented the morphologies of the evolved networks here so that any experiment with these networks can be

replicated with ease without having to resort to re-evolution. Otherwise, we have not explored any direct influence of the network parameters on the results of our analyses.
The mean maximized rank choice (on a scale of 0 to 1) of N1 is: 0.79

Morphology:

Weight matrix (connection from node *j* to node *i*)

	i=1	i=2	i=3	i=4
j=1	-2.853552	-2.087797	0.612068	2.975119
j=2	-0.585138	3.029250	-1.806782	0.330331
j=3	-3.342187	1.828053	1.184740	-4.909308
j=4	0.125955	0.641515	-1.510127	-1.685360

Table 1a. Evolved Weight matrix for N1

Other parameters of the i^{th} neuron

i	1	2	3	4
τ_i	3.563501	46.563980	0.327475	39.911218
βi	0.898813	-0.569968	-0.122038	-0.734500

Table 1b. Other evolved parameters for N1

Gain = 3.352800

The percentage of actively expressed preference (choice of any item other than the last item) and the average acceptance position per trial are plotted below in figures 1a and 2a after performing 100 simulations of 60 trials each.

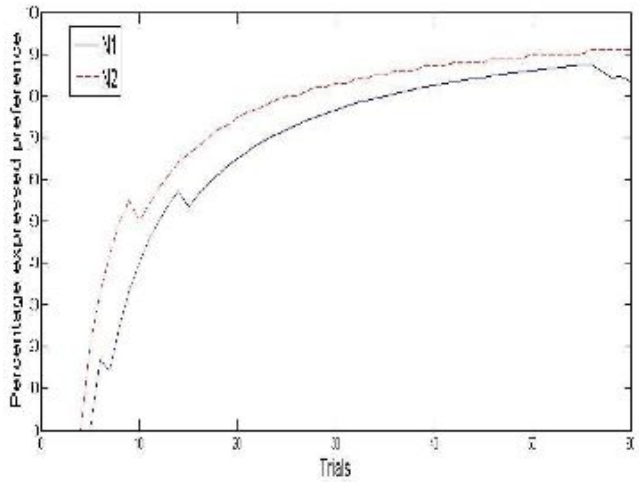


Fig 1a. Percentage of expressed preference per trial

The value 'percentage of expressed preference' at trial *i* denotes the percentage of the number of active choices made by the network until that trial. Acceptance position is the position in the sequence of a trial where an item is accepted (actively or passively) by the network.

Morphology and performance of network N2

The mean maximized rank choice (on a scale of 0 to 1) of N2 is: 0.75

Morphology:

Weight matrix (connection from node *j* to node *i*)

	i=1	i=2	i=3	i=4
j=1	-0.744026	2.395746	-0.296505	2.120096
j=2	0.818743	-4.829388	-3.410042	-3.145099
j=3	-0.344417	1.448848	4.833700	4.198339
j=4	-4.913282	1.830852	-0.096895	1.296698

Table 2a. Evolved Weight matrix for N2

Other parameters of the i^{th} neuron

i	1	2	3	4
τ_i	3.939339	358.841803	17.613176	22.553297
βi	-1.336118	0.079064	-1.928700	-1.408360

Table 2b. Other evolved parameters for N2

Gain = 4.167492

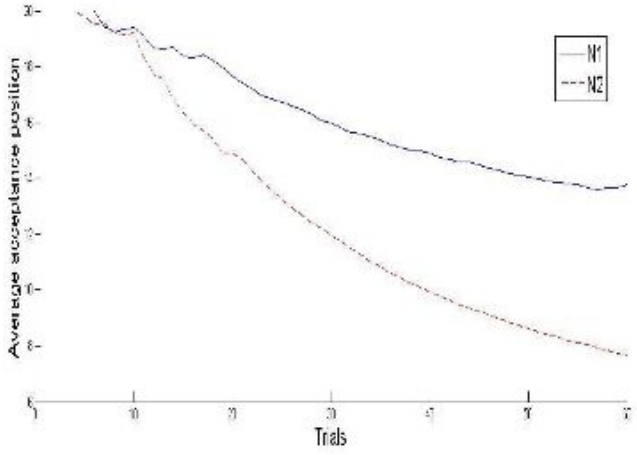


Fig 2a. Average acceptance position per trial

Figures 3a and 3b depict the performance in terms of average relative rank of the item chosen by the networks N1 and N2 in each of the 60 trials averaged over 100 simulations.
The horizontal dotted lines indicate the overall mean performance of the network.

Analysis

It can be seen from figures 3a and 3b that both N1 and N2 perform relatively worse towards the ends of the trial spectrum. We will now look at the activation level patterns of

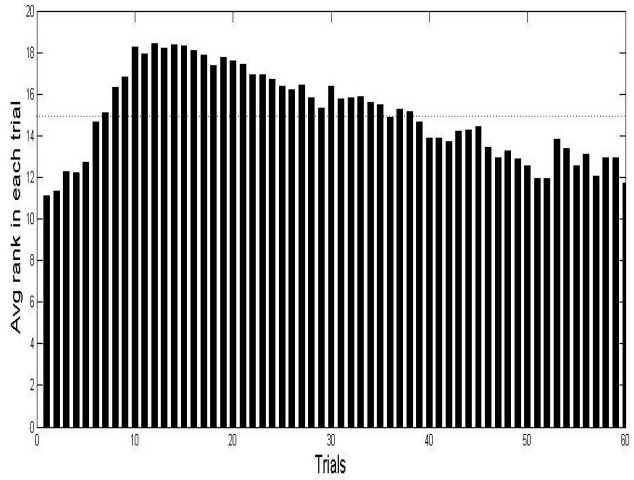


Fig 3a. Average rank choice per trial in N1

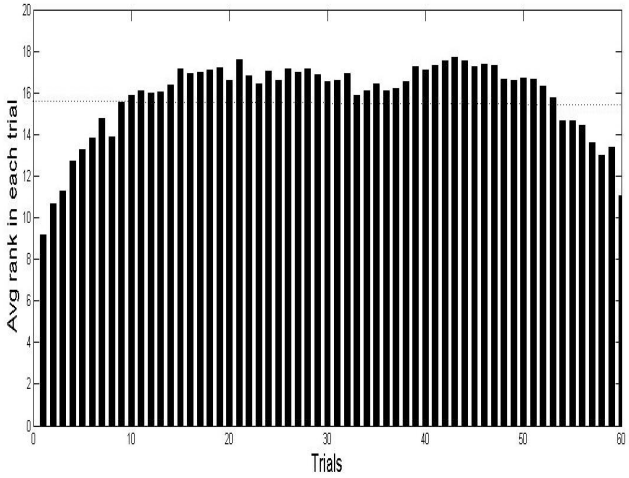


Fig 3b. Average rank choice per trial in N2

each network in specific problem scenarios in an attempt to look at the networks' behavior in detail. A test trial is presented from each of the following categories:

1. A randomly generated trial from the lower trials (c is between 1 and 5) where the network performs the worst
2. A randomly generated trial from the intermediate trials where the network performs the best
3. A randomly generated trial from the higher trials (c is between 55 and 60) where the network performs the worst

Figures 4a through 4d depict the activation level values of the output neuron during the exposure time (see 'Methods' section for definition) for each test trial.

Behavior of N1

Test trial 1

27, 8, 29, 5, 18, 20, 15, 24, 17, 16, 9, 3, 2, 19, 22, 4, 26, 23, 21, 12

Result: No active choice. Figure 4a shows the behavior of the network towards each item in the trial.

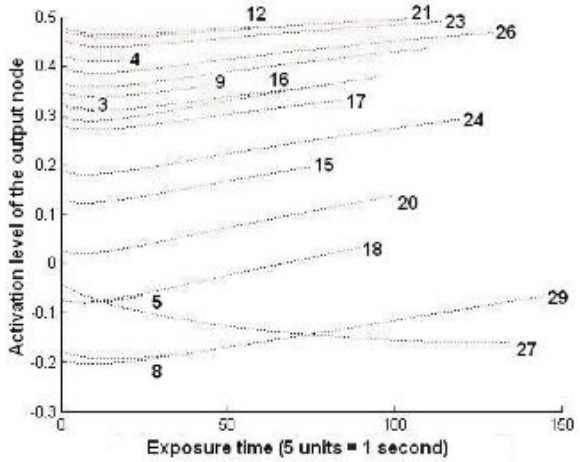


Fig 4a. Activation levels of the output node of N1 for trial 1

Test trial 2

22, 24, 31, 35, 27, 32, 33, 23, 39, 45, 36, 43, 25, 46, 40, 49, 29, 21, 26, 48

Result: Item = 46; Position = 14; Relative rank = 18. Figure 4b shows the behavior of the network towards each item in the trial.

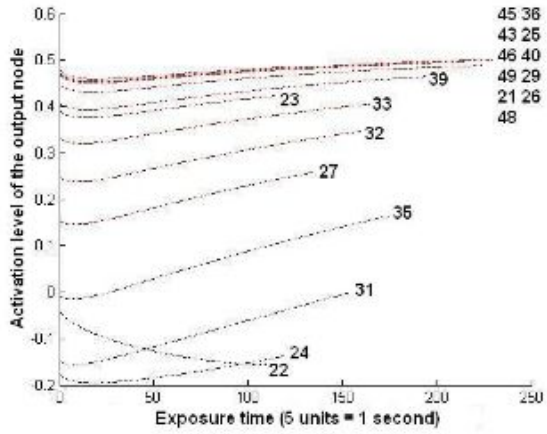


Fig 4b. Activation levels of the output node of N1 for trial 2

Test trial 3

86, 70, 73, 84, 65, 68, 60, 66, 82, 74, 83, 78, 71, 81, 61, 63, 79, 72, 76, 77

Result: Item = 82; Position = 9; Relative rank = 17. Figure 4c shows the behavior of the network towards each item in the trial.

Observations

From figures 4a, 4b and 4c, it can be seen that:

1. The network starts responding with a 'great dip' in all the trials for the first item, regardless of the absolute value of that item. The dip can continue for about 150 iterations (fig 4c) and then start rising. Its maximum destination value of activation could be about 0.05 (see fig 4c, first item = 86; max value could be 89).

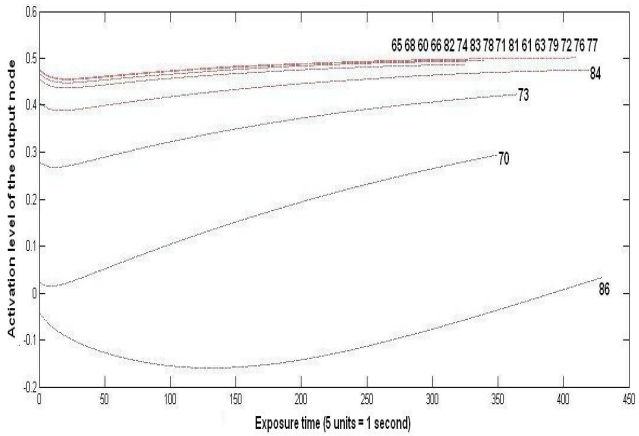


Fig 4c. Activation levels of the output node of N1 for trial 3

2. The number of items that spend their 'lives' below zero-AL ('AL' is short for Activation Level) decreases from test trial 1 to test trial 3.
3. In all trials, the vertical separation between response curves grows smaller with more items seen. They are the closest between activation levels 0.4 and 0.5.
4. In all the trials, for items following the first, there is a slight initial dip and it stays slightly longer for items at higher positions in the trial. Besides, this initial dip is longer for items in trial 3 than in the other two trials.
5. In all the trials, for each item, the network responds with a shoot-up after the initial dip. The slope of the shoot-up is almost the same for all the items in test trial 1 and also test trial 2 (except towards the last few items of the trial). However, this parallelism is less pronounced in test trial 3 where the network tends to flatten-out its response to the items it sees in the higher positions of the trial.

Interpretations and reasoning

The great dip is indicative of 'Never choose the first item, whatever it may be' as even if it is 89, it wouldn't be chosen (test trial 3). The extent of the dip will determine how fast the responses of the items can race to the 0.5-AL finishing line. This is the reason behind observation (2). As the network sees more items, it strives to settle for an item with as big a relative rank as possible by being more "cautious". This behavior can be seen in observation (4). The longer initial dip and therefore a more delayed start of the shoot-up, combined with a slightly smaller rate of shoot-up makes sure that only those items with relatively more persistence (longer response due to larger items) can move closer to the finishing line. Observation (4) is in turn the reason behind observation (3) since (4) results in a smaller difference between the initial and the final activation values for an item seen higher in the trial and therefore the following item's starting activation level is closer to that of the previous item. The reason behind observation (5) is that when it sees more values in a higher trial and still strives for rank maximization, it can't continue the trend of shoot-up (with a constant rate) as it does in a lower trial because if it does, items with relatively smaller values (like 65 or 70 in trial 3) can easily cross the 0.5-AL. Therefore, it seems that it has evolved to "stretch" its single strategy of rank maximization to the higher trials by lengthening its initial dip and slowing down its shoot-up. This also could be the reason why the last range of values (60, 89) is a bad performer (fig 3a) as the percentage of expressed preference slightly drops towards the end (fig 1a). That's because the response could flatten out so much that the network eventually refuses to actively accept any item as shown in the response levels in fig 4d below for the following trial:

63, 60, 76, 65, 71, 68, 77, 85, 75, 84, 62, 74, 86, 82, 64, 78, 88, 72, 73, 61

Result: No active choice

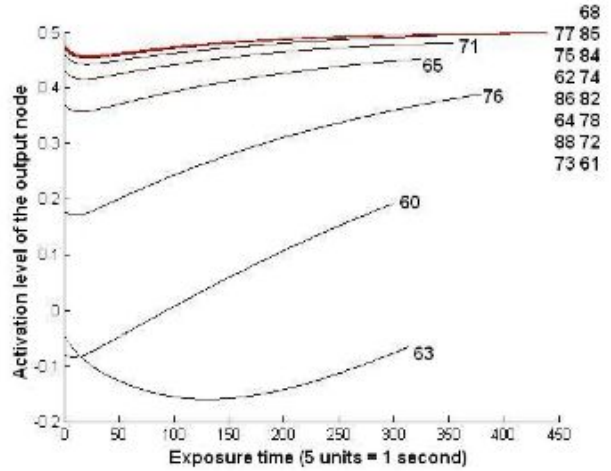


Fig 4d. Activation levels of the output node of N1 for a random trial

Strategy

Has the network evolved a strategy? We have not found a complete answer to this question, but we will present here a few directions along which the answer could be pursued.

The network appears to be following two stages of aspiration-setting: first, when it sees the first item and second, when it sees the rest as the response to the first item is significantly different from the rest. The term "aspiration" can be described as a value such that an item with a value greater than the aspiration could be considered as a candidate for selection. Here we describe a possible approach to uncover the aspiration-setting strategy of the network. See figure 5a. A sample response curve is plotted for a random item at a random position in a trial. Two types of typical responses are depicted past a random point B – a response crossing the 0.5-AL and a response flattening out just below the 0.5-AL line. When the response reaches point C (on the same level as the starting point A), the current item is roughly considered worthy of acceptance. The corresponding value 'p' on the x-axis when multiplied by 0.2 (step size) can be considered as

the lower bound of the current aspiration (LBA) set by the network. The actual aspiration can be calculated when the response crosses the 0.5-AL as at point D. Then the actual aspiration is *l**0.2. Yet there is a possibility that for an item, the response totally flattens out before it could cross the finishing line (fig 4d). In that case, an approximate value of the more accurate LBA can be calculated. At point E, in fig 5a, the response starts to flatten out.

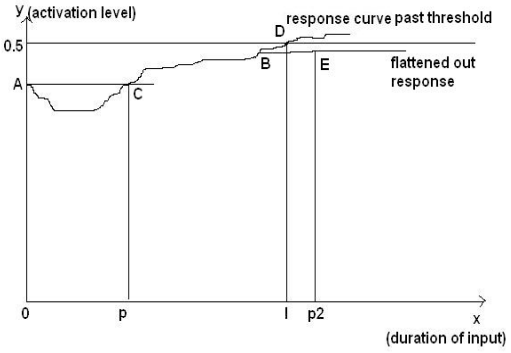


Fig 5a. Aspiration-setting in network N1

The LBA is then calculated as (*p2**0.2). If the item is not big enough (not a long enough response) to help reckon the aspiration, it can be incremented by 1 until it either reaches point D or E. This way, the aspiration at a particular position in a particular trial can be calculated. Such calculations when performed extensively over a wide range of trials and positions might help get a deeper insight into the underlying patterns of aspiration-setting. Further analysis of the great dip's impact also helped reveal a more interesting strategy. We ran about 1000 simulations of the network each containing the regular 60 trials. Fig 6a below depicts the percentage of choice made at a particular position of the trial in these simulations.

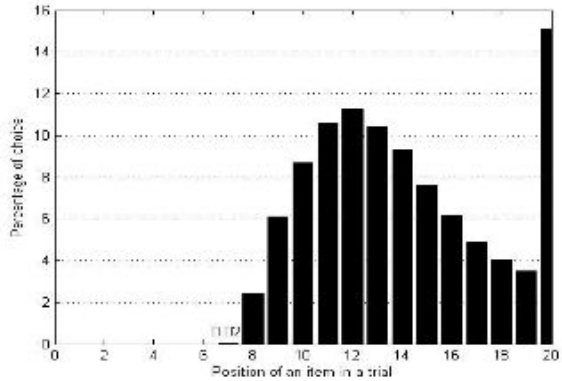


Fig 6a. Percentage of choice per position of N1

It can be seen that no choice is made at any of the first 6 positions. At the 7th position, the percentage of choice is extremely low (0.02%). Only starting from the 8th position, the network actively makes a good proportion of choices. The actual optimal strategy to the Secretary problem is to ignore (make no choice) the first 37% of the trial (Ferguson, 1989) and then to choose the best item that's better than any item seen so far. The length of a trial in our experiment is 20 and 37% of 20 is 7.4. Therefore, our network seems to have evolved a similar strategy as the actual optimal strategy at least as far as ignoring the first few items is concerned.

Comparison with optimal strategy

In this section, we compare the behavior of N1 with the analytic-optimal strategy described in the 'Background' section. As a note, the mean maximized rank choice of the optimal strategy is 0.80 as compared to 0.79 of N1. Fig 6b below depicts the trial-wise average acceptance position of N1 against that of the optimal strategy. It can be seen that the average acceptance of the optimal strategy is almost always about 14. It is because the first best item better than the best in the first 37% of the sequence (i.e., in the first 7) can appear anywhere between positions 8 and 20 with equal probability. Therefore, over a sufficiently large number of simulations (in this case, 1000) the average position will be the average of 8 and 20 which is 14. Consequently, it can be seen that the strategy of N1 is not wholly similar to the optimal strategy after ignoring the first 37% of the sequence. Still there is an overlap between the 2 plots between trials 23 and 27 and also at about 56. Could it mean that N1 behaves in the same way as the optimal strategy in these ranges of values? Further analysis provides the answer 'No'. Figure 6c depicts a trial-wise average rank choice by N1 (fig 3a repeated) against the choice by the optimal strategy. It can be seen that between trials 23 and 27, N1 fairs better than the optimal strategy. At the trials around 56, the optimal strategy performs better than N1. So their performances are different even though their average acceptance positions in these trials are the same.

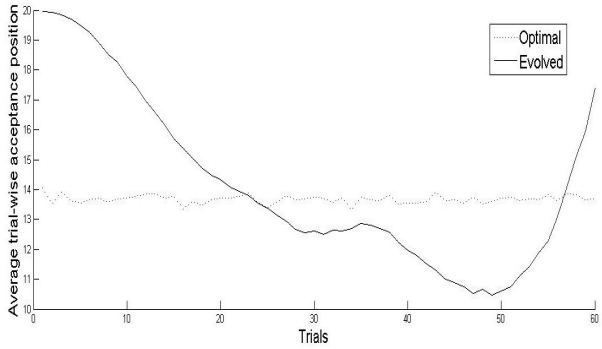


Fig 6b. Average acceptance positions – a comparison

It can also be seen from this figure (6c) and from figure 6b that even if N1's performance is the same as the optimal strategy in trials like 10 and 50, their average acceptance positions are different. The reason for the above 2 observations is that even though the average acceptance position is the same, the standard deviation between the acceptance positions of N1 and the optimal strategy is quite considerable as shown in figure 6d.

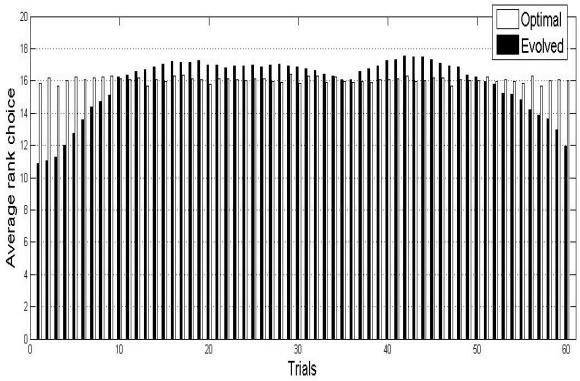


Fig 6c. Average trial wise rank choice – a comparison

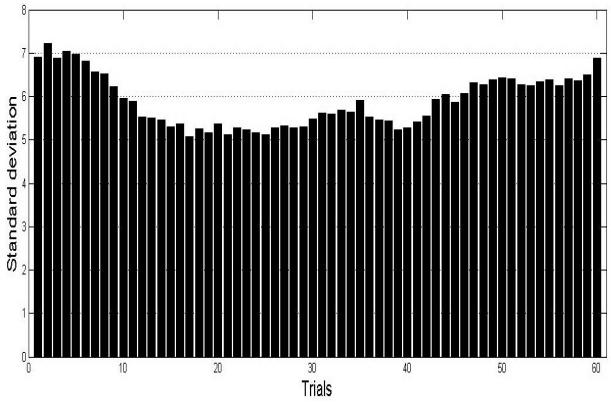


Fig 6d. Standard trial-wise deviation between average acceptance position of N1 and that of optimal strategy.

It can be seen that the minimum standard deviation is about 5. Also, in trial 42 where N1 seems to perform relatively the best (fig 6c, 3a), the standard deviation is approximately 5.5. At trial 17, where the standard deviation is the least i.e., 5 the performance is also among the highest. These observations suggest that the second part of N1's strategy is not as definitive and general as the optimal strategy (standard deviation is neither zero nor constant in the trials; see fig 6d) and yet not fully trial-dependent (there is no pattern displayed in the evolved strategy in fig 6b).

Behavior of N2

In this section, we describe the behavior of the network N2 when it is presented with the same 3 trials as N1 was presented with and compare their responses. The focus here is to point some significant differences between the networks' cognition even if there is no big difference between their overall performances.

Test Trial 1
27, 8, 29, 5, 18, 20, 15, 24, 17, 16, 9, 3, 2, 19, 22, 4, 26, 23, 21, 12
Result: No active choice. See figure 7a.

Test Trial 2
22, 24, 31, 35, 27, 32, 33, 23, 39, 45, 36, 43, 25, 46, 40, 49, 29, 21, 26, 48
Result: Item = 39; Position = 9; Relative rank = 13. See figure 7b.

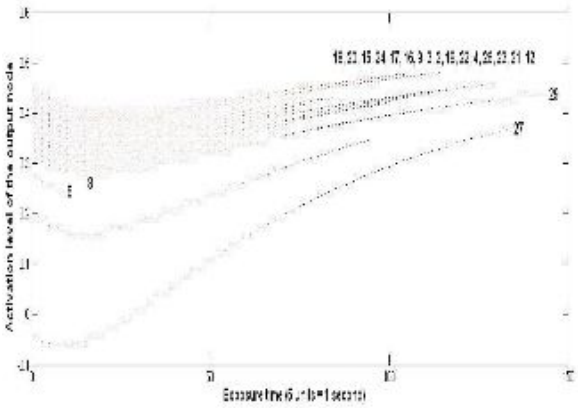


Fig 7a. Activation levels of the output node of N2 for trial 1

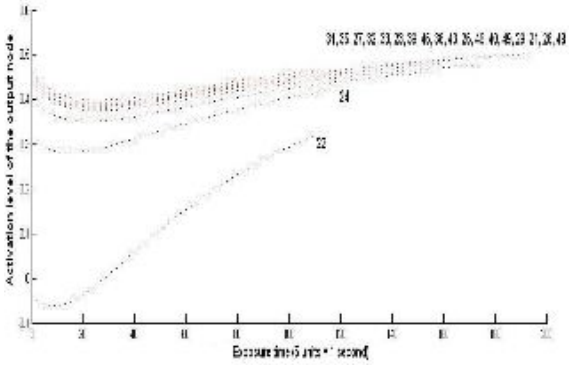


Fig 7b. Activation levels of the output node of N2 for trial 2

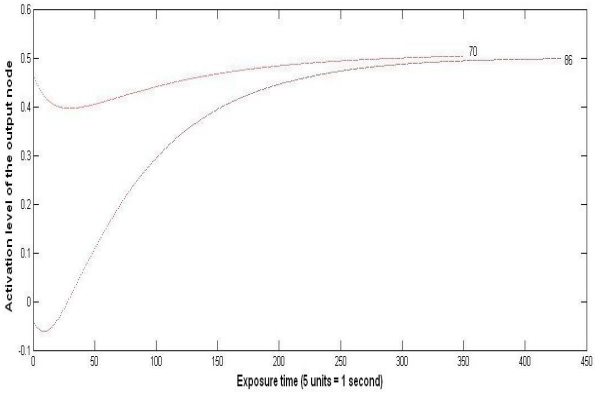


Fig 7c. Activation levels of the output node of N2 for trial 3

Test Trial 3
86, 70, 73, 84, 65, 68, 60, 66, 82, 74, 83, 78, 71, 81, 61, 63, 79, 72, 76, 77
Result: Item = 70; Position = 2; Relative rank = 7. See figure 7c.

Observations and interpretations

The network seems to have learnt the smallest and the largest values of the entire range. It can readily accept 89 wherever it sees it (see response for the item 86 in fig 7c) and never accepts 1 (always a dip). Consequently, in the last few trials, it ends up making a very early choice because they are the biggest numbers it has ever seen and in the first few trials, it ends up making no choice as they are some of the smallest. So, in these ranges the average rank is 10 as that is approximately the average rank at any position of the trial out of a maximum rank of 20. Some of the most significant differences between N1 and N2 are discussed below.

Strategy

From the observations above, it appears that N2 has learned to differentiate between the worth of the items it has seen during its evolution rather than differentiate them within a trial. Its aspiration seems to be set by the first item rather than by the first few items as in N1. It can be vaguely stated as "I need an item larger than the previous items but if it is large enough (say greater than 60), I might accept it". Unlike N1, initial activation level of an item is much lower than that of the previous item. It looks like the pre-caution that is taken by N1 in the first few items is taken after each item in the case of N2. Also, though the dip looks deeper in each item than N1, the LBA seems to be almost the same as what is set by N1 (see corresponding sub-figures of figures 4 and 7). Still, the shoot-up is more conspicuously non-linear than N1 and could differ quite drastically with each item. That is one of the reasons why it could become hasty (see trial 3).

Discussion

We note that artificial evolution has resulted in a CTRNN with a strategy strikingly similar to the optimal strategy developed using rigorous mathematical analysis. It makes us wonder how such a strategy could have evolved. Could the transitions in the evolution of the strategy have followed the same analytical steps that a mathematician uses in his method? What is more, there seems to be at least one more strategy to solve the problem as reflected from the behavior of N2 whose performance is quite comparable to that of N1. Though we have not been able to verbalize its strategy as we could do for N1, from a cognitive viewpoint, we have been able to describe how the network has learnt to assess an item in a sequence based on its position. It has particularly interesting implications on the cognition of judgment when we interpret the dips and shoot-ups in the behavior patterns as being weary and being optimistic respectively. It further implies that a definitive strategy like the analytic solution is not necessarily the only way to solve the Secretary problem; 'patterns of cognition' could work too. Of course the lack of the ability to generalize to the variants of the problem (Tuci et al. 2002) could draw criticisms against making such an inference. Still, it can not be conclusively said that the network has learnt (or rather 'memorized') the boundaries of each trial, thereby performing better than the optimal strategy in some trials; the smallest and largest item in a trial is not always the same. Therefore, the network should have learnt, to some extent, to what to expect based on what a 'judgment' from what it has seen. This 'judgment' is what we refer to as 'patterns of cognition'. What makes it interesting is that it does not seem to be a definitive strategy as the optimal strategy and yet yields a comparable performance.

Conclusion

Different CTRNNs were evolved to solve the Secretary problem and their behavior was analyzed. One of them evolved a strategy similar to the analytically optimal strategy. It was also observed that two different networks with almost equal average performances can evolve totally different behaviors. Above all, an investigative study of the neuronal activation levels has proved to be extremely useful in unveiling the CTRNN's behavior. This approach could particularly be useful when a level of analysis higher than the usual dynamical systems theoretical approach is necessitated. This kind of behavioral analytical approach could be a lot simpler to adopt in case of more complicated probabilistic decision making problems.

References

Beer, R. D (1996). Toward the evolution of dynamical neural networks for minimally cognitive behavior. *From Animals to Animates IV*: Proc of the 4th Intl. Conf. of Simulation of Adaptive behavior, pages 421-429.

Beer, R.D (2003). The Dynamics of Active Categorical Perception in an Evolved Model Agent. *International Society of adaptive Behavior,*11(4):209-243.

Cliff, D., Harvey, I., and Husbands, P (1993). Explorations in Evolutionary Robotics. *Adaptive Behavior*, 2(1): 73-110.

Czogala, E. and Roubens, M (1989). An approach to multi-criteria decision making problems using probabilistic set theory. *European Journal of Operational Research*, 43(3):263-266.

Ferguson, T.S (1989). Who solved the secretary problem? *Statistical Science*, 4(3):282-296.

Goldenberg, E., Garcowski, J., and Beer, R.D (2004). May we have your attention: Analysis of a selective attention task. *From Animals to Animats 8*: Proceedings of the Eighth International Conference on Simulation of Adaptive Behavior, July 13-17.

Harvey, I., Di Paolo, E., Wood, R., Quinn, M, and Tuci, E.A (2005). Evolutionary Robotics: A new scientific tool for studying cognition *Artificial Life*, 11(1-2), pp. 79-98.

Tuci, E., Harvey, I., and Todd, P.M (2002). Using a net to catch a mate: Evolving CTRNNs for the Secretary problem. *Proceedings of the Seventh International Conference on Simulation of Adaptive Behavior.*

Conceptual Structure in Cellular Automata: The Density Classification Task

Manuel Marques-Pita[1,2,3] and Luis M. Rocha[1,2]

[1]Indiana University
[2]Instituto Gulbenkian de Ciência
[3]Portland State University
marquesm@indiana.edu

Abstract

The notion of conceptual structure in cellular automata (CA) rules that perform the density classification task (DCT) was introduced by Marques-Pita et al. (2006). Here we investigate the role of *process-symmetry* in CAs that solve the DCT, in particular the idea of *conceptual similarity*, which defines a novel search space for CA rules. We report on two new highest-performing process symmetric rules for the DCT. We further discuss how our results are relevant to understand, control, and design the collective computation performed by other networks of automata, such as those used to model, for example, living systems.

Introduction

The intersection of biology and computer science has been a fertile ground for some time. Indeed, Von Neumann was a member of the mid-twentieth century Cybernetics group (Heims, 1991), whose main focus was the understanding of natural and artificial systems in terms of communication and control processes. It is interesting to notice that most early computer science developments were inspired by the models of cognition that orbited this group (e.g. seminal work by McCulloch and Pitts, 1943). Since then, the need to understand how biological systems are able to control and transmit information throughout the huge number of components that comprise them has only increased. Certainly, the study of complex network dynamics has been the subject of a substantial body of literature in the last two decades. From pioneering work on networks of automata (Kauffmann, 1969; Derrida and Stauffer, 1986; Kauffmann, 1993) to more recent systems biology models of gene regulation dynamics (Mendoza and Alvarez-Buylla, 1998; Albert and Othmer, 2003; Espinosa-Soto et al., 2004; Kauffmann, 2003), it is clear that to understand and control the biological organization, it is useful to study the dynamics and robustness of models based on complex networks of automata (Chaves et al., 2005; Willadsen and Wiles, 2007).

There has been much progress in understanding the structure of natural networks—be it at the level of their scale-free topology (see e.g. Barabási, 2002; Newman et al., 2006) or of their more fine-grained motifs (Alon, 2007)—as well as some progress on modeling specific biological systems as networks of automata. But we are still to fully grasp *how* the dynamics of complex networks can lead to collective computation and how to harness them to perform specific tasks (see e.g. Mitchell, 2006). Indeed, the need for a better understanding of collective computation in complex natural networks has been identified in many areas. For instance, we know that the way plants adjust their stomatal apertures for efficient gas exchanges on leaf surfaces is statistically indistinguishable from the dynamics of automata that compute (Peak et al., 2004). We also know that the high degree of inter-connectivity in biochemical intracellular signal transduction networks, endows them with the capability of emergent nontrivial classification—via collective computation (Helikar et al., 2008). Plenty more examples exist, which are too numerous to list here.

Clearly, a novel method for describing and understanding how networks collectively compute, would be welcomed. The work presented here is extremely promising in that regard. The conceptual properties uncovered by our "cognitively-inspired" algorithm provide a more compact and intuitive way to understand how complex networks perform the collective computation that they do. One way to think about the conceptual redescriptions produced by our algorithm is as "*dynamical motifs*". Rather than finding common structural network motifs (e.g. Alon, 2007), our redescriptions uncover patterns in the dynamics of automata networks, here specifically the case of cellular automata. Moreover, our redescriptions allow us to understand the global dynamic behavior of novel, high-level conceptual observables built from these redescriptions.

In this paper, we focus on a known problem of emergent computation in CA: the density classification task (DCT) (Mitchell et al., 1996). Specifically, we investigate the role of *process symmetry*, the main conceptual property shared by the majority of CAs that perform the DCT, in (1) defining conceptual spaces where rules with high performance can be found; (2) obtaining more intuitive explanations of the be-

havior of rules that perform the DCT; and (3) exploring the process-symmetric vicinity of high performance asymmetric rules. In forthcoming work, we will expand this approach to study other discrete complex networks such as Boolean networks of automata

Cellular Automata

A cellular automaton (CA) consists of a regular lattice of N cells. Each cell is in one of k allowed states at a given time t. Let $\omega \in \{0, 1, ..., k-1\}$ denote a possible state of a cell. Let state $\omega = 0$ be referred to as the *quiescent* state, and any other state as an *active* state. Each cell is connected to a number of neighbors. Let a local neighborhood configuration (LNC) be denoted by μ, and its size by n. For each LNC in a (n, k) CA an output state is assigned to each cell. This defines a CA rule string, ϕ, the size of which is k^n. In binary CAs, where only two states are allowed($k = 2$), it is possible to classify individual cell state-updates in three categories: (1) *preservations,* where a cell does not change its state in the next time instance $t + 1$; (2) *generations,* state-updates in which the cell goes from the quiescent to the active state; and (3) *annihilations,* state-updates where the cell goes from the active to the quiescent state. The *Initial Configuration* (IC) of states of a CA lattice is typically random. The execution of a CA for a number M of discrete time steps, from a given IC, is represented as the set Θ containing $M + 1$ lattice state configurations.

The Density Classification Task (DCT)

The Density Classification Task (DCT) is one of the most studied examples of collective computation in cellular automata. The goal is to find a binary CA rule that can best classify the majority state in the randomized IC. If the majority of cells in the IC are in the quiescent (active) state, after a number of time steps M, the lattice should converge to a homogeneous state where every cell is in the quiescent (active) state. Since the outcome could be undecidable in lattices with even number of cells (N), this task is only applicable to lattices with an odd number of cells. Devising CA rules that perform this task is not trivial, because cells in a CA lattice update their states based only on local neighborhood information. However, in this particular task, it is required that information be transferred across time and space in order to achieve a correct global classification. The definition of the DCT used in our studies is the same as the one by Mitchell et al. (1993).

The nine highest-performing 1-dimensional CA rules that perform the DCT were analyzed by Marques-Pita et al. (2006). The goal of that analysis was to determine whether there is *conceptual structure* in these rules, and in that case, to investigate the possible *conceptual similarity* among them. These explorations were supported by a cognitively-inspired method, Aitana (Marques-Pita, 2006). In essence, Aitana takes as input a CA rule in its look-up table form, and outputs the same rule but redescribed in a more compact abstraction. Specifically, the output is a set of schemata that can be used (for example) to reason about the conceptual structure concealed in the look-up table of the input rule. Three of these nine rules have been produced by human engineering: ϕ_{GKL} (Gacs et al., 1978; Gonzaga de Sá and Maes, 1992), $\phi_{Davis95}$ and ϕ_{Das95} (Andre et al., 1996); three were learned with genetic algorithms ϕ_{DMC} (Das et al., 1994) or coevolution methods ϕ_{COE1} and ϕ_{COE2} (Juillé and Pollack, 1998). Finally, three of the rules were learned with genetic programming or gene expression programming: ϕ_{GP1995} (Andre et al., 1996), ϕ_{GEP1} and ϕ_{GEP2} (Ferreira, 2001).

Marques-Pita et al. (2006) have shown that there is indeed conceptual structure in these CA rules. All of the studied CAs were redescribed in more compact schemata that made explicit certain conceptual properties most of these CAs have in common. Here we studied one of these properties (process-symmetry) in more detail. The next section summarizes the basics of Aitana's representational redescription architecture, and the conceptual properties found in the studied CAs that perform the DCT.

Aitana: Conceptual Representations of CA

Aitana is largely based on a framework for cognitive development in humans: the *Representational Redescription Model* developed by Karmiloff-Smith (1992), and the *Conceptual Spaces* framework proposed by Gärdenfors (2000). There are a number of (recurrent) phases in Aitana's algorithm: (1) *Behavioral Mastery,* during which CAs that perform some specific collective computation are learned using, for example, genetic algorithms or coevolution. The learned rules are assumed to be in a representational format we call *implicit* (conceptual structure is not explicit). (2) *Representational Redescription Phase I* takes as input the implicit representations (CA look-up tables) and attempts to compress them into *explicit-1* (E1) schemata by exploiting structural regularities within the input rules. (3) *Phase II* (and beyond) look for ways to further compress E1 representations, for example by looking at how groups of cells change together, and how more complex schemata are capable of generating regular patterns in the dynamics of the CA. The focus in this paper is on Phase I redescription.

E1 representations are produced by *modules* in Aitana. Here we focus on the *Wildcard* module. The CA rules studied were redescribed with the this module, introduced in the next section.

The Wildcard Module

This module uses regularities in the set of entries—one for each possible LNC—of a CA's look-up table, in order to produce E1 representations captured by *wildcard schemata*. These schemata are defined in the same way as the look-up table entries for each LNC of a CA rule, but allowing an extra symbol to replace the state of one or more cells within them. This new symbol is denoted by "#". When it appears in a E1 schema it means that in the place where it appears, any of the possible k states is accepted. The idea of using wildcards in representational structures was first proposed by Holland et al. (1986), when introducing Classifier Systems. The wildcard redescriptions used here are *Process-specific*, i.e. they do not allow a wildcard symbol in the place of an updating cell in a schema. This makes it possible for them to describe processes in the CA rule unambiguously. For example, a *generation*, schema $\{\#, \#, \#, 0, 1, \#, 1\}$ prescribes that a cell in state $\omega = 0$, with immediate-right and end-right neighbors in state $\omega = 1$ updates its state to $\omega = 1$ regardless of the state of the other neighbors.

The implementation of the wildcard module in Aitana consists of a simple McCulloch and Pitts neural network. In this *assimilation network,* input units represent each look-up table entry (one for each LNC), and ouput units represent all the possible schemata available to redescribe segments of the input rule (see Marques-Pita, 2006, for details).

Assimilation and Accommodation

Phase I redescription in Aitana depends on two interrelated mechanisms, *assimilation* and *accommodation*[1]. During Phase I, the units in the input layer of an assimilation network are activated to reflect the output states in the input CA rule to be processed. The firing of these units spreads, thus activating other units across the network. When some unit in the network (representing a E1 schema) has incoming excitatory fibers above a threshold it fires. This firing signals that the schema represented by the unit becomes an E1 redescription of the lower level units that caused its activation. When this happens, inhibitory signals are sent back to those lower level units so that they stop firing (since they have been redescribed). At the end of assimilation, the units that remain firing represent the set of wildcard schemata redescribing the input CA rule. Once the process of assimilation has been completed, Aitana will try to "force" the assimilation of any (wildcard-free) look-up table entry that was not redescribed *i.e.* any input unit that is still firing. This corresponds to the accommodation process implemented in Aitana (see Marques-Pita, 2006, for further details).

[1]These two processes are inspired in those defined by Piaget in his theory of Constructivism (see e.g. Piaget, 1952, 1955)

Conceptual properties of CAs for the DCT

One of the main novel findings reported in Marques-Pita et al. (2006) is the fact that most rules that perform the density classification task are *process-symmetric*. Process symmetry for binary CA rules is defined as a bijective mapping between the members of the only two possible sets of schemata prescribing state changes: *generation processes*, which refer to a cell state change from $\omega = 0$ to $\omega = 1$, and *annihilation processes* which refer to the reverse state change.

Using the concept of process symmetry, one can easily define a function that converts a generation into an annihilation process, and vice versa. Such a function of E1 redescriptions, transforms a schema s into its corresponding process-symmetric schema s' by (1) reversing the elements in s using a *mirror* function $M(s)$, and (2) exchanging ones for zeros, and zeros for ones (leaving wildcards untouched), using a *negation* function $N(s)$. Thus, in every process symmetric CA rule, given the set $S = \{s_1, s_2, ..., s_z\}$ of all schemata s_i prescribing a state-change process, the elements of the set of schemata prescribing the converse process $S' = \{s'_1, s'_2, ..., s'_z\}$ can be found by applying the bijective mapping between processes defined by the composition $s'_i = (M \circ N)(s_i)$. This property is illustrated in Figure 1, where the E1 schemata of the process symmetric rule ϕ_{GP1995} (Andre et al., 1996) are shown.

RULE	Generation	Annihilation
ϕ_{GP1995}	{#, #, #, 0, 1, #, 1} {#, #, 1, 0, #, #, 1} {1, #, #, 0, #, #, 1}	{0, #, 0, 1, #, #, #} {0, #, #, 1, 0, #, #} {0, #, #, 1, #, #, 0}

Figure 1: E1 schemata prescribing state changes for ϕ_{GP1995}. Any annihilation (right column) can be obtained by reversing the corresponding generation schema (to the left), and exchanging zeros for ones, and ones for zeros.

Six out of the nine rules analyzed by Marques-Pita et al. were found to be process-symmetric. The remaining three, ϕ_{COE1} and ϕ_{COE2} and ϕ_{DMC} are not.

It is interesting to note that the three non process-symmetric rules were discovered via evolutionary algorithms (GAs and coevolutionary search) which apply variation to genetic encodings of the look-up tables of CAs. Therefore, genotype variation in these evolutionary algorithms operates at the low level of the bits of the look-up table—what we referred to as the implicit representation of a CA. In contrast, the other forms of search of design that lead to the other six (process-symmetric) rules, while not looking explicitly for process symmetry, were based on mechanisms and reasoning trading in the higher-level behavior and

structure of the CA—what we refer to as the explicit representation of a CA. Marques-Pita et al. have also determined that it is possible to define conceptual similarity between the process symmetric CA rules for the DCT. For example, the rule ϕ_{GP1995} can be derived from ϕ_{GKL}. Moreover, the best process-symmetric rule known for this task (at the time) was found via conceptual transformations: ϕ_{MM401}[2] with performance $\mathcal{P}_{149}^{10^5} \approx 0.83$[3]. However, the performance of this rule is still below the performance of the best CA rule for the DCT, namely ϕ_{COE2}, with $\mathcal{P}_{149}^{10^5} \approx 0.86$.

The 4-Wildcard Process-Symmetric Space

Starting with the conceptual similarities previously observed between ϕ_{GKL} and ϕ_{GP1995}, we now report a search of the "conceptual space" where these two CA rules can be found: the space of process-symmetric binary CA rules with neighborhood size $n = 7$, where all state-change schemata have four wildcards. A form of evolutionary search was used to evaluate rules in this space as follows: the search starts with a **population** of sixty-four different process-symmetric rules containing only 4-wildcard schemata; the generation and annihilation **schema sets** for an individual were allowed to have any number of schemata in the range between two and eight; **crossover operators** were not defined; a **mutation operator** was set, allowing the removal or addition of up to two randomly chosen 4-wildcard schemata (repetitions not allowed), as long as a minimum of two schemata are kept in each schema set; in every **generation** the fitness of each member of the population is evaluated against 10^4 ICs, keeping the top 25% rules (elite) for the next generation without modification; **offspring** are generated by choosing a random member of the elite, and applying the mutation operator until completing the population size with different CA rules; **a run** consisted of 500 generations, and the search was executed for 8 runs. There are 60 possible 4-wildcard process-symmetric schemata-pairs. Thus, our search space contains approximately 3×10^9 rules defined by generation and annihilation schema sets of size between 2 and 8.

Our search found one rule with better performance than ϕ_{MM401}. This rule, ϕ_{MM0711}[4] has $\mathcal{P}_{149}^{10^5} \approx 0.8428$. The state-change schema sets for this rule are shown in Figure 2. Even though this search resulted in an improvement, the performance gap between the best process-symmetric rule, ϕ_{MM0711} and ϕ_{COE2} is still close to 2%. Is it possible then, that a process-symmetric rule exists "hidden" in the conceptually "messy" ϕ_{COE2}?

[2]In inverse lexicographical hexadecimal format, ϕ_{MM401} is `ffaaffa8ffaaffa8f0aa00a800aa00a8`

[3]The measure $\mathcal{P}_{149}^{10^5}$ refers to the proportion of correct classifications in 10^5 ICs of length 149

[4]In inverse lexicographical hexadecimal format, ϕ_{MM0711} is `faffba88faffbaf8fa00ba880a000a88`

RULE	Generation	Annihilation
ϕ_{MM0711}	{#, #, 0, 0, #, 1, 1} {1, #, #, 0, #, 1, 1} {1, 0, #, 0, 1, #, #} {1, #, 1, 0, #, #, #}	{0, 0, #, 1, 1, #, #} {0, 0, #, 1, #, #, 0} {#, #, 0, 1, #, 1, 0} {#, #, #, 1, 0, #, 0}

Figure 2: E1 schemata prescribing state changes for the CA rule ϕ_{MM0711}. This CA is process-symmetric.

Process-Symmetry in ϕ_{COE2}

Figure 3 shows the state-change schema sets for ϕ_{COE2}. The performance of this rule is $P_{149}^{10^5} \approx 0.86$. We tested this rule on two sets of 10^5 ICs, one with majority $\omega = 0$, the other with majority $\omega = 1$. Samples were taken from binomial dist. centered around 0.5 (most difficult cases to classify). The performances were, respectively, $P_{149}^{10^5} \approx 0.83$ and $P_{149}^{10^5} \approx 0.89$. Thus, even though on average this is the best CA rule for the DCT, it performs much better when there is a majority of "1's" in the ICs.

RULE	Generation	Annihilation
ϕ_{COE2}	g1 {1, 0, 1, 0, #, #, #} g2 {1, 0, #, 0, #, 1, 1} g3 {1, 1, #, 0, 1, #, #} g4 {1, #, 1, 0, 1, #, #} g5 {1, #, 1, 0, #, 0, #} g6 {1, #, #, 0, 1, 1, #} g7 {1, #, #, 0, 1, #, 1} g8 {#, 0, 0, 0, 1, 0, 1} g9 {#, 0, 1, 0, 0, 1, #} g10 {#, 0, #, 0, 0, 1, 1} g11 {#, 1, 1, 0, 1, #, 0} g12 {#, 1, 1, 0, #, 0, #}	a1 {0, 0, 1, 1, 1, 1, #} a2 {0, 0, #, 1, #, 1, 0} a3 {0, 1, 0, 1, 1, #, #} a4 {0, #, 0, 1, #, #, 0} a5 {1, 0, 0, 1, #, 0, #} a6 {#, 0, 0, 1, #, #, 0} a7 {#, #, 0, 1, 1, 0, #} a8 {#, #, 0, 1, #, 0, 0} a9 {#, #, #, 1, 0, #, 0}

Figure 3: E1 schemata prescribing state changes for ϕ_{COE2}. This is the highest-performing rule for the DCT. ϕ_{COE2} is not process-symmetric.

We claim that this divergence in behavior is due to the fact that ϕ_{COE2} is not process-symmetric. Evaluation of split performance on the ten best rules for the DCT supports this hypothesis (see Table 1). The difference between the two biased performance measures for the non-process-symmetric rules is one or two orders of magnitude larger than for the process-symmetric rules. This indicates that process symmetry seems to lead to more balanced rules—those that respond equally well to both types of of problem.

It is then reasonable to ask: Is there a process-symmetric rule in the conceptual vicinity of ϕ_{COE2}, whose performance is as good (or higher) than the performance ϕ_{COE2}? To answer this question we pursued a number of tests. First, we looked at the CA rule resulting from keeping all annihilations in ϕ_{COE2}, and using only their process-symmetric generations. The performance of that rule was $P_{149}^{10^5} \approx 0.73$.

	$P_{149}^{10^5}$ M→0	$P_{149}^{10^5}$ M→1	*P.* DIFF.
ϕ_{GKL}	0.8135	0.8143	0.0008
$\phi_{Davis95}$	0.8170	0.8183	0.0013
ϕ_{Das95}	0.8214	0.8210	0.0004
ϕ_{GP1995}	0.8223	0.8245	0.0022
ϕ_{DMC}	0.8439	0.7024	0.1415
ϕ_{COE1}	0.8283	0.8742	0.0459
ϕ_{COE2}	0.8337	0.888	0.0543
ϕ_{GEP1}	0.8162	0.8173	0.0011
ϕ_{GEP2}	0.8201	0.8242	0.0041
ϕ_{MM0711}	0.8428	0.8429	0.0001

Table 1: Split performances of the ten best DCT rules. Darker rows correspond to process-symmetric rules; white rows refer to non-process-symmetric rules. For the latter, there is a significant difference in performance: ϕ_{DMC} is better at classifying cases where state 0 is in the majority; ϕ_{COE1} and ϕ_{COE2} are considerably better at solving the problem when state 1 is in the majority. The difference between the split performance measures is one to two orders of magnitude larger for the non-process-symmetric rules.

A second test was the reverse of the first one: keeping all generations of ϕ_{COE2}, and using only their process-symmetric annihilations. The resulting rule has a performance $P_{149}^{10^5} \approx 0.47$.

For the next test, we looked at the *degree of process symmetry* already existing in ϕ_{COE2}. To find this we used the matrix-form representation of ϕ_{COE} shown in Figure 4. Each column contains each of the 128 LNCs for a one-dimensional binary CA rule and neighborhood radius three. These LNCs are not arranged in lexicographical order, instead they are arranged as process-symmetric pairs: the first and last LNCs are process-symmetric, the second, and next to last are also process-symmetric and so on, until the two LNCs in the center are also process-symmetric. Each row corresponds to the E1 (wildcard) state-changing schemata for ϕ_{COE2}. The first nine rows correspond to the annihilation schemata, and the subsequent ones the twelve generation schemata for ϕ_{COE2}.

In any of the first nine rows, a shaded-cell represents two things: (1) that the LNC in that column is an annihilation; and (2) that the LNC is part of the E1 schema labeled in the row where it appears. The twelve rows for generation schemata are reversed. This makes it simple to inspect visually what process-symmetric LNCs are present in the rule, which is the case when for a given column, there is, at least, one cell shaded in one of the first nine rows (an active annihilation), and at least one cell shaded in one of the bottom nine rows (an active generation). Let the $schemata \times LNC$ binary matrix representation in Figure 4 be denoted by A, where all shaded elements in the figure represent 1s and the rest are 0s. In the figure the lighter colored matrix elements are used to distinguish annihilation processes from generation processes, which are shown in a darker color.

Given the ordering of elements in the columns of Figure 4, if a generation row is isolated, and then reversed, the result can be matched against any of the annihilation rows to calculate the total degree of process symmetry between the two schemata represented in the two rows. A total match means that the original generation schema is process-symmetric with the matched annihilation schema. A partial match indicates a degree of process symmetry. This partial match can be used by Aitana's accommodation mechanism to force the highly process-symmetric pair into a fully process-symmetric one, keeping the modified representation only if there is no loss of performance.

More concretely, the degree of process symmetry existing between two schemata S_g and S_a prescribing opposite processes (a generation schema, and an annihilation respectively) is calculated as follows:

1. Pick rows S_g and S_a from matrix A such that S_g corresponds to a generation and S_a to an annihilation.
2. Reverse one of the rows (e.g. S_a). This makes it possible to compare each LNC (the columns) with its process-symmetric pair, by looking at the i^{th} element of each of the two row vectors.
3. Calculate the degree of process symmetry as:

$$\frac{2 \times S_g \cdot S_a}{|S_g| + |S_a|}$$

where, the dot product of binary vectors, $S_g \cdot S_a$ is the number of component-matches; and $|S|$ is the number of ones in a binary vector.[5]

All the generation rows were matched against all the annihilation rows in matrix A, recording the proportion of matches found. Table 2 shows the results of this matching procedure (only highest matches shown). The darker rows correspond to schema pairs that are fully process-symmetric. The first three light gray rows (with matching score 66%) show an interesting, almost complete process symmetry subset, involving generation schemata *g1, g4* and *g5,* and annihilation schema *a9*.

Using the accommodation mechanism in Aitana, we "generalized" the schemata *g1, g4* and *g5* into the more general process symmetric pair of *a9* (that encompasses

[5]While $|x|$ is the notation typically used for cardinality of sets, here, we use it to represent the 1-norm, more commonly denoted by $||x||_1$.

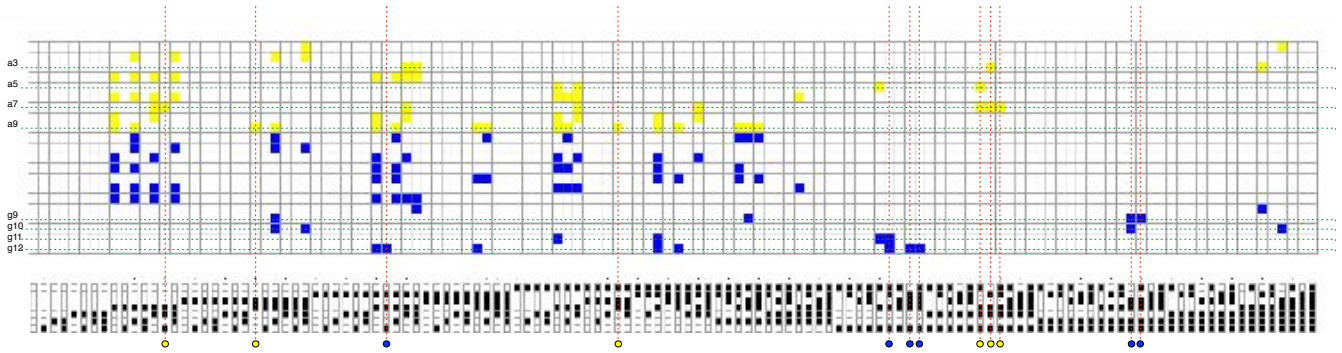


Figure 4: E1 processes for ϕ_{COE2}, without the preservations. Here, the generation rows have been reversed, so that it becomes much easier to determine what LNCs do not have their process-symmetric LNC active. The dotted vertical lines show these LNCs. Each of these state-change prescriptions were removed in $\phi_{COE2_{clean}}$.

Generation schemata	Annihilation schemata	Matching score
g1	a9	66%
g2	a2	100%
g3	a8	100%
g4	a9	66%
g5	a9	66%
g6	a6	100%
g7	a4	100%
g8	a3	66%
g9	a2	25%
g10	a1	66%
g11	a5	50%
g12	a9	33%

Table 2: Degree of process symmetry amongst all the generation and annihilation schemata in ϕ_{COE2}. Gray rows indicate full process symmetry, pink rows indicate a high degree of process symmetry

the three of them), and tested the resulting CA rule. We also "specialized" by breaking *a9* into the three process-symmetric schemata of *g1*, *g4* and *g5*, with performance , $P^{10^5}_{149} < 0.6$ in both cases.

Still working with the degree of process-symmetry in ϕ_{COE2}, it is possible to extract a matrix representation A', containing *only* those LNC process-symmetric pairs in A. In other words, each column in A' will be exactly as in A, as long as the column contains 1s for at least one annihilation and one generation row, otherwise the column is all 0s (the latter is the case for all columns marked with dotted lines in Figure 4). We will refer to the rule represented by the matrix A' as $\phi_{COE2-clean}$—the CA rule that preserves all the process symmetry in ϕ_{COE2}. The "orphan" LNCs removed from A are shown in Figure 5 (white background). Their process-symmetric pairs are in the same Figure (gray background). We will refer to this set of LNC pairs as R.

The last test to be reported consisted in evaluating the CA rules derived from (1) taking $\phi_{COE2-clean}$ as base (each time); (2) adding to it a number of process symmetric *pairs* from R to it; and (3) evaluating the resulting CA rule. This set contains all CA rules that are the same as $\phi_{COE2-clean}$, but adding *one* of the twelve pairs in R; it also contains all the rules that are as $\phi_{COE2-clean}$, including combinations of two pairs from R (66 rules), and so on. The total number of CA rules derived in this way is 4096[6].

The performance of the 4096 rules is shown in Figure 6. Each column shows the performance of the subsets of rules adding one pair of LNCs from R, subsets adding combinations of two pairs, and so on. Note that the median performance in each subset decreases for rules containing more pairs of LNCs from R. However, the performance of the best CA rules in each subset increases for all subsets including up to six LNC pairs, and then decrease.

One of the tested CAs, containing six LNC pairs added to $\phi_{COE2-clean}$, is the best process-symmetric CA for the DCT with $P^{10^5}_{149} \approx 0.85$. The schemata for this CA, ϕ_{MM0802}, are shown in Figure 7. ϕ_{MM0802}, has a performance that is very close to that of the second highest-performing rule known for the DCT, ϕ_{COE1} (see Marques-Pita et al., 2006). However, ϕ_{MM0802} is the highest-performing CA for split performance for the DCT—which means that it classifies correctly the two types of IC it can encounter (majority 1s or majority 0s).

[6]Note that each of the rules tested comes from adding a particular combination of pairs each time to the original $\phi_{COE2-clean}$, as opposed to adding pairs of LNCs cumulatively to $\phi_{COE2-clean}$.

Generation	Annihilation
{0, 1, 1, 0, 1, 0, 1}	*{0, 1, 0, 1, 0, 0, 1}*
{0, 1, 1, 0, 1, 0, 0}	*{1, 1, 0, 1, 0, 0, 1}*
{0, 1, 1, 0, 0, 0, 1}	*{0, 1, 1, 1, 0, 0, 1}*
{0, 1, 1, 0, 0, 0, 0}	*{1, 1, 1, 1, 0, 0, 1}*
{0, 0, 1, 0, 0, 1, 1}	*{0, 0, 1, 1, 0, 1, 1}*
{0, 0, 1, 0, 0, 1, 0}	*{1, 0, 1, 1, 0, 1, 1}*
{0, 1, 0, 0, 1, 1, 1}	{0, 0, 0, 1, 1, 0, 1}
{1, 1, 1, 0, 0, 1, 1}	{0, 0, 1, 1, 0, 0, 0}
{1, 1, 1, 0, 0, 1, 0}	{1, 0, 1, 1, 0, 0, 0}
{0, 1, 0, 0, 1, 1, 0}	{1, 0, 0, 1, 1, 0, 1}
{0, 1, 0, 0, 1, 0, 1}	{0, 1, 0, 1, 1, 0, 1}
{0, 1, 0, 0, 1, 0, 0}	{1, 1, 0, 1, 1, 0, 1}

Figure 5: The set R of twelve LNCs in ϕ_{COE2} (white background) for which their corresponding process-symmetric LNCs are preservations in the original CA rule (italics).

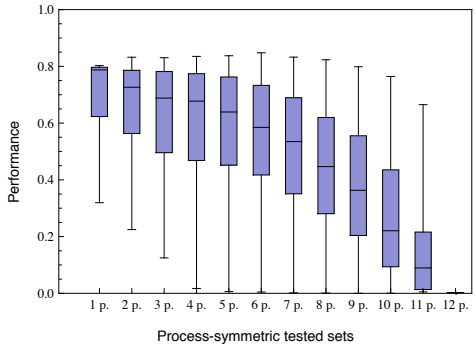


Figure 6: Performances of the 4096 process-symmetric CAs in the immediate conceptual vicinity of ϕ_{COE2}. The best specimen CA is ϕ_{COE2_clean} plus one of the combinations of 6 process-symmetric pairs from R.

Conclusions and Discussion

Besides the two new best process-symmetric CA rules for the DCT, perhaps the most important conclusion from this work is concerned with the fact that representational redescription gives us a new method to relate the local interactions of automata in networks, to the dynamic patterns of collective computation of the network as a whole. Indeed, this constitutes an unexpected advance. When working with implicit CA rules and genetic algorithms, Mitchell et al. (1993) noted that there is no geometry in the space of CA rules represented as look-up state transition tables. Specifically, there was no way of knowing the effect of changing one output in the rule table on its ability to perform a specific collective computation. However, using the conceptually redescribed search spaces we explored here, this is clearly not the case. Conceptual manipulations of CAs in this space result in CAs with similar dynamics—although not necessarily always high performance (Marques-Pita et al., 2006). We

RULE	Generation	Annihilation
Φ_{MM0802}	{1, 0, 1, 0, #, #, #}	{0, 0, 1, 1, 1, 1, #}
	{1, 0, #, 0, #, 1, 1}	{0, 0, #, 1, #, 1, 0}
	{1, 1, #, 0, 1, #, #}	{0, 1, 0, 1, 1, #, #}
	{1, #, 1, 0, 1, #, #}	{0, #, 0, 1, #, #, 0}
	{1, #, 1, 0, #, 0, #}	{1, #, 0, 1, #, 0, #}
	{1, #, #, 0, 1, 1, #}	{#, 0, 0, 1, #, #, 0}
	{1, #, #, 0, 1, #, 1}	{#, 1, 0, 1, #, 0, #}
	{#, 0, 0, 0, 0, 1, 1}	{#, 1, #, 1, 0, #, 0}
	{#, 1, 0, 0, 1, #, #}	{#, #, 0, 1, 0, #, 0}
	{#, 1, #, 0, 1, 0, #}	{#, #, 0, 1, 1, 0, #}
	{#, 1, #, 0, 1, #, 0}	{#, #, 0, 1, #, 0, 0}
	{#, #, 0, 0, 1, 0, 1}	{#, #, #, 1, 0, 1, 0}

Figure 7: Schemata prescribing state changes for ϕ_{MM0802}, the best process-symmetric rule for the DCT.

are not claiming process-symmetry is the important discovery *per se*. Instead, the result we consider to be an important advance is the discovery of conceptual structure in a form of complex network—since, by representing concepts (e.g. process-symmetry), it becomes possible to reason about collective computation in new, less perplexing ways.

Here we showed that the ability to redescribe the dynamics of automata networks into a form that is both easier to understand and to search for new robust behaviors, was very useful for the DCT and CA rules at large. Our results indicate that there seems to exist conceptual structure in the dynamics of networks of automata that perform collective computation. But it should be emphasized that our methodology is not applicable only to CAs and the DCT; it is applicable the study of other complex networks of automata. We are currently exploring the conceptual structure in biochemical networks modeled using Boolean networks. If we can understand the dynamics of, say, a gene regulation network as a form of computation and we uncover the dynamical motifs responsible for that computation, not only do we gain a greater insight about the function of the network, but we can also discover similar network configurations that can be more robust, or those that lead to alternate behaviors more easily. This could prove useful in understanding phylogenetic differences, or differences from wild-type phenotypes. Thus, while here we only present results for the DCT in CA, the approach is quite relevant for both Artificial Life and Computational Biology.

Acknowledgements

We would like to thank Melanie Mitchell for valuable comments. This work was partially supported by Fundação para a Ciência e a Tecnologia (Portugal) grant 36312/2007. We also thank the FLAD Computational Biology Collaboratorium at the Gulbenkian Institute (Portugal) for hosting and providing facilities used for part of this research.

References

Albert, R. and Othmer, H. (2003). The topology of the regulatory interactions predicts the expression pattern of the segment polarity genes in drosophila melanogaster. *J Theor Biol*, 223(1):1–18.

Alon, U. (2007). Network motifs: theory and experimental approaches. *Nat Rev Genet*, 8(6):450–461.

Andre, D., III, F. B., and Koza, J. (1996). Discovery by genetic programming of a cellular automata rule that is better than any known rule for the majority classification problem. In Koza, J., Goldberg, D., and Fogel, D., editors, *Proceedings of the First Annual Conference on Genetic Programming*, pages 3–11. MIT Press.

Barabási, A.-L. (2002). *Linked: The New Science of Networks.* Perseus Books, New York, USA.

Chaves, M., Albert, R., and Sontag, E. D. (2005). Robustness and fragility of boolean models for genetic regulatory networks. *J Theor Biol*, 235(3):431–449.

Das, R., Mitchell, M., and Crutchfield, J. (1994). A genetic algorithm discovers particle-based computation in cellular automata. In Davidor, Y., Schwefel, H. P., and Männer, R., editors, *Proceedings of the Int.Conf. on Evolutionary Computation.*, pages 344–353.

Derrida, B. and Stauffer, D. (1986). Phase transitions in two-dimensional Kauffmann cellular automata. *Europhys. Lett.*, 2:739–745.

Espinosa-Soto, C., Padilla-Longoria, P., and Alvarez-Buylla, E. R. (2004). A gene regulatory network model for cell-fate determination during arabidopsis thaliana flower development that is robust and recovers experimental gene expression profiles. *Plant Cell*, 16(11):2923–2939.

Ferreira, C. (2001). Gene expression programming: A new adapive algorithm for solving problems. *Complex Systems*, 13(2):87–129.

Gacs, P., Kurdyumov, L., and Levin, L. (1978). One-dimensional uniform arrays that wash out finite islands. *Probl. Peredachi. Inform.*, 14:92–98.

Gärdenfors, P. (2000). *Conceptual Spaces: The Geometry of Tought*. MIT Press/Bradford Books.

Gonzaga de Sá, P. and Maes, C. (1992). Gacs-Kurdyumov-Levin automaton revisited. *Journal of Statistical Physics*, 67(3-4):507–522.

Heims, S. J. (1991). *The Cybernetics Group*. MIT Press.

Helikar, T., Konvalina, J., Heidel, J., and Rogers, J. A. (2008). Emergent decision-making in biological signal transduction networks. *Proc Natl Acad Sci U S A*, 105(6):1913–1918.

Holland, J., Holyoak, K., Nisbett, R., and Thagard, P. (1986). *Induction: Processes of Inference, Learning and Discovery*. MIT Press.

Juillé, H. and Pollack, B. (1998). Coevolving the ideal trainer: Application to discovery of cellular automata rules. In Garzon, M. H., Goldberg, D. E., Iba, H., and Riolo, R., editors, *Genetic Programming 1998: Proceedings of the Third Annual Conference*, San Francisco. Morgan Kauffmann.

Karmiloff-Smith, A. (1992). *Beyond Modularity: A Developmental Perspective on Cognitive Science*. MIT Press.

Kauffmann, S. (1969). Metabolic stability end epigenesis in randomly constructed genetic nets. *Journal of Theoretical Biology*, 22:437–467.

Kauffmann, S. (1993). *The Origins of Order: Self-Organization and Selection in Evolution*. Oxford University Press.

Kauffmann, S. (2003). Random Boolean network models and the yeast transcriptional network. *Proc Natl Acad Sci U S A*, 100(25):14796–14799.

Marques-Pita, M. (2006). *Aitana: A Developmental Cognitive Artifact to Explore the Evolution of Conceptual Representations of Cellular Automata-based Complex Systems*. PhD thesis, School of Informatics, University of Edinburgh, Edinburgh, UK.

Marques-Pita, M., Manurung, R., and Pain, H. (2006). Conceptual representations: What do they have to say about the density classification task by cellular automata? In Jost, J., Reed-Tsotchas, F., and Schuster, P., editors, *ECCS'06, European Conference on Complex Systems.*

McCulloch, W. and Pitts, W. (1943). A logical calculus of ideas immanent in nervous activity. *Bulletin of Mathematical Biophysics*, 5:115–133.

Mendoza, L. and Alvarez-Buylla, E. R. (1998). Dynamics of the genetic regulatory network for arabidopsis thaliana flower morphogenesis. *J Theor Biol*, 193(2):307–319.

Mitchell, M. (2006). Complex systems: Network thinking. *Artificial Intelligence*, 170(18):1194–1212.

Mitchell, M., Crutchfield, J., and Das, R. (1996). Evolving cellular automata with genetic algorithms: A review of recent work. In *Proceedings of the First International Conference on Evolutionary Computation and its Applications (EvCA'96)*. Russian Academy of Sciences.

Mitchell, M., Crutchfield, J., and Hraber, P. (1993). Revisiting the edge of chaos: Evolving cellular automata to perform computations. *Complex Systems*, 7:89–130.

Newman, M., Barabási, A.-L., and Watts, D. J. (2006). *The Structure and Dynamics of Networks*. Princeton University Press, Princeton, NJ.

Peak, D., West, J. D., Messinger, S. M., and Mott, K. A. (2004). Evidence for complex, collective dynamics and distributed emergent computation in plants. In *Proceedings of the National Academy of Science*, volume 101, pages 918–922.

Piaget, J. (1952). *The Origins of Intelligence in Children*. International University Press.

Piaget, J. (1955). *The Child's Construction of Reality*. Routledge and Kegan Paul.

Willadsen, K. and Wiles, J. (2007). Robustness and state-space structure of boolean gene regulatory models. *J Theor Biol*, 249(4):749–765.

The Longevity of Distinct Cultures in an Agent-Based Model of Memetic Drift

Jamie Matthews[1]

[1]Evolutionary and Adaptive Systems Group
University of Sussex, Brighton, BN1 9QH, UK
jamie.matthews@gmail.com

Abstract

When a distinct cultural region forms, its rate of absorption into the surrounding culture may be an important variable to take into account when attempting to minimise conflict. This paper describes a re-implementation of Axelrod's agent-based model of cultural dissemination, and uses it to investigate how random drift influences the longevity of distinct regions. Cultural regions are found to be surprisingly resistant to such memetic drift.

Introduction

Cultural artefacts such as beliefs, behaviours, attitudes, languages, art and music tend to spread through populations. Dawkins (1976) proposes a framework for viewing this spread as a Darwinian process. He calls the cultural replicators themselves "memes" and suggests that many aspects of human society may be explained using this paradigm.

Given that beliefs, attitudes and behaviour tend to be passed between people when they interact, how is cultural diversity maintained? Axelrod (1997) describes an abstract agent-based simulation of cultural dissemination to show that global diversity can be maintained despite local convergence.

When a distinct cultural region forms or arrives within a larger culture it may take some time before it becomes assimilated into its surroundings. An understanding of this phenomenon may be important in our desire for a peaceful society, free of tensions between cultural groups.

This paper describes a re-implementation of Axelrod's model and extends it to investigate how cultural drift (random mutation of cultural traits) affects the longevity of cultural distinctions.

Background

Memetics

In *The Selfish Gene* (1976), Dawkins introduces the concept of the *meme* to highlight the fact that there is nothing special about the gene as the fundamental unit of natural selection. This honour should be given to the more abstract *replicator*, "any unit of which copies are made, with occasional errors, and with some influence or power over their own probability of replication" (Dawkins, 2003, pp.149). Memes are another example of a replicator, and have arisen relatively recently on Earth. They are units of human culture which are passed on by imitation. Examples include ideas, melodies, beliefs, fashions, and technologies.

Like genes, memes fulfil the three criteria necessary for the Darwinian algorithm to operate. They are passed from individual to individual through imitation (heredity). Some are more successful at spreading than others (selection). Imitation may be imperfect, and new cultural artefacts may arise as novel combinations of others (variation).

Dawkins went on to suggest how this paradigm could be useful in explaining some features of human culture (such as religions: large complexes of many mutually supportive co-adapted memes). Others have taken the idea further and expanded it to other fields, including the problem of human consciousness itself, the "ultimate" meme complex (Blackmore, 1999).

Maintenance of Differences: Axelrod's Model

If this process of memetic transmission between individuals when they interact is common, how is cultural diversity maintained? Several mechanisms have been proposed to explain why cultural convergence stops before it reaches completion. Most are based on the semantics of the cultural artefacts themselves, such as "preference for extreme views" (Abelson and Bernstein, 1963) or on the specifics of the environment the population inhabits (for example, geographical isolation).

Axelrod (1997) proposes an abstract model based on the fundamental principle that "the transfer of ideas occurs most frequently between individuals . . . who are similar in certain attributes such as beliefs, education, social status, and the like."

Modelling Cultural Dissemination

The Abstract Meme

Although he never uses the terminology of memetics, Axelrod's cultural attributes are clearly analogous. His model

abstracts away the *content*, or semantics, of the memes and leaves behind a list of cultural features. Each feature may take one of a range of values. The values can be thought of as metaphors for the alternative forms of the cultural artefact (if the feature is a hat, the alternative values may be a red hat, a blue hat, a green hat etc). A culture is represented as a string of digits such as "5, 2, 4, 5, 1". In this case the first feature has the fifth of its possible values, and so on.

In abstracting away the semantics of the memes, Axelrod has removed two of the three prerequisites of the Darwinian process. No trait on the cultural string is any more likely to be passed on than any other, and so there is no concept of "fitness" upon which selection may operate. Also, when a trait is passed from one culture to another it is always copied with perfect fidelity: there is no mutation and therefore no variation. His model is only one of heredity, and he asks whether cultural diversity can be maintained even in this most basic situation.

Key Assumptions

Axelrod makes two simple assumptions in his model:

- People are more likely to interact with others who are more similar to them, i.e. share more of their cultural traits.
- Interactions between people are likely to facilitate cultural transmission, increasing the number of traits shared between the two interacting parties.

The Model

Axelrod's model can be described as a randomly updated asynchronous cellular automata. The basic configuration is a square grid of cells. Each cell in the grid represents an agent, and each has its own culture string. Agents may be thought of as individual people, but due to their non-mobility, Axelrod treats them as homogeneous "villages" with a single culture string.

At each step, a site is chosen at random to be active. One of its neighbours (north, south, east or west) is also chosen at random.

> With probability equal to their cultural similarity, these two sites interact. An interaction consists of selecting at random a feature on which the active site and its neighbour differ (if there is one) and changing the active site's trait on this feature to the neighbour's trait on this feature (Axelrod, 1997).

These steps are then repeated for as many events as desired.

For example, consider an initial set of sites with randomly assigned cultures. A site is selected at random to be "active", and has the culture string "5, 2, 4, 5, 1". One of its neighbours is then selected, which has the culture string "3, 9, 4, 5, 7". By chance, these sites share two of their five cultural features (the third and fourth) and so have a 40% similarity and thus a 40% chance of interacting. If they do interact, one of the traits they do not share is copied from the active site to its neighbour. They are now 60% similar, and thus more likely to interact in the future if they are again selected at random.

Distinct Cultural Regions

Can this process of local convergence produce globally distinct cultural regions? To illustrate this, a sample run of the model is described here. The same parameters as Axelrod's initial example were used: five cultural features, each with ten possible values, on a 10×10 grid.

The similarity between two adjacent agents on the grid is shown by the opacity of the line separating them. A black line (100% opaque) indicates no shared features, while a white line (0% opaque and invisible against the white background) indicates that all features are shared. The darker the line, the lower the similarity.

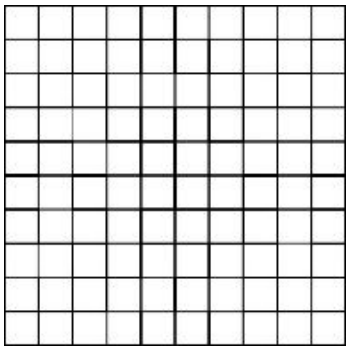


Figure 1: Initial configuration

Initially the value of each cultural feature is chosen at random for each agent (Figure 1). They are unlikely to share many features in common, and so most of the dividing lines are black.

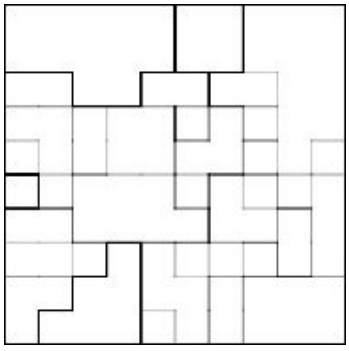


Figure 2: After 25,000 events

After 25,000 events, many cultural regions (groups of adjacent sites with identical cultures) have begun to form (Figure 2).

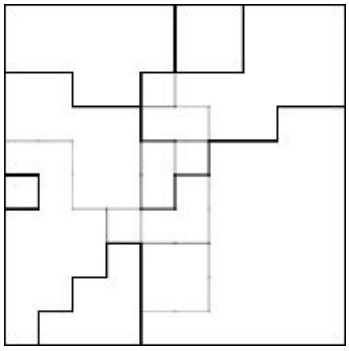


Figure 3: After 50,000 events

After 50,000 events, the cultural regions have become larger, encompassing more sites. Many of the remaining boundaries are light grey, indicating that the sites they divide differ by only one or two features. (Figure 3).

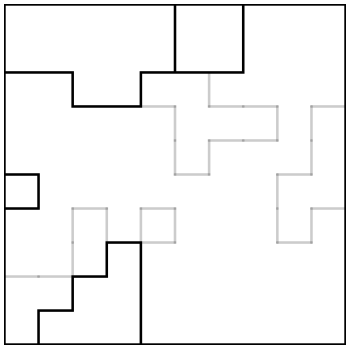


Figure 4: After 100,000 events

By 100,000 events, five clear cultural regions have emerged. The sub-regions within the largest region differ by only one feature. (Figure 4).

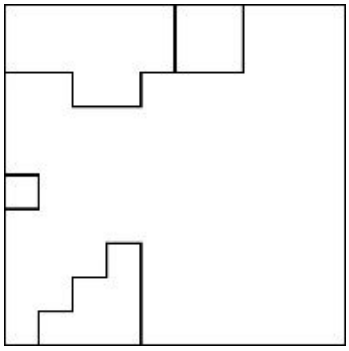


Figure 5: After 125,000 events

By 125,000 events, all sub-regions have disappeared and five main regions are clear. These are surrounded by opaque black lines, indicating that the adjacent sites at their boundaries have no features in common. The simulation is now stable as the probability of any further interactions between members of different regions is zero. (Figure 5).

The above run is a representative example of the behaviour of the model over time with different (randomly selected) initial cultural traits. It is clear that global cultural distinctions can emerge from local convergence.

The Number of Regions

To check the validity of the re-implementation, the simulation was run twenty times with different initial random culture strings. A mean of 3.35 stable regions was found, which is close to Axelrod's average of 3.2 regions for a model with the same parameters.

Axelrod's Experiments

Axelrod goes on to perform several experiments using the simulation by varying parameters (dimensions of grid, number of features in the culture string, number of possible traits of each feature, number of neighbours the active cell can interact with). He draws some interesting conclusions from these experiments, including the non-intuitive result that the average number of stable regions formed *decreases* as the size of the territory increases.

The difficulty with making such inferences from an abstract model is that it is not clear whether they are fundamental properties of the system (and thus could be applied to the more realistic, non-abstract situations upon which the model is based) or whether they are artefacts of the simplifications and assumptions built into the simulation itself.

In order to reduce this problem, it is useful to reintroduce one or more of the features that was removed for the sake of simplicity.

Cultural Drift

Axelrod suggests several possible extensions to the model, one of which he calls *cultural drift*, modelled as a spontaneous change of the value of one of the cultural features in a culture. This is analogous to a "mutation" of a meme, a random change in some aspect of culture. If the feature in question is a red hat, a mutation might involve dropping it into a bucket of green paint. [1] Intuitively, such drift may be common in real populations of interacting individuals.

Modelling Drift

It is simple to add random drift to the above model. At each step of the simulation, each feature of the current active site has probability p of undergoing a mutation. If a feature is selected for mutation, its trait is simply changed to some random value in the range of acceptable traits.

[1] The mutation may also not be random from a semantic point of view - perhaps an individual comes up with a novel new idea which can then be passed on to others. For the purposes of the model, though, such creative acts remain irrelevant. All changes are treated as random.

Analysing the Effect of Drift

As Axelrod points out, it is not obvious how to analyse the effect of drift on the basic model. Without drift, the model eventually stabilises and no further change takes place. The number of distinct regions can then be used as a measure of the heterogeneity of the grid.

When random drift is added, the model never completely stabilises, because a mutation may increase the similarity of two distinct regions (allowing future interactions) or decrease the similarity of two sites within a region (creating a slight boundary between them which may then increase due to future interactions). This raises two practical questions: how to measure heterogeneity of the grid, and when to end the simulation. Axelrod proposes several possible answers to each question, and suggests that preliminary work has shown the interaction between drift and the other parameters of the model to be quite complex.

However, this approach suffers from the same problems discussed above. While it may be possible to perform extensive experiments on the model to analyse the effect of drift, it is not clear how any results found would transfer to the real world.

For example, it may be possible to find a balance between mutation rate and the other parameters which allows the emergence of distinct regions to be preserved despite drift. However, many other factors may be present in the real world which influence this equilibrium but are ignored by the model. Gatherer (2004) uses a genetic algorithm on a similar model to Axelrod's to locate such an equilibrium, and finds that maximal memetic isolation depends on an unlikely combination of parameters. However, his model does not take into account many real-world variables which may be significant.

A more fruitful question to ask might be: given that a process of local convergence may form distinct cultural regions in the *absense* of drift, how does drift affect the stability of those regions? This approach has two main advantages:

- By analysing the effect of drift on the stability of pre-existing regions, we make no assumptions about the original source of the cultural regions themselves. Axelrod's model proposes one mechanism by which such distinctions may form, but he presents many alternative possibilities which have been suggested by others, and which may co-exist with his model. All of these can be taken into account.
- By simplifying the initial conditions of the model significantly, it is much easier to assess the stability of regions by observation, reducing the practical problems discussed above.

The Stability of Distinct Cultural Regions

To analyse how drift affects the stability of cultural regions, the initial conditions of the model were first altered to make the simulation more simple.

Instead of an initially random set of culture strings, the entire grid was made homogeneous by setting the value of every feature to zero. Then, a single mutant site was created by choosing a site at random and mutating all of its features to some random value other than zero. A sample initial configuration of the simulation is shown in Figure 6.

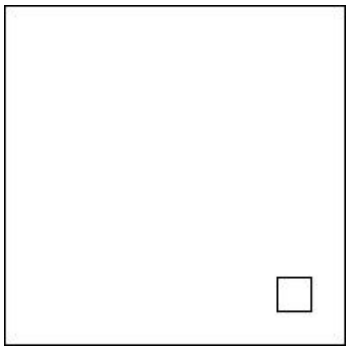


Figure 6: Initial configuration

The single mutant site is entirely distinct from the surrounding region (it shares no traits with its neighbours). In the absence of any drift, this configuration would be completely stable.

Cultural drift was then added to the model in the manner described above (see section entitled "Modelling Drift"). All other features of the model, such as the random process of convergence between neighbouring sites, remain unchanged. Also, the parameters used above (five cultural features, each with ten possible values, on a 10×10 grid) were maintained for simplicity.

The model was then run for a large number of events, and observed until the distinct mutant site disappeared (was absorbed into the surrounding region or became otherwise indistinguishable from the background activity). [2]

This process of absorption begins when a neighbouring site happens to acquire the same value in one of its features as the mutant site (either by direct mutation of that site's feature, or by the spread of that trait from elsewhere on the grid by the normal process of interaction). Now the mutant site shares a feature with one of its neighbours, it has a chance of an interaction with that neighbour which would increase the similarity yet further. In general, once a single interaction between the mutant site and its neighbour took place, the mutant site tended to disappear fairly rapidly (< 5000 events).

A sample run of this process, using a probability $p = 0.0001$ of mutation per feature at the active site per event, is shown below.

[2] To allow for unattended monitoring, this process was observed by repeatedly taking "screenshots" of the grid at 5000 event intervals. So the results are accurate to the nearest 5000 events following the disappearance of the mutant region.

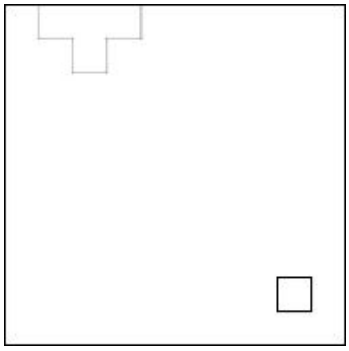


Figure 7: After 30,000 events

Single sites which are mutated become slightly differentiated from the surrounding region. Often, these are immediately reabsorbed, but occasionally they can form small clusters of differentiated sites (Figure 7).

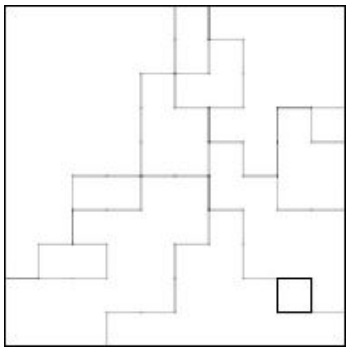


Figure 8: After 40,000 events

Usually, these small regions are short-lived, but occasionally they can "seed" larger disturbances, and chaotic patterns of differentiated regions can grow to cover much of the grid. (Figure 8).

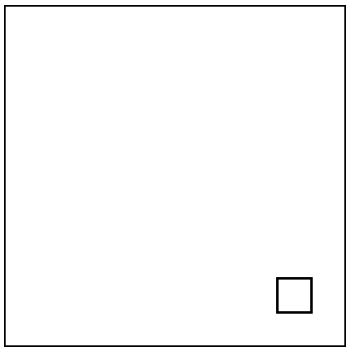


Figure 9: After 70,000 events

Often, even these large disturbances eventually resettle into stability without affecting the mutant site. (Figure 9).

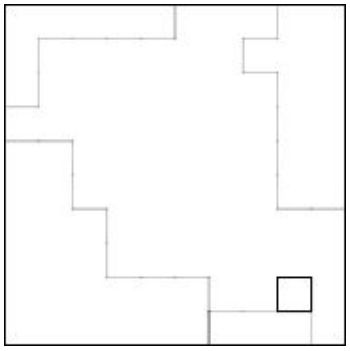


Figure 10: After 120,000 events

After 120,000 events, a new set of distinct regions has emerged and made contact with the single mutant site (Figure 10).

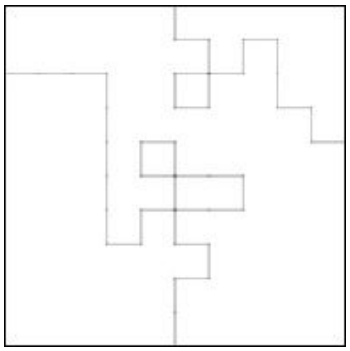


Figure 11: After 125,000 events

Just 5000 events later, the mutant site has been absorbed by the surrounding region and is no longer visible (Figure 11).

From the above run, it can be seen that with low mutation probabilities (low levels of drift), distinct cultural regions can survive significant numbers of interaction events before disappearing. Over ten such runs (with $p = 0.0001$) the mean number of events before the mutant region was absorbed was 108,000.

Rate of Cultural Drift

To further investigate these findings, the experiment was repeated with various mutation probabilities (rates of cultural drift). For each probability value, the model was run ten times, and the mean number of events to the disappearance of the mutant region was found. A graph of the results is shown in Figure 12.

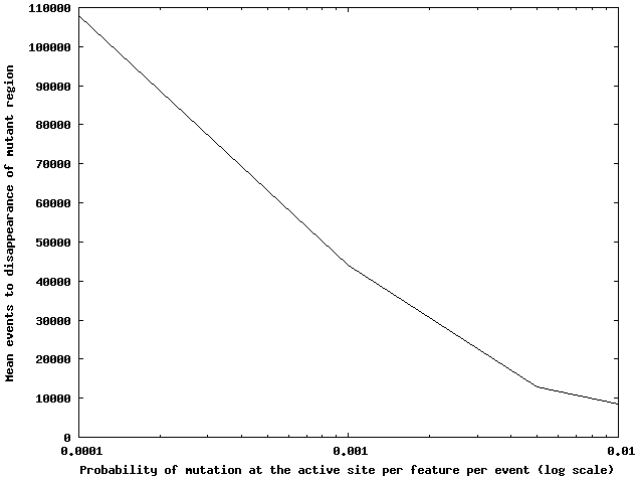


Figure 12: The effect of cultural drift on the longevity of distinct regions

What is surprising about these findings is that even as the mutation probability is increased by several orders of magnitude, the longevity of the mutant cultural region only decreases relatively slowly. One might expect that in a culture with a very high rate of drift, new cultural regions may be absorbed very rapidly as common features may appear regularly by chance, facilitating interaction across boundaries. The results of this experiment suggest that despite such high levels of drift, distinct regions may persist for significant periods of time.

Discussion

It is difficult (and probably unhelpful) to equate these findings with any concrete figures which may be found in the real world, as it is not clear what the rate of interaction (number of "events" per year, say) would be, and such values may vary widely in different regions.

In general though, it is possible to conclude that in relatively homogeneous cultures with low rates of cultural drift (as may be expected to be found in isolated, monocultural regions), any distinct cultures which do form are likely to persist for significant periods of time before being assimilated into the surrounding culture. These distinct cultures may appear through a number of possible mechanisms (including perhaps Axelrod's suggested local-interaction model), but an obvious example might be an invading or migrating group of people from a distant region with a very different culture. Finding aspects of culture in common with the invaders may be difficult, reducing the chances of further interaction and absorption.

The second result suggests that even in a culture with a high rate of drift (such as a modern, fast-changing multicultural society) it may take a considerable amount of time for a new cultural group to integrate into its surroundings. This is often intuitively true when new groups or individuals move into an established culture from afar.

Note that this finding *does not* depend on the content of the memes in either the host culture or the new, distinct culture. It is purely a stochastic interaction process between two different cultures, indifferent to the semantics of the cultural features.

It is important to bear in mind that although the reintroduction of cultural drift brought the model more closely in line with the real world, many important aspects are still missing. The main one is the third feature (in addition to heredity and variation) necessary for Darwinian evolution to take place: selection. The heart of the memetic paradigm is that individual memes or groups of memes may have a greater "reproduction" rate than others, and so may come to dominate in the population. There are several ways this could be added to the model, and this would be an interesting direction for future work. Even without this, it may prove to be useful to translate Axelrod's experiments and conclusions into the language of memes, as it may allow them to be integrated into existing memetic theory.

Summary

This paper has described a re-implementation of Axelrod's agent-based model of cultural dissemination, and discussed its parallels with memetic theory. The model was then used to investigate the longevity of distinct cultural regions in the presence of varying levels of cultural drift. It was found that cultural distinctions can be surprisingly robust, even if mutation rates are high.

Acknowledgements

Thanks to Inman Harvey for helpful advice, suggestions and encouragement.

References

Abelson, R. P. and Bernstein, A. (1963). A computer simulation model of community referendum controversies. *The Public Opinion Quarterly*, 27(1):93–122.

Axelrod, R. (1997). The dissemination of culture: A model with local convergence and global polarization. *The Journal of Conflict Resolution*, 41(2):203–226.

Blackmore, S. (1999). *The Meme Machine*, chapter 17. Oxford University Press, Oxford.

Dawkins, R. (1976). *The Selfish Gene*, chapter 11. Oxford University Press, Oxford.

Dawkins, R. (2003). *A Devil's Chaplain*, chapter Chinese Junk and Chinese Whispers. Phoenix, London.

Gatherer, D. (2003–2004). The inherent instability of memetic systems. *Journal of Memetics - Evolutionary Models of Information Transmission*, 8(2).

Fitness Transmission: A Genealogic Signature of Adaptive Evolution

Thomas Miconi
School of Computer Science
University of Birmingham
Birmingham B15 2TT, UK
txm@cs.bham.ac.uk

Abstract

We introduce *fitness transmission* as a simple statistical signature of adaptive evolution within a system. Fitness transmission is the correlation between the fitness of parents and children, where fitness is evaluated after the number of grandchildren, suitably normalised. This measure is a direct calculation based on a genealogical record, rather than on genetic or phenotypic observation. We point out that the Bedau-Packard statistics of evolutionary activity cannot be used as a reliable system-wide signature of adaptive evolution, because they can produce positive signals when applied to certain "random", non-evolutionary systems. We apply fitness transmission to simple evolutionary algorithms (as well as neutral equivalents) and demonstrate its capacity to accurately detect the presence or absence of Darwinian evolution.

Introduction: Are we evolving yet?

Consider the following problem: imagine that you are observing a simulation, in which a population of agents move, interact and reproduce. The simulation is complex, or its output is obscure (or both), and it is not easy to grasp what, if anything, is going on. Knowing that these agents reproduce, we may ask ourself the question: are they also *evolving*? Are they undergoing genuine natural selection and adaptive evolution? Or are they just perpetuating random genetic traits, following a chaotic trajectory through genotype space without ever undergoing any meaningful evolution?

This question arises from the fact that when a population of reproducing agents is observed, it is not always immediately clear whether the dynamics of the population result from Darwinian evolution, or merely from random variations and stochastic effects such as genetic drift. The particular system at hand may also introduce its own effects, which may bias or alter the dynamics of the population in unpredictable ways. When this system is sufficiently complex, determining whether a population is evolving in a Darwinian sense may not be a trivial task.

Besides its conceptual implications, the question is of practical interest. It is often desirable to determine whether natural selection and evolutionary adaptation are occurring within a given system, especially in the fields of evolutionary computation and artificial life. Indeed in some situations, the onset of significant adaptive evolutionary activity is by itself a major objective of the system: for example, artificial environments such as Echo (Hraber et al., 1997) and Geb (Channon, 2006) were explicitly designed with the aim of exhibiting meaningful evolutionary activity. Being able to detect the presence of genuinely adaptive evolution is a fundamental pre-requisite for the validation of such systems.

Related Work

Traditional methods for detecting natural selection

The problem of detecting natural selection has a long history in biology. Endler's authoritative treatment (Endler, 1986) describes the traditional (that is, non-molecular) methods for detecting natural selection. However, all these methods are based on *phenotypic observation* of chosen traits: they require collecting statistics on the frequencies of certain, predefined traits, and then performing some calculations to determine whether or not natural selection has acted on these traits. This is precisely what we seek to avoid here: we do not ask whether natural selection has acted on this or that trait, but simply whether it is active in the population. Also we want to dispense with detailed phenotypic observation.

The molecular revolution in biology has made it possible to collect vast amounts of genetic data. This creates new possibilities for the detection of natural selection, based on direct assessment of nucleotide variation (Sabeti et al. (2006) provide a recent review). But these approaches require access to a full genetic record. Furthermore, biological genomes are simple sequences of symbols from a four-letters alphabet; but artificial life models need not be so simple in their structure, and this may affect the applicability of these methods.

The Bedau-Packard measure of evolutionary activity

Bedau and Packard (Bedau and Packard, 1992; Bedau et al., 1998) have developed a groundbreaking set of concepts and methods to "discern whether or not evolution is taking place in an observed system." Bedau and Packard are specifically

interested in the *innovations* produced by evolution, and in the capacity of various systems to keep on producing adaptive innovations over time - or not. This requires a method to determine whether an apparent innovation is indeed adaptive or merely the result of random fluctuations, which clearly relates to our own concerns. To this end, Bedau and Packard introduce a set of methods to compute the "*evolutionary activity*" of components and, by extension, of systems.

The Bedau-Packard measures of evolutionary activity are based on *persistence* of adaptive innovations: they identify components that persist over time at a level that exceeds what would be expected under purely random conditions. In the words of Bullock and Bedau (Bullock and Bedau, 2006), "if a particular element persists in the system for a long time, this is likely to be because it is being maintained by selection."

If we are to use persistence "for a long time" as a criterion for detecting evolution, we need a method to determine what "a long time" is. When do we decide that a given element has persisted long enough to be regarded as 'adapted'? To tackle this problem, Bedau and Packard introduced the idea of using a neutral "shadow" of the system under study: a replication of the original system, in which birth, reproduction and death of individuals occur in synchronisation with the real system, but are applied to randomly chosen individuals. More precisely, every time a new individual is being created in the real system under study, a new individual is also created in the shadow; but with the difference that, in the shadow, the parents of the new individual are chosen randomly. Thus the neutral shadow is expected to show the behaviour that would be seen in the system, in the absence of any selective pressure. By comparing the persistence data obtained in this "shadow" to that obtained in the real system, Bedau and Packard argue, it should be possible to detect whether selection and adaptive evolution are present.

Building upon the concept of enduring persistence as a measure of evolutionary activity, Packard and Bedau have developed a series of evolutionary statistics based on persistence information. These statistics include diversity D (the number of different components present at a given time in the population), activity $a_i(t)$ (the age of component i at time t, indicating how long it has persisted so far), cumulated activity $A_{\text{cum}}(t)$ (the sum of the ages of all components present at time t), and new activity $A_{new}(t)$ (the sum of the ages of all components present in the system at time t that are new, but sufficiently aged to indicate adaptive value, divided by diversity at time t).

Bedau-Packard statistics and non-evolutionary systems

Bedau and Packard's measures are arguably the most widely known of their kind. They have been applied to several systems, including artificial ecologies such as Echo, and natural components such as the genera within the fossil record (Bedau et al., 1998). Other researchers have successfully applied them to various systems (Standish, 2002; Channon, 2006; Taylor, 1999). However, it is *not* suitable as a test to detect the presence of adaptive evolution within a system. The basic reason why the Bedau-Packard statistics cannot be used as a detector of evolution by natural selection is that they may attribute a positive score to "random" processes, which are clearly not evolutionary. Importantly, this is the case even if a shadow is used to normalise activity scores. The crux of the matter is that these statistics essentially track "excess" variance in the persistence of components, which is used as a proxy for selection and therefore (it is argued) for adaptive value. The shadow is used to define the level of persistence which can be termed "excess." But excess variance in persistence may be caused by other factors than natural selection. What if, due to some quirk in the rules of the system, some high variance in persistence occurs that is not related to heritable characteristics? If we apply the Bedau-Packard statistics to such a system, we may find that the Bedau-Packard measure classifies such a system as adaptive, even though it is not - even if we use a shadow.

It is easy to devise examples of systems which illustrates this distinction. For instance, consider a population in which reproduction, selection and evolution occurs normally, except for the fact that fitness is randomly attributed to each individual at birth, independently of its genome. That is, while genetic material is transmitted as expected from parent to offspring, this genetic material has no influence over fitness, which is chosen randomly for each new individual. Note that no heritable variance in fitness occurs, nor does any adaptation take place. However, those individuals that happen to be highly "fit" (out of sheer luck) will tend to persist for a long time, and may flood the population with their (short-lived, but nevertheless genetically similar) offspring. No such thing will be observed in the shadow, where reproduction and survival will be random, leading to random diffusion of the genetic material throughout genotype space. Therefore, a difference will occur between the activity counts (and diversity counts) of the shadow and of the real system, creating a positive signal on the Bedau-Packard measure and associated tests.

In figure 1 we describe the results of Bedau-Packard statistics applied precisely to such a system.[1] The system is a simple steady-state genetic algorithm in which, at every "generation", 10 out of the 100 individuals are eliminated and replaced by new individuals, created by copying and mutating a surviving parent. Survivors are selected by fit-

[1] In these experiments we have applied the Bedau-Packard statistics to entire genotypes, in order to follow the authors' method (Bedau et al., 1998). However we are not at all certain that whole genotype persistence is a reliable indicator of evolution. We note that in nature, as soon as recombination and mutation are involved, it is very unlikely that any genotype ever persists for more than one generation.

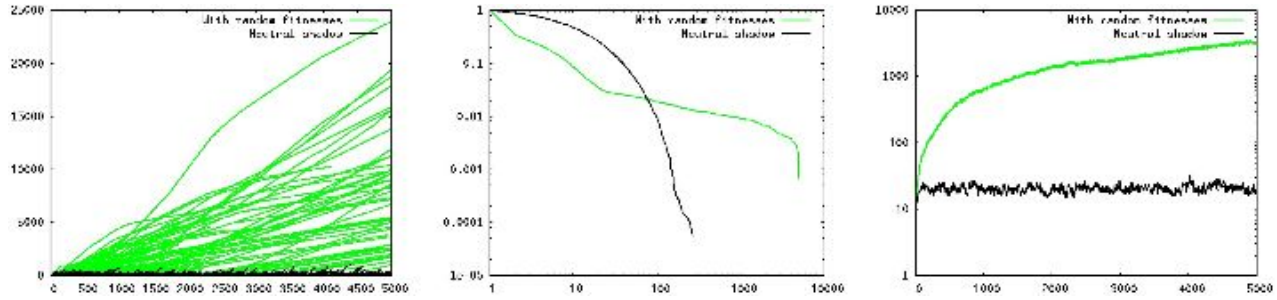


Figure 1: Graphs showing the results of Bedau statistics for a non-Darwinian system, as well as for a corresponding shadow system. The leftmost graph indicates the cumulative frequency counts for each genome over time (that is, the running sum of the frequency of each genome within the population at each generation.) The middle graph indicates the cumulative distribution of persistence counts for all genomes over the history of a run (that is, for each value, the number of individuals that survived longer than this value.) The rightmost graph shows the average cumulative activity $\bar{A}_{\text{cum}}(t) = \frac{A_{\text{cum}}(t)}{D(t)}$ - the sum of all persistence counts of genomes present at a time, divided by the number of genomes present at that time. These graphs are consistent with what is expected from real evolutionary systems. (Bedau and Packard, 1992; Bedau et al., 1998)

ness ranking, and selection of parents occur through tournament selection, very much as in a normal genetic algorithm. However, the fitness of individuals is randomly chosen at birth, independently of their genome. The actual method to "calculate" fitness is to increase a certain counter repeatedly until a random number picked between 0 and 9 is equal to 0 (thus the distribution of fitnesses is exponential.) In the shadow systems for both experiments, selection of survivors and parents are random (thus the shadow systems for both experiments are essentially identical, which predictably results in similar graphs).

Note, in particular, the onset of high average activity, the flattening of the cumulative distribution of persistence counts (with an order of magnitude difference between the longest-living genotypes of real and shadow systems), and perhaps most significantly the appearance of large "telic waves" (Bedau and Packard, 1992) (tall, lengthy lines) in genome frequency plots, despite the decidedly non-teleological nature of these environments. All these are regarded as positive signals of evolutionary activity in the Bedau-Packard framework. (Bedau and Packard, 1992; Bedau et al., 1998)

Surely many other examples could be found. More generally, these simple systems illustrate the fact that high variance in persistence can be caused by many other processes than natural selection. "Random" systems, in which no meaningful evolution or adaptation occurs, can still obtain high marks on the Bedau-Packard measure if they produce high variance in genetic persistence.

Of course, in our toy system, it is easy to see (just by looking at the rules) that variance in persistence is due to random fluctuations, and that no true natural selection exists. But this is precisely the heart of the matter. First, when we study a real system, we may not have access to its internal rules, so clearly in this case we cannot use the Bedau-Packard statistics as a test of Darwinian evolution. But even if we *do* have full access to the rules of the system, the complexity of even mildly elaborate systems may prevent us from asserting with absolute certainty whether or not a "random force" generates strong variance in persistence. For example, considering a system similar to Echo (Hraber et al., 1997), can we really exclude, a priori, that such a factor could come into play? Can we offer absolute guarantee, simply by looking at the rules of the system, that no weird effect will arbitrarily and significantly favour certain individuals rather than others (without being based on these individuals' heritable features)? The answer, of course, is that we cannot. It follows that, if we apply the Bedau-Packard statistics on such a system and obtain a positive result, we cannot (in the absence of further information) use this fact alone to conclude that adaptive evolution is active in the system.

It is important to be clear about the meaning of this result: this should not be interpreted as a minimisation of the importance of Bedau-Packard statistics. Rather, this is a reminder that these statistics should not be used to detect adaptive, Darwinian evolution within a system, even by normalising against a shadow. *If* we know, a priori and through other means, that the system is indeed affected by genuine adaptive evolution, and if we can rest assured that "weird" effects will be nil or negligible, *then* we can fruitfully apply the Bedau-Packard measure to assess the dynamics of long-term evolutionary innovation within this system. The valuable contribution of these statistics in this regard has often been pointed out. However we cannot use these statistics to determine the presence of evolution by natural selection within a system, as opposed to any system-induced dynamics which create high variance in persistence: the Bedau-Packard statistics are not designed to distinguish the former from the latter, even by using a shadow system.

Fitness Transmission: A test statistic for natural selection

Darwinian evolution: randomness, selection and heredity

In general, evolution is simply defined as a change in the frequencies of heritable innate characteristics within a reproducing population, from one generation to the next. Natural selection, one of the mechanisms that guide evolution, is broadly defined as variance in reproductive success caused by heritable innate characteristics. Darwin realised that adaptive evolution automatically results from the existence of fitness-impacting, heritable variations. Variations that improve fitness will be propagated quickly, initiating thriving lineages; while those that reduce fitness will hinder their own propagation, creating feeble (or even quickly extinct) lineages. Thus lineages constantly branch out into variants, and the uneven distribution of these branches, being dramatically skewed towards those which result from fitness-enhancing variations, will result in the overall effect that the newer descendants of the original lineage will tend to be those better adapted to their current, local environment: heritable fitness-affecting variation will have "steered" the original lineage towards adaptive directions among all those encountered by mutational variations.[2]

Note that although this process will usually result in a modification of the species over sufficiently long periods of time, it will also often result in temporary stasis. If a species happens to be located at a convenient local optimum in the fitness landscape, then variations which depart from the optimum will mostly reduce the fitness of the individual. In this case the differential transmission of characteristics enforced by natural selection will actively maintain the population around the optimum: the population will be constantly steered back towards its current position. This phenomenon, known as 'stabilising selection', is actually regarded as more common than directional selection (see (Ridley, 1993), Chap. 4.4).

Fitness Transmission: A genealogic signature of Darwinian evolution

From this discussion we can deduce a method to detect the active presence of natural selection. If fitness-impacting, heritable traits are actually being transmitted and propagated, then this should have an impact on the genealogical record: individuals sharing a common lineage, being more likely to inherit common fitness-impacting characteristics, should therefore tend to exhibit slightly similar fitnesses in comparison to the rest of the population. In other words, if some fitness-affecting traits are being transmitted, then there should be some degree of *correlation* between the fitnesses (that is, the reproductive success) of individuals from a common lineage: the transmission of heritable, fitness-affecting traits should result in some degree of *differential transmission of fitness*.

Fitness transmission is our proposed signature for natural selection. It is, quite simply, the statistical correlation between the fitness of children and parents. The basic idea of fitness transmission is that, when natural selection is active in a population, parents and children should exhibit a *tenuous*, but *persistent* correlation in fitness.

Calculation of Fitness Transmission

Number of grandchildren as a measure of fitness

The term "fitness" is notoriously ambiguous and can be a significant source of confusion (Dawkins, 1982, Chap. 10). A common practical measure of an individual's fitness is its number of grandchildren, rather than number of children. To have many grandchildren, an individual must not only have many children, but these children themselves must also be successful in reproducing; this corresponds to the intuitive notion of fitness as ability to pass on one's genes. We will use the number of grandchildren as a measure of individual fitness. Therefore, to measure fitness transmission, we measure the statistical correlation between the number of grandchildren (NOGC) of an individual, and that of its children.

Fitness correlation is a local measure in time. That is, we divide the record in time periods, or "slices," and calculate fitness transmission independently for each period. This is done by only considering individuals born within this time period for the "child" data set of each period (the parents of these individuals are then collected in the "parents" data set, independently of their time of birth). However, the reproductive success for a given individual may be collected over its entire history, even if it goes beyond the time-slice being considered.

Comparing what is comparable

As usual when calculating statistical correlations, care should be taken in only comparing what is comparable: conflating data from widely different distributions may result in artificial, spurious correlations. In some artificial systems, selective conditions may change widely over the course of an evolutionary run, even with a fixed fitness function. This may wreak havoc on undiscerning evaluations of statistical correlation. For example, in a simple genetic algorithm, if strict ranking is used, surviving and reproducing entails dislodging a previous survivor; but as evolution proceeds towards an optimum, and new champions are increasingly well-adapted, it becomes increasingly difficult (and thus rare) for new individuals to dislodge previous champions. This means that the children's fitness will tend to go down (because more of them disappear without a descent) and the parent's fitness will tend to go up (because they remain in the population longer) over time. This alone is sufficient to

[2] Or, in short: as creatures replicate, genes mutate, adaptations proliferate, and species originate.

create a strong, *negative* correlation between the fitnesses of parents and children over the whole process: earlier parents would have a moderate number of grandchildren, each with a good chance to reproduce; while later parents would accumulate enormous numbers of grandchildren, which would have comparatively low reproductive success.

To avoid this, we must ensure that we only consider quantities (that is, fitnesses) obtained under similar conditions. To this end, the periods over which reproductive successes are measured should start at the same point in time, so that we can ensure that they are obtained over equivalent conditions. In practice, this means that when we compare the NOGC of an individual X and its parent, we should only consider the grandchildren of the parent that were born at the same time as X or later. This ensures a "fair game" between the parent and the child: both scores will be obtained under similar circumstances, and results obtained by the parents in earlier (possibly harsher or milder) circumstances will not spoil the data.

Necessary normalisations

Unfortunately, the choice of using NOGC as a measure of fitness introduces an obvious problem: the NOGC of an individual and that of its children are clearly not independent quantities. Saying that A has many grandchildren is saying that A's children have many children, and therefore, *out of this fact alone*, are likely to have many grandchildren themselves, even with random reproduction. This problem can be easily addressed by normalisation to make the considered values independent. To do this, we do not use the raw NOGC for the parents; rather, for every parent-child pair, we consider the parent's NOGC *minus the number of children of this particular child*. This modified NOGC is an estimation of the parent's fitness that is not biased by this particular child's own success, and thus any correlation represents a true correlation in fitness.

Another, less significant problem is that, in general, the population of interest will be finite. The consequence is that the reproductive successes of individuals living during the same period of time are not independent: any child for a given individual is one less opportunity for another individual to have a child. Even with random mating and reproduction, if one individual happens to have more children than average, then any other randomly picked individual is mechanically more likely to have fewer children than average. In other words, limited population introduces a slight negative correlation between the modified NOGC of parents and children. This effect is much less important than the previous one, but may be noticeable, especially with small populations. A simple solution to this problem is to normalise the modified NOGC of the parent: for every parent-child pair (P_i, C_i) from the slice, we divide the modified NOGC of P_i by the total sum of all grandchildren of all other parents within the slice - minus C_i's children. The resulting proportion is independent of this child's own success.

Those normalisations are made necessary by the fact that the quantities under scrutiny are not independent. They would become unnecessary if, instead of evaluating fitness transmission from parents to children, we attempted to calculate it between grandparents and grandchildren. The problem, of course, is that any signal would be much weaker due to the increased indirection - often to the point of being drowned in noise.

Calculation method for fitness transmission

Where does this leave us? From all these considerations, we can deduce the following calculation method for fitness transmission:

- Divide the entire genealogic record into discrete periods of time. If the system is generational, generations may be used as time periods.
- For every time period within the genealogic record, perform the following operations:
  1. For every individual C_i born during this time period, find its parent Pi (which may be born at any time before C_i, not necessarily during this time period) and store the resulting parent-child pair (P_i, C_i). Note that any given individual may occur in several pairs.
  2. For every stored parent-child pair (P_i, C_i), retrieve their respective total number of grandchildren (NOGC) $N(P_i)$ and $N(C_i)$, born during or after (not before) this time period.
  3. Elimination of dependency: for every pair (P_i, C_i), subtract the number of children of C_i from the $N(P_i)$, resulting in the new value $N'(P_i)$.
  4. Normalisation: for every parent P_i in the set of parent-children pairs for this time period, divide $N'(P_i)$ by the sum of all grandchildren of all other parents $P_{j\neq i}$ - carefully excluding C_i and its descendants from the count. This results in a final value $N''(P_i)$
  5. Calculate the statistical correlation between the $N''(P_i)$ and the $N(C_i)$ variables over all parent-child pairs for this time period, using the standard Pearson formula:

$$\mathrm{Corr}(X,Y) = \frac{\sum_{i=1}^{N}(x_i - \bar{X})(y_i - \bar{Y})}{(N-1)\sigma_X \sigma_Y}$$

The resulting value $\mathrm{Corr}(N''(P_i), N(C_i))$, for every time period, is our estimator for the intensity of fitness transmission during that time period.

Experiments

Experimental settings

Our purpose in this section is to set up a couple of experiments in order to determine whether fitness transmission

is indeed a reliable indicator of Darwinian evolution. To do this, we will use simple evolutionary systems with predictable dynamics, in which the presence or absence of evolution can be easily controlled. We will apply our calculation method to these systems and determine whether the presence or absence of Darwinian evolution was successfully detected.

To perform our experiments, we used genetic algorithms involving a population of 1000 individuals, over 100 generations. We considered two optimisation problems: the Rosenbrock function $100(x^2 - y^2)^2 + (1 - x)^2$ (using genomes of 2×12 bits) and a very simple OneMax problem over 20 bits. The Rosenbrock function is a commonly used test function in the field of optimisation. The purpose of the simple OneMax problem is to examine the behaviour of different algorithms on very easy problems, when the the global optimum is discovered quickly. In our algorithms, at each generation, a new population is created either by applying bitwise mutation to a parent selected from the previous generation, or (with 66 % probability) by applying one-point crossover between two parents, and then applying bitwise mutation to the resulting offspring. The probability of mutating (flipping) each bit is the inverse of the total number of bits in the genome, rounded to the closest higher percent; thus, on average, each genome should undergo about one mutation. As explained below, we tested different methods of selection and replacement.

As a point of comparison, we need a "neutral" version of the genetic algorithm, which preserves as many features of the algorithm as possible, while effectively removing Darwinian evolution. We chose to use a system in which every new individual was attributed a random genotype (and therefore a random fitness) at birth, regardless of the genetic make-up of its parents. This is different from purely random selection in that selection still occurs, and is still based on fitness; however the randomness of the reproductive process prevents any meaningful evolution: fitness-affecting traits are still present, but not heritable. A satisfactory measure of evolutionary activity should be able to detect the absence of real evolution and return a zero value for this situation.

A simple genetic algorithm

We first describe the calculation of fitness transmission in a standard simple genetic algorithm, using tournament selection. In this algorithm, each new individual is created by selecting parents from the previous generation (using tournament selection), and generating offspring as previously described. The process is iterated until the new population is filled.

Figure 2 shows the results of these calculations, applied to the "fossil record" generated by our simple genetic algorithm. This figures shows the results for the Rosenbrock function optimisation problem with 20 bits, both with normal reproduction and with reproduction based on random

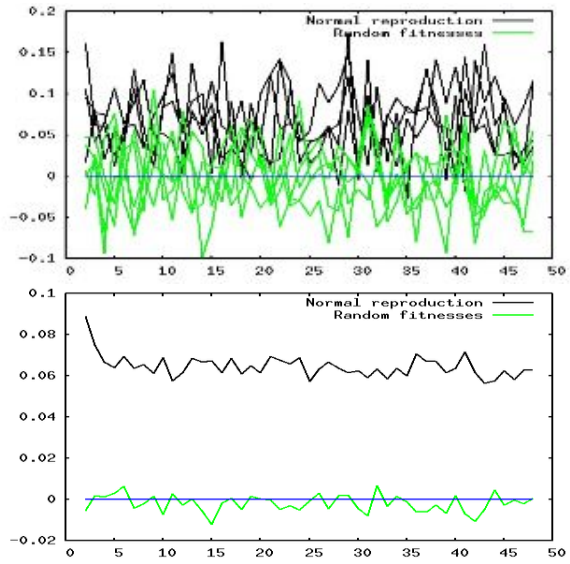


Figure 2: Rosenbrock function, non-overlapping generations, 5 different runs (top) and average of 50 different runs (bottom).

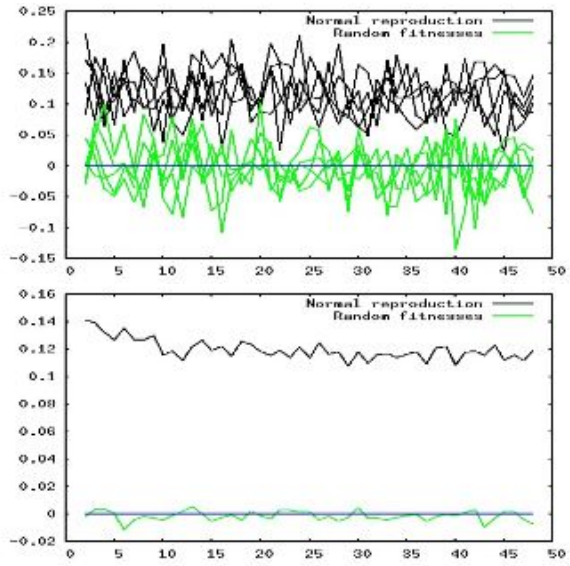


Figure 3: OneMax function, non-overlapping generations, 5 runs (top) and average of 50 runs (bottom).

phenotypes. The top graph shows the results of 5 different run for each of these reproduction methods, while the bottom graph shows average curves over 50 runs. Figure 3 shows the same data for the OneMax problem. In the normal selection case, the correlation between the number of children of parents and children is distinctly positive (especially at the very beginning at the run) and stabilises to a positive plateau. The enduring positive value indicates that the population is constantly and actively maintained in the

vicinity of the global optimum (which is reached quite early in the OneMax problem) through active evolutionary forces. Even though the optimum has been reached, mutation constantly disperses the population, and Darwinian evolution constantly drives it back. Stabilising selection results in a positive value for differential fitness transmission. In the case of random genotypes, as expected, no meaningful fitness transmission occurs.

That the enduring presence of fitness transmission in this case is caused by mutation can be seen quite readily. If we set the mutation rate to zero, then the population converges totally: all individuals end up sharing the exact same genome, and diversity disappears. From this point on, all individuals having exactly the same genotype, evolution simply stops. The result is that evolutionary activity, as indicated by fitness transmission, quickly goes to zero (with noise oscillations) after an initial phase of high activity (see Figure 4). This illustrates the capacity of fitness transmission to distinguish between active stabilising selection on the one hand, and passive stillness caused by absence of genetic variation on the other (though this ability breaks down in extreme situations, as discussed in section .)

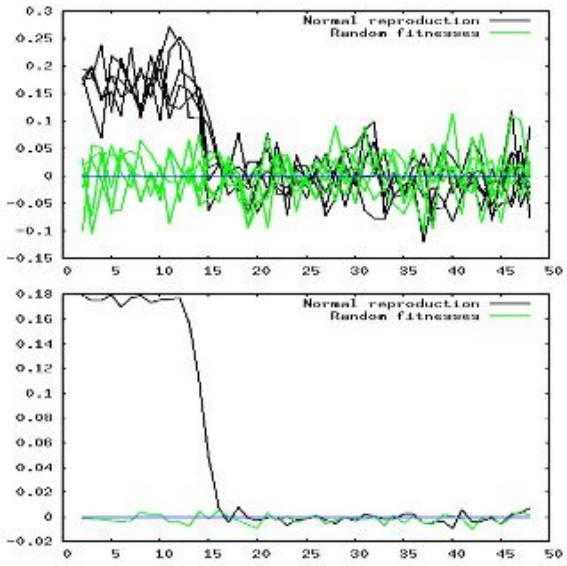


Figure 4: OneMax function, non-overlapping generations, without mutation, 5 runs (top) and averages of 50 runs (bottom).

Removing selective gradient among parents

Here we try to make the problem more challenging problem by reducing the scope of selection. To do this, we modify our algorithm as follows: at every generation, a small set of survivors is selected from the population through strict ranking selection, and the parents for the next generation are then *randomly* selected from among this set of survivors. Offspring are created as previously mentioned (66% crossover, mutation, etc.) The effect of this modification is to effectively remove any selective gradient among parents. This is because the only effect of selection in this system is to decide which individuals become parents in the first place. Once individuals have been selected as parents, their number of children is random, and as a result is not affected by natural selection. In particular, note that if we had tried to evaluate fitness by the number of children alone, then no fitness transmission could be detected: no correlation can exist between the number of children of parents and children, simply because all parents have a random number of children. However, as shown in figures 6 and 5, our measure for fitness transmission is able to detect the signal created by this more indirect form of natural selection.

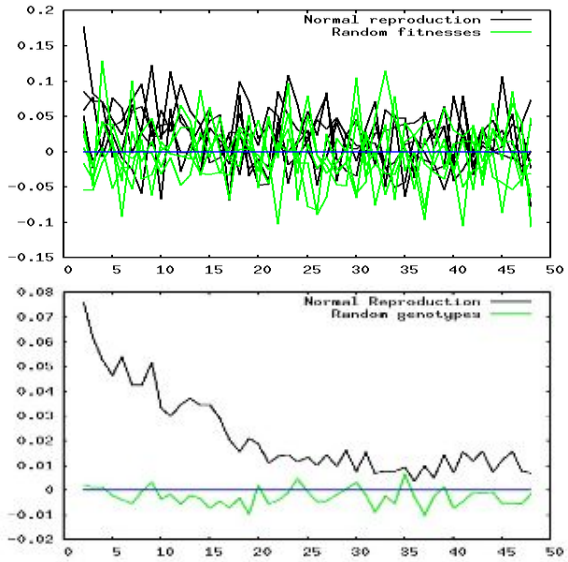


Figure 5: Rosenbrock function, non-overlapping generations with ranking-based survival and random parent selection, 5 runs (top) and averages over 50 runs (bottom). The initially high signal goes to a very low, but still noticeably non-zero value.

Limitations of fitness transmission

Although fitness transmission is valuable as a signature of adaptive evolution, several limitations must be mentioned.

Extreme stabilising selection: While fitness transmission is able to detect moderate stabilising selection, it breaks down in the extreme situation of *absolute* stabilising selection - that is, when only one genotype is viable, and any individual that differs from the optimum systematically fails to reproduce. In this case, no heritable variation in fitness exists. In this situation, stabilising selection has the effect of effectively freezing the reproducing population, and therefore becomes invisible to fitness transmission.

Only one extant lineage: More generally, there are pathological situations in which genealogic methods can not be

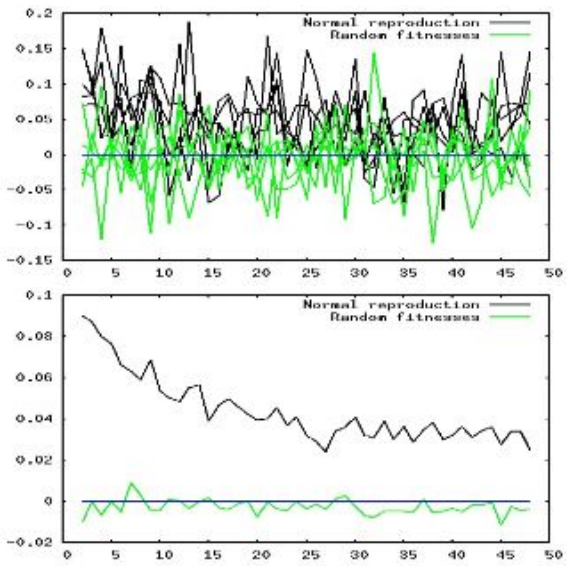


Figure 6: OneMax function, non-overlapping generations with ranking-based survival and random parent selection, 5 runs (top) and averages over 50 runs (bottom).

used at all. One such situation occurs when all individuals present at any given time share the exact same genealogic tree - in other words, when there is never more than one lineage in the population. In this case, while Darwinian evolution can certainly occur, the presence of only one lineage at any time within the population prevents the possibility of inter-lineage comparison, upon which genealogic analysis relies. For example, consider a non-overlapping generational system, such that at every generation, two individuals are selected to serve as parents for the next generation, and *all* the individuals from the new generation are children of *both* of those selected parents. Since all individuals will always share the exact same set of parents, grandparents, and so on, fitness transmission cannot be applied. We believe that this situation is sufficiently exotic to preserve the usefulness of genealogical analysis. In addition, such situations can be easily detected in any system for which a genealogical record exists.

Non-biological selection: A more subtle aspect of fitness transmission is that it detects natural selection in the most general sense, applying to *any* heritable character, including those that we might not think of as "biological". Any kind of heritable trait that affect reproductive success (genetic, epigenetic, cultural, etc.) will be detected by fitness transmission. If the objective is to detect biological natural selection *alone*, then fitness transmission should not be used on its own.

Conclusion

We have shown that differential fitness transmission is a useful signature of Darwinian evolution, which can be detected in genealogical record by using simple statistics. We believe that this signature may be more suitable for this purpose than previously suggested methods for detecting evolution. We have applied this statistic to the genealogical records generated by real evolutionary algorithms, demonstrating its capacity to detect the presence or absence of adaptive evolution.

Acknowledgements

This research was conducted at (and funded by) the School of Computer Science at the University of Birmingham, under the supervision of Alastair Channon. This paper benefited greatly from the comments of anonymous reviewers for the *Artificial Life* journal.

References

Bedau, M. A. and Packard, N. H. (1992). Measurement of evolutionary activity, teleology, and life. In Langton, C., Taylor, C., Farmer, D., and Rasmussen, S., editors, *Artificial Life II: Proceedings of the Workshop on Artificial Life*, pages 431–461, Redwood City, CA. Addison-Wesley.

Bedau, M. A., Snyder, E., and Packard, N. H. (1998). A classification of long-term evolutionary dynamics. In Adami, C., Belew, R. K., Kitano, H., and Taylor, C. E., editors, *Artificial Life VI: Proceedings of the Sixth International Workshop on the Synthesis and Simulation of Living Systems*, pages 228–237, Cambridge, MA. MIT Press.

Bullock, S. and Bedau, M. (2006). Exploring the dynamics of adaptation with evolutionary activity plots. *Artificial Life*, 12:1–5.

Channon, A. D. (2006). Unbounded evolutionary dynamics in a system of agents that actively process and transform their environment. *Genetic Programming and Evolvable Machines*, 7(3):253–281.

Dawkins, R. (1982). *The extended phenotype: The gene as the unit of selection*. Freeman.

Endler, J. (1986). *Natural Selection in the Wild*. Princeton University Press.

Hraber, P. T., Jones, T., and Forrest, S. (1997). The ecology of Echo. *Artificial Life*, 3(3):165–190.

Ridley, M. (1993). *Evolution*. Blackwell Scientific.

Sabeti, P., Schaffner, S., Fry, B., Lohmueller, J., Varilly, P., Shamovsky, O., Palma, A., Mikkelsen, T., Altshuler, D., and Lander, E. (2006). Positive Natural Selection in the Human Lineage. *Science*, 312(5780):1614–1620.

Standish, R. K. (2002). Diversity evolution. In Standish, R. K., Bedau, M., and Abbas, H. A., editors, *Artificial Life VIII: Proceedings of the Eighth International Workshop on the Synthesis and Simulation of Living Systems*, pages 131–137. MIT Press.

Taylor, T. (1999). *From artificial evolution to artificial life*. PhD thesis, University of Edinburgh.

Anticipating Future Experience using Grounded Sensorimotor Informational Relationships

Naeem Assif Mirza, Chrystopher L. Nehaniv, Kerstin Dautenhahn and René te Boekhorst

Adaptive Systems Research Group, University of Hertfordshire, College Lane, Hatfield, AL10 9AB, U.K.
c.l.nehaniv@herts.ac.uk

Abstract

Operational definitions and applications of the sensorimotor *experience* of an artificial embodied organism are presented along with a mathematical metric for distance between experiences based on Shannon information. We describe a simple robotic experiment that illustrates how an artificial embodied agent can use its own history of experience combined with the experience metric to predict future experience. Present sensorimotor experience is used to find the most similar past experience using the geometry of its growing and changing experience metric space. This is then used to ground the ontogeny of autonomous prospective capability in interacting with the environment, e.g. to anticipate forthcoming changes in environment based on temporally extended past experiences.

Introduction

Increasingly, the importance of embodiment and situatedness within complex and rich environments are becoming recognized as a crucially important factors in engendering intelligence in an artifact (cf. for example Clancey (1997); Pfeifer and Bongard (2007), and the philosophical position regarding 'structural coupling' of Maturana and Varela (1987)). Living organisms in particular experience and re-experience particular recurring patterns of trajectories of interactions with the environment through their sensing and acting; and these habitual trajectories can form the basis of prospection, further development, and adaptation (Varela et al., 1991). [1]

Moreover, it is in how an artificial agent develops its capabilities over its life-time of interactions (*ontogeny*) that is important in building a *grounded* intelligence, able to adapt to unknown and changing environments (including long- and short-term variations in its embodiment and in its sensory or motor repertoire). Especially given the complexity of interactions in natural environments, and the richness of sensors available to modern robots, whose properties change over time in different environments or with changing embodiment, it is largely infeasible and impractical to attempt to foresee and model the situations a robot (or other artificial agent) may encounter and how to adapt to them in advance (e.g. Brooks (1999)). Instead, autonomous methods for bootstrapping development without prior knowledge of the structural coupling relationship based on enactive construction and development of intelligence behaviour warrant investigation, both from the perspectives of engineering applications as well as from the viewpoint of a generalized biology. Building on basic 'phylogenetic' capabilities, such an approach is hypothesized to allow for a basic of autonomous, enactive development in embodied models of developmental cognitive systems with expanded temporal horizon of their perception and action (Nehaniv et al. (2002),Vernon et al. (2007), Mirza et al. (2007)).

Our goal is to research methods that can be used by an artificial embodied agent to develop its capabilities through its ongoing interactions with its environment, while scaffolding its adaptation on the basis of previous experience and previously achieved adaptation. In earlier work we introduced formal mathematical metrics on sensorimotor experience and its geometry, as well as heir use as part of a developmental architecture for robots that bases future action on previous experience (Nehaniv, 2005; Mirza et al., 2005a, 2007). In this paper we present results from a robotic experiment that illustrates how a history of embodied experience, combined with a metric measure for comparing experiences, can be used to predict temporally extended future experience. This is an important result for our developmental architecture as it demonstrates the efficacy of the metric measure, and in turn its suitability for directing future action and behaviour based on the individual's past experience.

Other Related Work. Olsson et al. (2006) use information distance to develop basic sensorimotor maps in interaction with the environment, beginning from raw uninterpreted sensors. Independently of our work, Oates et al. (2000) have also described experiences as a time-series of multi-variate sensorimotor data (which is essentially identical to our operational definition of experience), but computing distance

[1] This work was conducted within the EU Integrated Project RobotCub ("Robotic Open-architecture Technology for Cognition, Understanding, and Behaviours"), funded by the EC through the E5 Unit (Cognition) of FP6-IST under Contract FP6-004370.

between time-series and clustering experiences to produce prototypes. Experiences are associated with the actions that initiated them, so robot can generalize about potential outcomes of its actions. Distances between experiences are calculated by using Dynamic Time Warping followed by measuring the area between the curves, and clusters formed by taking averages of time-warped experience curves. In contrast, our framework uses an information-theoretic metric on such experiences.

Kaplan and Hafner (2005) use information distances between sensors in an Aibo robot to compare simple behaviours of the robot. In that method, rather than reducing the dimension by summation within groups as we have done, they consider distances between different behaviours as distances between the full matrix of distances between all sensors. Long continuous examples of each behaviour (1000 timesteps) are used, and the whole sequence used rather than a moving window. The resulting distances between behaviours are shown as a projection onto a two-dimensional map, and they find that similar behaviours group together. This research supports the view that robot behaviour can be clustered using information relationships between sensor time-series. However, the incremental formulation of our approach allows us to propose a system that can be used for ontogeny, and the use of the experience metric allows for better comparison of past behaviour and experience.

Continuous Case-Based Reasoning (CCBR) (Ram and Santamaria, 1997) has many similarities to the approach described here. However, in our approach the information metric allows for a more robust comparison of sensorimotor details concentrating on the statistics of the particular time-series, and so better able to recognize regularities in time-series than a simple Euclidean metric. Also, the metric nature of the space is also able to recommend a number of increasingly distant matches (neighbours) and is able to weight their similarity along with a qualitative value from the environmental feedback to provide, potentially, more appropriate actions.

Sensorimotor Experience and Metric

A robot or other embodied agent's entire view of the world is experienced through its sensors, including those that measure internal factors such as temperature, actuator positions, and other more general internal variables. Any sensor can be modelled as a random variable $\mathcal{X}$ changing with time, taking values $X(t) \in \mathcal{A}_{\mathcal{X}} = \{x_1, \ldots, x_m\}$ from a probability distribution $\mathcal{P}_{\mathcal{X}}$. Time is taken to be discrete (i.e. t will denote a natural number). A robot's experience, then, can be considered as the stream of all readings $(X^1(t), \ldots, X^n(t))$ from all these variables $\mathcal{X}^i$ over a given time period (i.e. $t \in [t', t'+h]$ for some *temporal horizon* $h > 0$). This is a purely operational sensorimotor view of experience and, by itself, says nothing about the quality or meaning of that experience.

Formally, an agent's *experience* from time t over a temporal horizon h can be defined as

$$E(t,h) = (\mathcal{X}^1_{t,h}, \ldots, \mathcal{X}^N_{t,h}) \tag{1}$$

where $\mathcal{X}^1_{t,h}, \ldots, \mathcal{X}^N_{t,h}$ is the set of random variables available to the agent constructed or estimated according to time-series of sensorimotor readings from N sensorimotor variables $(X^1, \ldots, X^N)$ ending at time t with a horizon h timesteps (from time $t-(h-1)$ to t).

Experience Metric

Given a definition of Sensorimotor Experience and the information metric, a formal measure of distance between experiences can be defined. This is useful as it allows a direct, scaled comparison between different sets of sensorimotor readings of a robot or agent. A metric for comparison of sensorimotor experiences is important as it is then possible to talk of proximity and distance between different experiences in a quantitative and geometrically meaningful way.

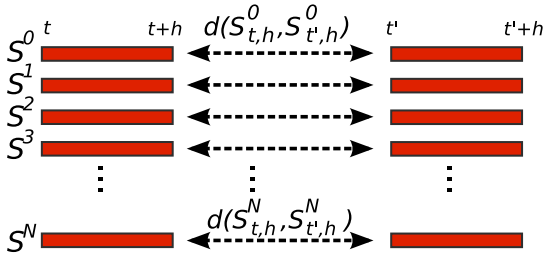


Figure 1: *Experience Metric*. A visual illustration of the experience metric. Each experience is shown as a collection of sensor readings of length h starting at time t and t'. The information distance between each respective sensor over time is summed to give the Experience Metric.

We define the *Experience Metric*, a metric on experiences of temporal horizon h, as

$$D(E,E') = \sum_{k=1}^{N} d(\mathcal{X}^k_{t,h}, \mathcal{X}^k_{t',h}) \tag{2}$$

where $E = E(t,h)$ and $E' = E(t',h)$ are experiences of an agent at time t and t' over horizon h, and d is the Crutchfield-Rényi information metric (Crutchfield, 1990), or more simply, the *information distance* between jointly distributed random variables. That is, $d(\mathcal{X},\mathcal{Y}) = H(\mathcal{X},\mathcal{Y}) - I(\mathcal{X},\mathcal{Y})$, where H denotes entropy and I denotes mutual information (see (Cover and Thomas, 1991) for an introduction to these concepts of information theory)[2]. D is measured in *bits*; see also Figure 1. That D is a metric follows from the fact that the metric axioms (equivalence, symmetry, and the triangle

[2] $d(\mathcal{X},\mathcal{Y}) = 2H(\mathcal{X},\mathcal{Y}) - H(\mathcal{X}) - H(\mathcal{Y})$ and is estimated directly from the frequency distributions of binned sensor values.

inequality) hold for each of the components in the summation, since d is a metric (Nehaniv, 2005). For a visual proof that d (and hence D) is a metric, see (Nehaniv et al., 2007).

Earlier Experiments

In Mirza et al. (2005b) we describe an experiment showing ball-path prediction using the experience distance measure. In that experiment an Aibo robot (see Figure 2 and below) remained stationary while a ball was moved in view of its head mounted camera. The predicted ball path was plotted in real-time overlaid on the images from the camera. This experiment illustrated that sensor experience can be used to match experience successfully. This experiment builds on that result, but uses the full embodied experience to match previous experience. The camera images do not, by themselves, give information about the position of the ball so self-experience is important.

Experiment

Interactive Path Prediction

A simple robotic experiment was devised that would illustrate how an artificial embodied agent can use its own history of experience combined with the experience metric described above to predict future experience. The robot follows the motion of a ball moved in front of it by using a simple reactive behaviour to adjust its head motors to attempt to centre the ball in its field of vision. The robot continually builds a metric space of experiences from its ongoing sensorimotor experience, including its own proprioceptive sense of movement arising through interaction with the environment. A closest historical experience, in terms of experience distance, to the current one is then found. Experiences temporally following the historically closest experience then provide a model for anticipation of future experience. How good this model is depends on both the predictability and consistency of the environmental interaction as well as how "good" the historical matching is. Thus, the analysis of the experiment focuses on measuring how well matched the historical experience is to the current one. Note that predicting the trajectory of the tracked object corresponds to prospection regarding part of a future temporally extended interval of sensorimotor experience.

It is important to note that, the robot is not matching current ball position with previous ball position, rather all sensory and motor variables are used as information sources to detect similarity between experiences.

Implementation and Experimental Setup

The robot used was a Sony Aibo ERS-7. The control and sensory collection software was implemented in Java with URBI (Baillie, 2005) providing the robot control layer and ball detection. Sensor readings are sent over wireless to a personal computer approximately every 80-120ms. Reception of each frame of data defines a *timestep*. Video images were received from the robot head camera approximately every 400ms, however visual sensors were computed at the rate of the sensor data using the most recent image from the camera. Experiences were formed from data streams from 33 internal sensors (including proprioceptive motor positions and infrared distance measurements, and 9 sensors formed from average pixel values in a 3×3 grid over the image.



Figure 2: *Sony Aibo ERS-7, and Pink Ball*

The robot was stationary in a "sitting" position, with the head pointed forward (Figure 2). A pink ball was moved in the air in view of the robot's head camera at a distance of approximately 30cm. No particular effort was made to "sanitize" the environment to aid ball-detection against the background. Thus, it is likely that other items in the environment provided potentially useful information about any interaction. The robot executes a continuous reactive behaviour to follow the motion of a ball with its head. The algorithm is simple, making appropriate incremental adjustments to the neck, headTilt and headPan motors, such that the position of the ball is brought closer to the centre.

The metric space creation and prediction was implemented in Java and ran on-line in real-time. The horizon length of the experiences was $h = 20$ timesteps or approximately 1700ms. The data was quantized into $Q = 10$ bins in the probability distribution estimation algorithm.

The ball was moved such that the time for the ball to describe a circle (or to move horizontally or vertically for a complete cycle) was 6-7 seconds. Thus the horizon length was shorter than, but of the same order of magnitude as, a single cycle of the repeated behaviour and the experiences would comprise approximately a half of a cycle.

The full interaction sequence lasted 965 timesteps (~ 84 seconds) constituting 945 experiences of horizon length $h = 20$. The movements of the ball consisted of a number of horizontal and vertical movements, and a number of clockwise circles; see Table 1.

Visualizing Ball Path: A projection of the current ball position relative to the robot is plotted in two dimensions by estimating the direction in which the head is pointed from

Table 1: Path Prediction Experiment - Sequences of Movements (TS denotes time step number)

Start TS	End TS	Movement Type	Iterations
91	185	Horizontal, Left to Right	2 full
201	272	Vertical movements, Top to Bottom	2 full
283	361	Horizontal, Right to Left	1 full
376	453	Vertical, Top to Bottom	2 full
463	534	Horizontal, Right to Left	1 full
548	593	Vertical, Top to Bottom	1 full
607	852	Circular, Clockwise	4 full
866	929	Vertical, Bottom to Top	2 full

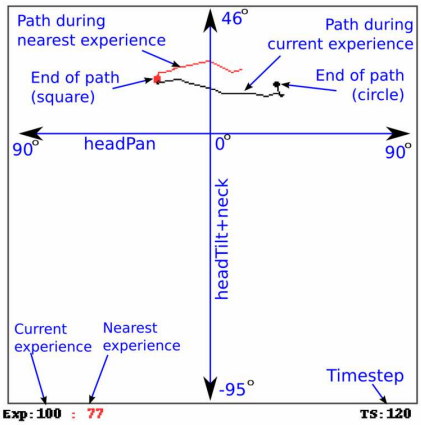


Figure 3: *Ball Path Traces*. The diagram shows the parts of the ball path diagrams used to visually analyse the traces of the ball in a neck-centred coordinate system derived from motor positions. See Figures 6 and 7.

the positions of three motors contributing to head motion. The coordinates for the ball position in the plot are given by:

$$(x, y) = (W \times headPan, H \times (headTilt + neck)/2)$$

where W and H are the image width and height, and $headPan, headTilt$ and $neck$ are the motor values at any instant normalized into the range $(0, 1)$. See the explanatory diagram of Figure 3. Note that the plots are created for analysis of the experiments, and this abstraction of the sensoriomotor flow is *not* available to the robot. Instead it allows us as external observers to gain insight into what the robot 'expects' will happen in an interval of the near future based on its own previous experiences, and how accurate these expectations are (again to an external observer).

Error Measurements: Two different measurements of path error were used. The first measured the sum of the Euclidean distance between each corresponding point of the paths. The second calculated a vector direction for each path and returned the angular difference in radians between the vectors as the error.

Table 2: Improvement of Experience Matching Over Time

Type	Iteration	Number $< \pi/4$	Total Number	Percentage $< \pi/4$
HORIZ	1	0	41	0.0%
HORIZ	2	27	73	37.0%
HORIZ	3	25	75	33.3%
HORIZ	4	27	72	37.5%
VERT	1	0	34	0.0%
VERT	2	8	51	15.7%
VERT	3	15	30	50.0%
VERT	4	42	61	68.9%
VERT	5	32	52	61.5%
VERT	6	27	49	55.1%
CIRCLE	1	9	65	13.8%
CIRCLE	2	13	54	24.1%
CIRCLE	3	27	66	40.9%
CIRCLE	4	31	63	49.2%

Results and Analysis

Figures 4 and 5 show, using different methods of error estimation, the error between the current path and the path corresponding to the nearest previous experience in terms of information distance. Figures 6 and 7 show traces of the paths from experiences in regions where horizontal and vertical movements were taking place. As can be seen from the traces, which are selected from regular intervals, it is often the case that the paths are similar and so the experiences are well matched. However, the objective measure of error indicates that the actual path is not exactly the same. This is to be expected as there do not exist any *precisely* identical experiences in a real situation.

The opposite direction path (but of the same type) is regularly matched. As the sensors are not biased left or right, and the experience distance measure is the sum of information distances between variables, then a symmetric error such as this is likely. Indeed, such experiences are *informationally* very close to their 'opposites'. Out-of-phase periodic variables can have a small or zero[3] information distance.

In terms of angle, the error is less than $\pi/4$ (*i.e.* closer to parallel than orthogonal) 55.13% of the time and is greater

[3]Variables that have a zero information distance are *recoding equivalent* and are not necessarily identical (see Crutchfield, 1990).

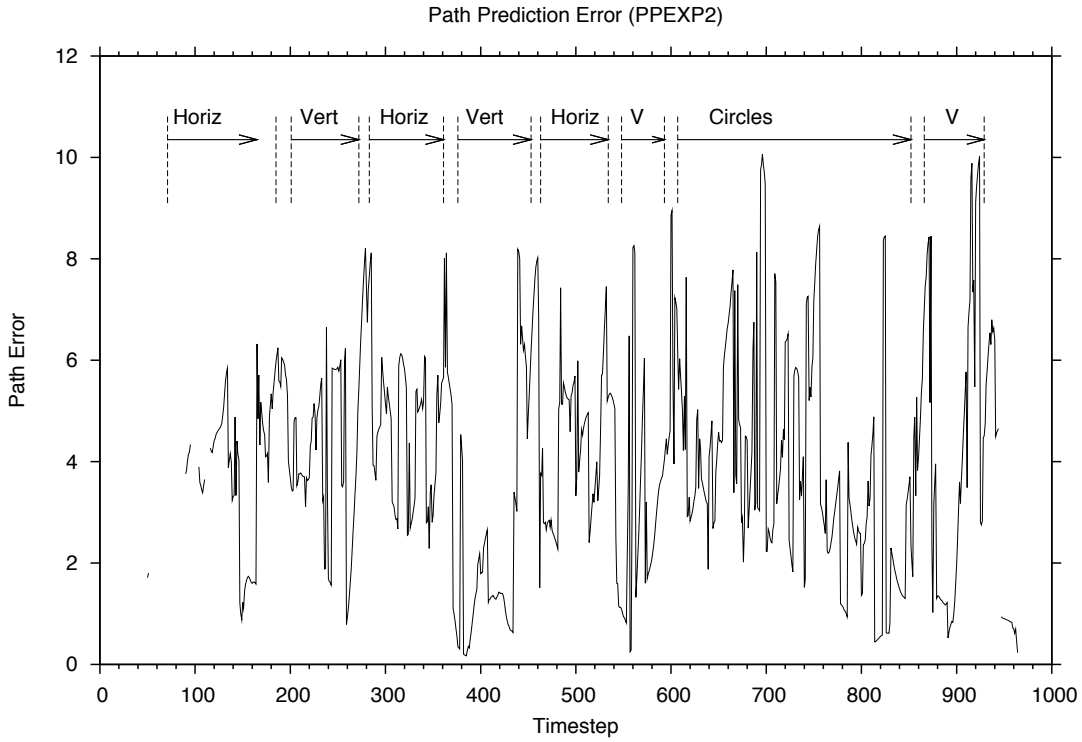


Figure 4: Euclidean distance (error) between the paths of the ball during the current and nearest previous experience. The error is often exaggerated as experiences of paths of the same type but opposite direction are often matched. The top part of the graph shows the behaviour (See Table 1). The *Path Error* (pixels) in this case is the sum of the Euclidean distance between corresponding points. Temporal horizon $h = 20$, number of bins $Q = 5$.

than $3\pi/2$ (*i.e.* closer to opposite than orthogonal) 29.21% of the time. This indicates that the path and therefore the experience is generally well matched, however due to the nature of the measure, experiences from the opposite phase in a cycle are often selected. This error is compensated for in Figure 5 by reflection about $\pi/2$. It is interesting to note the opposite phase corresponds to time-reversed motion, and that the present metric relies on probability distributions constructed from sensorimotor flow and that these distributions do not encode the directionality of time.

Examining the progression of the error over time in these data, one would expect to see an improvement as the same kinds of behavioural interaction are re-experienced. How the matching of experiences improves over time is examined, referring to Table 2 and Figure 5. During the horizontal motions after one full cycle, 37% of experiences can be matched to similar ones in the history. Vertical motions show that the success rate peaks at 68.9% with the 4th presentation. The success rate drops slightly thereafter as there are more experiences to select from. The Circle movements also show marked improvement as experience grows. The initial 13.8% success rate of the very first circular motion reflects the fact that parts of the circular motion are being matched with previous horizontal and vertical experiences, with some limited success, even before any such motions had been observed.

Conclusions

The work describing the construction and use of information metrics for the comparison of robot behaviour demonstrates achievement of a degree of temporally extended prospection by an embodied agent, based on its raw sensorimotor experience. The experience metric was first described in (Mirza et al., 2005a) and with mathematical proofs of the mathematical metric properties along with some alternative metrics on experience in (Nehaniv, 2005). As mentioned, an operational formulation of experience (but not of the metric) was previously described in (Oates et al., 2000). A non-metric measure of distance between experiences was described there that used the area between time-warped experience curves. The fact that independent research groups both developed essentially the same notion operationalizing an agent-centred definition of experience suggests that this definition is a natural one.

Experiments were described that use fairly large numbers of robotic sensors to describe robotic experience such that a simple sort of prediction can be achieved by the matching of present experience with experiences in the history and

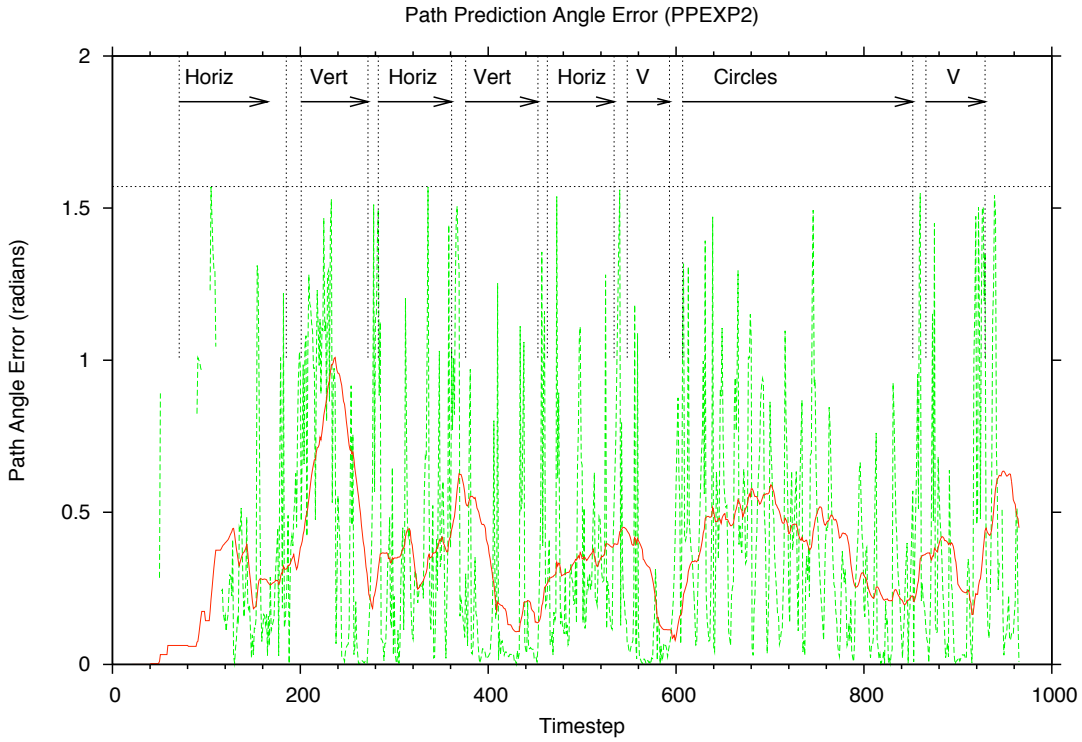


Figure 5: Angle error and the average angle error (over the last 40 timesteps) between the paths of the ball during the current and nearest previous experience. The graph shows the error reducing, on average, *within* a given behaviour sequence. The top part of the graph shows the behaviour (See Table 1). The *angle error* is the difference in radians between the vector direction of each path. For errors$> \pi/2$, $\pi - error$ is shown (reflection about $\pi/2$). Temporal horizon $h = 20$, number of bins $Q = 5$.

extrapolating forward from the matched past experience. It was found that *proximity in terms of experience metric corresponds well with an external observer's notion of similarity of experience*. Future research may consider using the anticipated experience for active perception and in human-robot interaction.

The sensorimotor variables were treated by the autonomous robot in an uninterpreted "agnostic" manner, that is, no sensor is regarded as being different from any another or special in any way, in terms of finding close experiences. This performance was achieved despite many of the sensors not providing any seemingly useful information about the current experience. Proprioceptive motor experience was important in this experiment in determining the experience and matching it to the appropriate past experience.

The capability of the experience metric to find suitable matching experiences was found to increase as more examples of a particular type of behaviour were presented. This appears to level-off, and potentially become worse as more examples are presented. However, the experiments described had too short a run time for a definitive conclusion to be drawn on the latter observation. Another important aspect of the experience metric is that it appears to confuse a behaviour with its 'opposite' (phase-shifted or time-reversed counterparts), as these are informationally nearly identical. This can be seen clearly in both the simple and interactive ball-path prediction experiments as opposite direction of path.

Needless to say, the ontogeny of prospective ability of children and other mammals is an extended process lasting years and we cannot yet hope to mirror its complexity and success in artificial systems, although the work presented here suggests that we have made a small start in this direction.

References

Baillie, J.-C. (2005). URBI: Towards a universal robotic low-level programming language. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems 2005*. http://www.urbiforge.com.

Brooks, R. (1999). *Cambrian Intelligence: The Early History of the New AI*. MIT Press, Cambridge, Mass.

Clancey, W. J. (1997). *Situated Cognition: On human knowledge and computer representations*. Learning in doing: Cognitive and compuational perspectives. Cambridge University Press.

Cover, T. and Thomas, J. (1991). *Elements of Information Theory*. Wiley, New York.

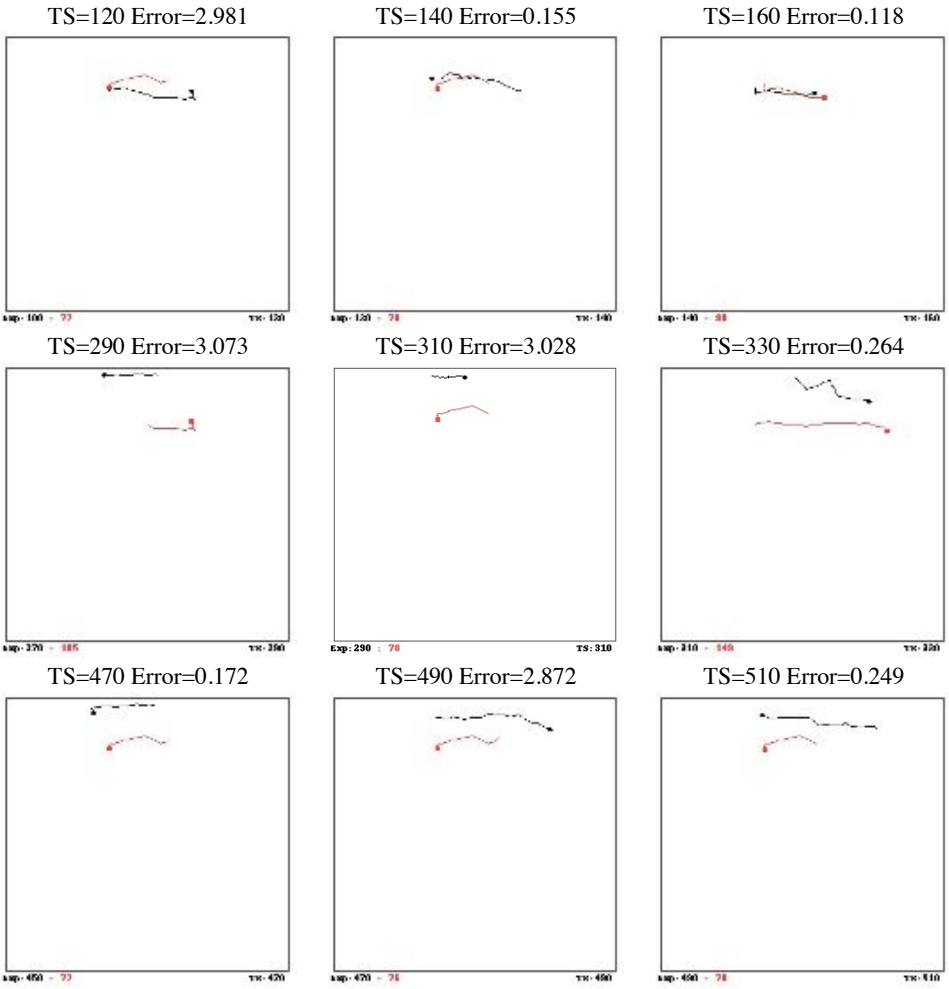


Figure 6: *Head Movement Traces and Matched Historical Traces for Prediction*. Images are from evenly spaced timesteps from three separate *horizontal* movement regions starting at timestep TS=120, 290 and 470. Each diagram shows the path of the ball, as determined by robot head movements, for both the current experience at that timestep (dark line) and for the matched (nearest previous) experience (red/grey line). Path direction indicated by circle/square at the end of the path. (See Figure 3). The angle error between the path directions is used to analyse how well the path and thus experience are matched. Temporal horizon $h = 20$, number of bins $Q = 5$.

Crutchfield, J. (1990). Information and its metric. In Lam, L. and Morris, H., editors, *Nonlinear Structures in Physical Systems - Pattern Formation, Chaos and Waves*, pages 119–130. Springer-Verlag, New York.

Kaplan, F. and Hafner, V. (2005). Mapping the space of skills: An approach for comparing embodied sensorimotor organizations. In *Proc. 4th IEEE International Conference on Development and Learning (ICDL-05)*, pages 129–134. IEEE.

Maturana, H. R. and Varela, F. J. (1987). *The Tree of Knowledge: The Biological Roots of Human Understanding*. New Science Library (Shambhala), Boston.

Mirza, N. A., Nehaniv, C. L., Dautenhahn, K., and te Boekhorst, R. (2005a). Using temporal information distance to locate sensorimotor experience in a metric space. In *Proc. 2005 IEEE Congress on Evolutionary Computation*, volume 1, pages 150–157, Edinburgh, Scotland. IEEE Press.

Mirza, N. A., Nehaniv, C. L., Dautenhahn, K., and te Boekhorst, R. (2007). Grounded sensorimotor interaction histories in an information theoretic metric space for robot ontogeny. *Adaptive Behaviour*, 15(2):167–187.

Mirza, N. A., Nehaniv, C. L., te Boekhorst, R., and Dautenhahn, K. (2005b). Robot self-characterisation of experience using trajectories in sensory-motor phase space. In *Proc. 5th International Workshop on Epigenetic Robotics*. Lund University Cognitive Studies.

Nehaniv, C. L. (2005). Sensorimotor experience and its metrics. In *Proc. 2005 IEEE Congress on Evolutionary Computation*, volume 1, pages 142–149, Edinburgh, Scotland. IEEE Press.

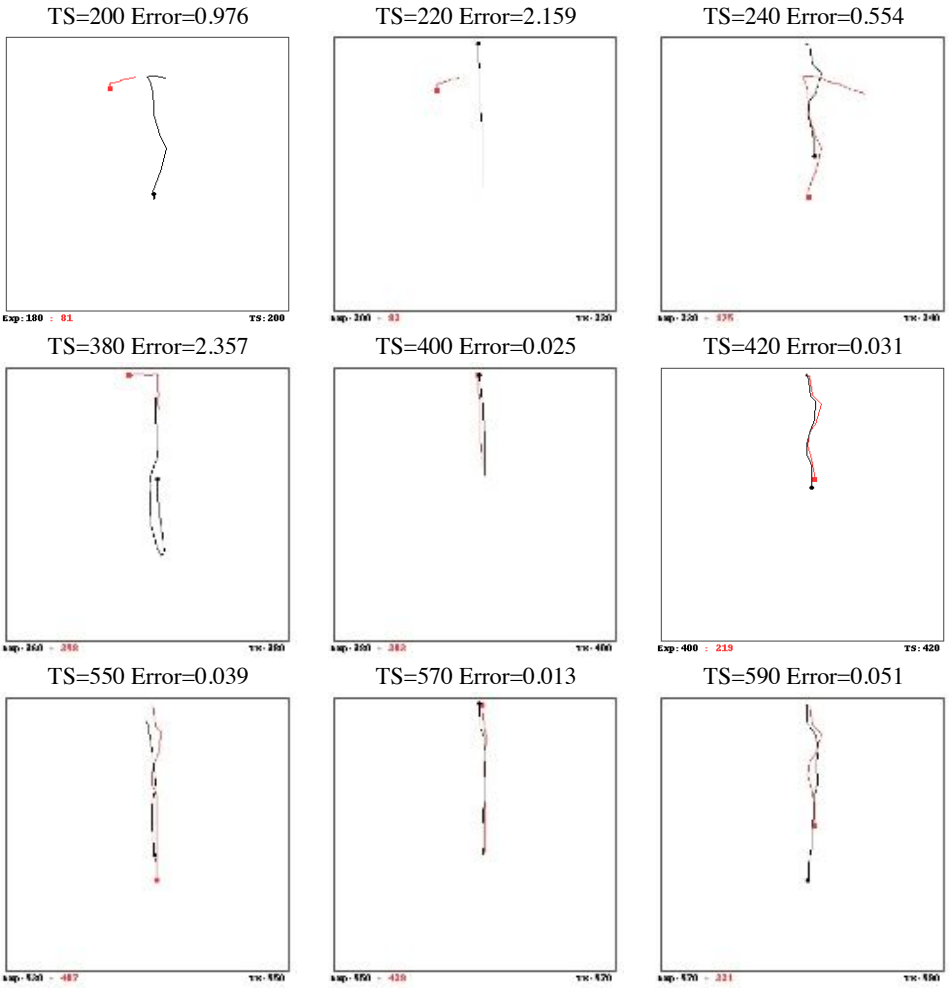


Figure 7: *Head Movement Traces and Matched Historical Traces for Prediction*. Images are from evenly spaced timesteps from three separate *vertical* movement regions starting at timestep TS=200, 380 and 550. Each diagram shows the path of the ball, as determined by robot head movements, for both the current experience at that timestep (dark line) and for the matched (nearest previous) experience (grey line). Path direction indicated by circle/square at the end of the path. (See Figure3). The angle error between the path directions is used to analyse how well the path and thus experience are matched. Temporal horizon $h = 20$, number of bins $Q = 5$.

Nehaniv, C. L., Mirza, N. A., and Olsson, L. (2007). Development via information self-structuring of sensorimotor experience and interaction. In *50 Years of Artificial Intelligence*, volume 4850 of *LNCS*, pages 87–98. Springer.

Nehaniv, C. L., Polani, D., Dautenhahn, K., te Boekhorst, R., and Cañamero, L. (2002). Meaningful information, sensor evolution, and the temporal horizon of embodied organisms. In *Artificial Life VIII*, pages 345–349. MIT Press.

Oates, T., Schmill, M. D., and Cohen, P. R. (2000). A method for clustering the experiences of a mobile robot that accords with human judgements. In *Proceedings of the Seventeenth International Conference on Artificial Intelligence*, pages 846–851, Menlo Park, California. AAAI Press.

Olsson, L., Nehaniv, C. L., and Polani, D. (2006). From unknown sensors and actuators to actions grounded in sensorimotor perceptions. *Connection Science*, 18(2):121–144.

Pfeifer, R. and Bongard, J. (2007). *How the Body Shapes the Way We Think: A New View of Intelligence*. MIT Press.

Ram, A. and Santamaria, J. C. (1997). Continuous case-based reasoning. *Artificial Intelligence*, 90(1-2):25–77.

Varela, F. J., Thompson, E., and Rosch, E. (1991). *The embodied mind: Cognitive science and human experience*. MIT Press, Cambridge, MA, USA.

Vernon, D., Metta, G., and Sandini, G. (2007). The icub cognitive architecture: Interactive development in a humanoid robot. In *IEEE International Conference on Development and Learning, Imperial College, London*.

Common Concepts in Agent Groups, Symmetries, and Conformity in a Simple Environment

Marco Möller[1,2] and Daniel Polani[2]

[1]Theory of Complex Systems Group, Institute of Solid State Physics, Technical University of Darmstadt, Germany
[2]Adaptive Systems Group, School of Computer Science, University of Hertfordshire, UK
marco.moeller@juforum.de*, d.polani@herts.ac.uk

Abstract

We analyze representations of the world attained through an infomax principle by agents acting in a simple environment. The representations obtained by different agents in general differ to some extent from each other in different instances. This gives rise to ambiguities in how the environment is represented by the different agents. We now develop an information-theoretic formalism able to extract a "common conceptualization" of the world for a group of agents. It turns out that the common conceptualization intuitively seems to capture much higher regularities or symmetries of the environment than the individual representations.

We formalize the notion of identifying symmetries in the environment - with respect to "extrinsic" operations on the environment as well as with respect to "intrinsic" operations, i.e. the reconfiguration of the agent's embodiment. In particular, using the latter formalism, we can re-wire an agent to conform to the highly symmetric common conceptualization to a much higher degree than an unrefined agent; and that without having to re-optimize the agent from scratch. In other words, we can "re-educate" an agent to conform to the de-individualized "concept" of the agent group with comparatively little effort.

Motivation

In the search of how agents aim to model their environment, there is a huge collection of candidates. However, it has been suspected earlier that, whatever the detailed mechanism would entail, they might follow principles of information parsimony or optimal information processing (Barlow (1959); Laughlin (2001)). A concrete model for maximum Shannon information processing has been proposed in the infomax model by Linsker (1988).

We are interested in how agents can model their environment based on informational considerations. Using infomax principles to do that, one obtains a classification or representation of a given environment (in the following also called concept) for a given agent. We use the perception-action (PAL) loop from Klyubin et al. (2007) to model the agent and its interaction with the environment, i.e. the model and according tasks for the agent are not part of this work.

In general, the representations of the environment developed in an infomax process differ w.r.t. the agent. Even in very simple and highly symmetric scenarios, they can considerably vary from agent to agent as a result of the infomax optimization i.e. different global and (good) local optima can be returned. This is similar to a biological evolution optimization process: the individuals also vary to some extend from each other. This raises the issue of how similar the obtained concepts are. We will discuss what the different concepts of those agents have in common. Is it possible to develop a concept which is mutually compatible to each of these input concepts (see e.g. Philipona and O'Regan 2006; Steels 1997; i Cancho and Solé 2003)? If so, what properties of the environment or the agents do such common concepts capture? How do they relate to the individual agents' concepts?

We will not model how agents agree on a common concept or how they communicate but we will discuss some information-theoretical criteria for such a common concept. In general, we are not interested in processes but in their outcome. We do not analyze mechanisms but the underlying principles.

Analyzing the quality of concepts with respect to certain goals, we observed that "good" concepts have more regularities. That led us to analyze the concepts' symmetries. In general, they are not symmetric in a strict mathematical way. So we needed a method to measure also not perfectly fulfilled symmetries. We developed an information-theoretical approach to analyze these "weak" symmetries. One hypothesis is that common concepts will reveal symmetries of the whole agent/environment system that are broken by the individual concepts. We can now ask under which conditions the individual agents can relate to this expected higher regularity of the common concept. In a second approach of analyzing these symmetries, we study the influence of the agents embodiment on the agent and try to find a way of "asking the agent what he considers to be a symmetry of the environment".

The technical challenges arising from these issues are manifold. We aim to find a description that is consistent with a fundamentally information-theoretical picture of the agents and their environment. For this, one needs to suitably

formulate the development of a common concept of a set of agents. Also, one needs to model the concept of regularity or symmetry in a suitable way.

The contributions of this paper are information-theoretic techniques to construct common concepts for a group of agents and to evaluate weak symmetries, and their application to some simple, but informative scenarios.

Background

To be able to introduce our model for the agents and their interaction with the world, we have to introduce some notations and quantities first. Consider random variables $X, Y, Z, \ldots$ denoted by capital letters which take in values $x, y, z, \ldots$ in corresponding sets $\mathcal{X}, \mathcal{Y}, \mathcal{Z}, \ldots$. For the probability that a given random variable X assumes a value $x \in \mathcal{X}$ we write $\Pr(X = x)$ or, if it is clear from context just $p(x)$. For the probability for the joint variable $(X_1, \ldots, X_n)$ we will write simply $\Pr(X_1 = x_1, \ldots, X_n = x_n) \equiv p(x_1, \ldots, x_n)$. The (*Shannon*) *entropy* of a random variable X is given by

$$H(X) := -\sum_{x \in \mathcal{X}} p(x) \log p(x) \tag{1}$$

whereby the logarithm in this paper is always to the basis of 2, so the unit for entropy is the *bit*. The *conditional entropy* of X given Y is given by $H(X|Y) := H(X,Y) - H(Y)$ and the *mutual information* between X and Y by

$$I(X;Y) := H(X) + H(Y) - H(X,Y). \tag{2}$$

A generalization of mutual information is the *multiinformation* between a collection of random variables $X_1, \ldots, X_n$

$$I(X_1; \ldots; X_n) := \left[\sum_{i=1}^{n} H(X_i)\right] - H(X_1, \ldots, X_n) \tag{3}$$

its conditional form, if the random variable Y is observed is

$$I(X_1;\ldots;X_n|Y) := \left[\textstyle\sum_{i=1}^{n} H(X_i|Y)\right] - H(X_1,\ldots,X_n|Y). \tag{4}$$

To measure the "difference" between two random variables X, Y we can use the unnormalized version of the information distance (Crutchfield (1990))

$$D(X,Y) := H(X|Y) + H(Y|X) \tag{5}$$

which fulfills the conditions for a metric including the triangle inequality. Note that D vanishes for a deterministic bijective dependency between X, Y.

To model agents in an environment we will use the formalism from Klyubin et al. (2007) based on *causal Bayesian network* (CBN). A CBN is given by a directed acyclic graph $\mathcal{G} = (\mathcal{N}, \mathcal{E})$ whose nodes $n \in \mathcal{N}$ are representing random variables X_n and the edges $e \in \mathcal{E} \subseteq \mathcal{N} \times \mathcal{N}$ causal conditional probability dependencies between them. The distribution of X_n is given by $p\left(x_n | x_{\mathrm{Pa}(n)}\right)$ whereby $\mathrm{Pa}(n) := \{n' \in \mathcal{N} | (n', n) \in \mathcal{E}\}$ is the set of parent nodes n' from node n. If a node n has no parent nodes $\mathrm{Pa}(n) = \emptyset$, we identify $p\left(x_n | x_{\mathrm{Pa}(n)}\right) \equiv p(x_n)$ with an unconditional probability distribution. The joint distribution of the whole network is given by

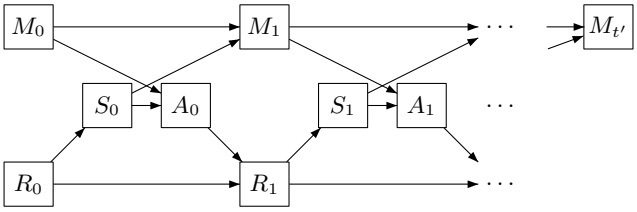


Figure 1: Perception-action loop unrolled in time as a CBN

$$p\left(x_1, \ldots, x_{|\mathcal{N}|}\right) = \prod_{n \in \mathcal{N}} p\left(x_n | x_{\mathrm{Pa}(n)}\right). \tag{6}$$

Model

A generic model for an *agent* interacting with a world is the *perception-action loop* (PAL). It is here only briefly presented, for a full presentation and motivation see Klyubin et al. (2007). Such an agent can *sense* the *world* R through its *sensor* S and *manipulate* it through its *actuator* A which together form the *embodiment* of the agent. This process can be formalized by the CBN shown in Fig. 1. All random variables depend on the time t: M_t, A_t, R_t, S_t. More precisely the *controller* of the agent has the possibility to store information in the *memory* M. It can be described by a probabilistic mapping

$$\text{controller} : M_t \times S_t \to M_{t+1} \times A_t \tag{7}$$

which is time t independent.

In our experiments, we chose a deterministic controller (Wennekers and Ay (2005); Klyubin et al. (2007)) and used a two dimensional infinite grid-world $\mathcal{R} = \mathbb{Z}^2$. The memory M is a number contained in a finite subset of $\mathcal{M} \subset \mathbb{N}$. The initial memory M_0 is deterministically set to a default state 0. The initial position in the world R_0 is uniformly distributed over possible starting positions $\mathcal{R}_0 = \{-d, \ldots, d\}^2$ where the *radius* d depends on the experiment. The actuator A can take on values $\mathcal{A} = \{\downarrow, \leftarrow, \uparrow, \rightarrow\}$ where these 4 actions can move the agent (changing its position in the world, encoded in R) to one of its 4 adjacent positions in the grid-world. The first discussed sensor (*setup s+*) has 4 possible sensor values $\mathcal{S} = \{\downarrow, \leftarrow, \uparrow, \rightarrow\}$. If we imagine a "pheromone" gradient emitted by a source at the origin (Fig. 2 - center), this sensor points to the adjacent position with the highest concentration of pheromone. If this is not unique (e.g. at the origin), one direction is randomly chosen. Setup s+ is visualized in the left of Fig. 2, whereby for each position $(x, y) \in \mathcal{R}$ all possible sensor "directions" are shown with their arrow-length corresponding to their probability. A variation of this setup used in this work is a sensor (*setup sq*) where 4 of such sources exists at $\{-5, 5\}^2$

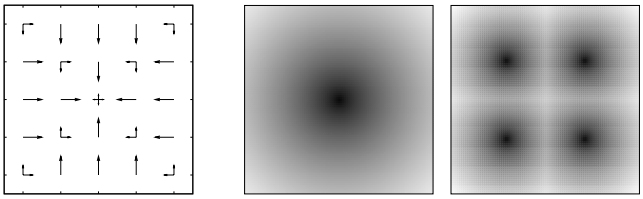


Figure 2: Setups

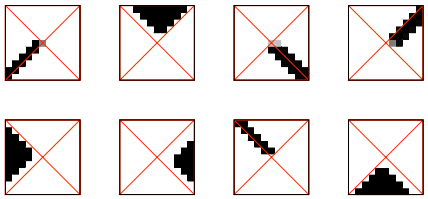


Figure 3: Solution of initial position capturing

(Fig. 2 - right) and the sensor is pointing always to the nearest source.

Our fundamental task for the agent is to capture as much information about its initial position as possible by its "final" memory state at time[1] $t = 15$ as suggested by Klyubin et al. (2007). This can be denoted information-theoretically as maximizing

$$I\left(R_0; M_{15}\right). \tag{8}$$

The search space for this problem contains all possible controller mappings from Eq. 7. To solve this and all following optimization problems, we used *Simulated Annealing* with some heuristic improvements described elsewhere but such tasks can be performed by any generic optimization tool. We do not aim to model the details of the process of agent evolution / adaptation and its ability to capture the information about the initial position but only the outcome of such a process. This output corresponds to the solutions returned by Simulated Annealing.

Common Concepts

Concepts

Consider an agent with setup s+ and memory size $|\mathcal{M}| = 8$ who is able to capture the initial position R_0 (what is uniformly distributed with $\mathcal{R}_0 = \{-5, \ldots, 5\}^2$) by maximizing $I\left(R_0; M_{15}\right)$. Therefore an appropriate controller has to be found. To interpret this agent, consider Fig. 3 where each of the 8 squares shows in gray scale the conditional probability $p\left(r_0|m_{15}\right)$ for a final memory state. To make this precise, it shows the probability that this agent has been initially at position r_0 in the world for a memory content $m_{15} = 0, 1, 2, \ldots, 7$ at time $t = 15$ of the end of the run. For each state m_{15} we use a separate normalization so $\max_{r_0} p\left(r_0|m_{15}\right)$ is represented by black and $p\left(r_0|m_{15}\right) = 0$ by white. The agent shown has an utility value of $I\left(R_0; M_{15}\right) = 2.906$ bit which is very near to the limit of $\min\left[\log|\mathcal{R}_0|, \log|\mathcal{M}|\right] = 3$ bit. These 8 possible memory values m_{15} can be understood as a *concept* of the world R_0. Each value for m_{15} has a certain "meaning" for stating positions, like "north-triangle", "north-east-diagonal", "east-triangle", "south-east-diagonal", etc. We call a pair of random variables (R, Y) (e.g. (R_0, M_{15})) jointly distributed a *concept* if Y is "representing" R in some way, i.e. $I\left(R; Y\right) > 0$. We call the values $y \in \mathcal{Y}$ *symbols* of the concept.

As mentioned earlier there also exist other solutions for the problem to find a good initial position capturer with an equal or similar utility value $I\left(R_0; M_{15}\right)$, for example just an agent with a "rotation" of the concepts by 90°. This "rotation-symmetry" will be discussed later. Here we are interested in how representative the shown example concepts and how similar other solutions are. We will do this by discussing the possibilities to find a *common concept* (R, Y_*) from a set of *input concepts* $\left\{\left(R, Y^{(1)}\right), \ldots, \left(R, Y^{(n)}\right)\right\}$. This concept can be interpreted as common concept of a group of agents in a world R. In the spirit from above philosophy, we emphatically only model the information-theoretical principle. The process of agreeing of the individuals about the common concept is wittingly not modeled to be independent of the algorithm. We will present in the following two possibilities to define such a common concept.

For the Objective Common Concept consider the CBN from Fig. 4. A deterministic mapping $R \to Y_*^{obj}$ which maximizes

$$\sum_i \left[I\left(Y_*^{obj}; Y^{(i)}\right) - \alpha \cdot I\left(R; Y_*^{obj}\right)\right] \tag{9}$$

defines the *objective common concept* $\left(R, Y_*^{obj}\right)$. The first term $I\left(Y_*^{obj}; Y^{(i)}\right)$ maximizes the mutual information between the common concept and every input concept, so as to make it as similar as possible the input concepts. The term $\alpha \cdot I\left(R; Y_*^{obj}\right)$ is a bottleneck type (Tishby et al. (1999)) parameter $\alpha \in [0, 1]$ countering the trivial behavior of just building $Y_*^{obj} = Y^{(1)} \times \ldots \times Y^{(n)}$ as cross product of all input concepts if the number of states in Y_*^{obj} is sufficiently large $\left|\mathcal{Y}_*^{obj}\right| \geq \prod_i \left|\mathcal{Y}^{(i)}\right|$. For our experiments we set $\alpha = 0.2$. This method is called objective because it has explicit knowledge about the world R.

For the Subjective Common Concept consider the CBN from Fig. 5. A deterministic mapping $Y^{(1)} \times \ldots \times Y^{(n)} \to Y_*^{subj}$ which minimizes

$$I\left(Y^{(1)}; \ldots; Y^{(n)} | Y_*^{subj}\right) \tag{10}$$

[1] $t = 15$ is an arbitrary choice for our experiments, other choices lead to similar results.

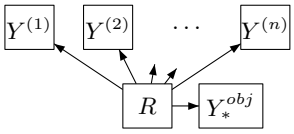


Figure 4: CBN for objective common concept

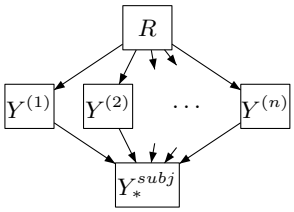


Figure 5: CBN for subjective common concept

defines the *subjective common concept* $\left(R, Y_*^{subj}\right)$ by applying the rules for the joint distribution of a CBN. The minimization makes sure that Y_*^{subj} "absorbs" all information common by $Y^{(1)}, \ldots, Y^{(n)}$. This method is called subjective because it has only implicit knowledge about R through the input concepts.

Results Common Concept

A Comparison of Objective and Subjective Common Concept is calculated for the 4 input concepts shown in Fig. 6. We see in each of the 4 columns one concept $\left(R_0, M_{15}^{(i)}\right)$ generated by initial position capturing agents with setup s+ and $|\mathcal{M}| = 6$. Figure 7 shows an objective and subjective common concept (R_0, M_*) of size $|\mathcal{M}_*| = 8$ each. The superstition of the subjective common concept is with $H\left(M_*|R_0\right) = 0.03$ bit vanishingly small[2]. The information distance between objective and subjective common concept is with $D\left(M_*^{obj}, M_*^{subj}\right) = 0.38$ bit also quite small. The only significant difference is that the symbol for "south-east" is split in the subjective method, therefore it has no symbol for "north-west". Because of their similarity we will not continue to calculate both common concepts. Especially if we consider the computational complexity we will, in further investigations, only use the objective common concept. For the subjective common concept the computational complexity, and the size of the search space are growing exponentially with the size of the input concept $\left|\mathcal{M}_{15}^{(i)}\right|$ and their number n. Some further objective common concepts are shown in Fig. 11.

For lack of space the preferred common concept size will not discussed here.

[2]The superstition of the objective common concept is 0 by definition because M_* deterministically depends on R_0.

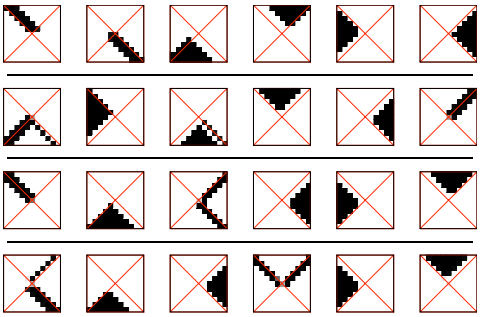


Figure 6: 4 input concepts of size $|\mathcal{M}| = 6$ for Fig. 7

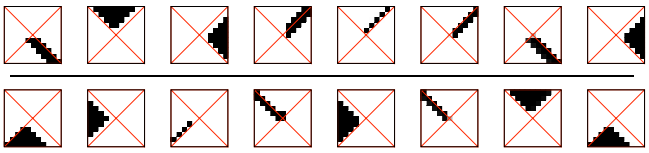


Figure 7: Objective (upper half) and subjective (lower half) common concept

Symmetry

As mentioned earlier, all (common) concepts exhibit a large degree of symmetry. We will present two methods to measure and analyze these symmetries. Common to both methods is the idea of transforming concepts and comparing them by measuring the mutual information between the transformed concept and e.g. the original. The *extrinsic symmetry* transforms the concept by applying a combination of a rotation, mirroring and translation on the world. So it tests if some explicitly known symmetries of the world also hold for the concept. The *intrinsic symmetry* opposed searches for invariants of the world from the agents perspective. So it is able to extract what seems to be a symmetry for the agent.

With these methods we developed a framework for analyzing the role of regularities in agent↔world interaction and especially what kind of regularities are used by agents in their interaction with the world.

Extrinsic Symmetry

An *extrinsic symmetry* operating on a concept (R, Y) transforms the grid-world $\mathcal{R} = \mathbb{Z}^2$ by applying an *extrinsic symmetry operation* $\xi^{\theta,\varphi,x_0,y_0}$, a combination of a rotation φ (in 90° steps), mirroring θ, and translation (x_0, y_0)

$$\xi^{\theta,\varphi,x_0,y_0} : \mathbb{Z}^2 \to \mathbb{Z}^2 \tag{11}$$

$$\xi^{\theta,\varphi,x_0,y_0} := \xi_{\text{trans}}^{x_0,y_0} \circ \xi_{\text{rot}}^{\varphi} \circ \xi_{\text{mir}}^{\theta}. \tag{12}$$

The mirroring (at the y-axis) is described by $\theta \in \{+1, -1\}$

$$\xi_{\text{mir}}^{+1}(x, y) := (x, y) \qquad \xi_{\text{mir}}^{-1}(x, y) := (-x, y),$$

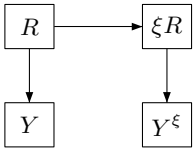


Figure 8: CBN for calculating extrinsic symmetry utility

the rotation (in 90° steps counterclockwise) by $\varphi \in \{0°, 90°, 180°, 270°\}$

$$\xi_{\text{rot}}^{0^\circ}(x,y) := (x,y) \qquad \xi_{\text{rot}}^{90^\circ}(x,y) := (-y,x)$$
$$\xi_{\text{rot}}^{180^\circ}(x,y) := (-x,-y) \qquad \xi_{\text{rot}}^{270^\circ}(x,y) := (y,-x)$$

and finally the translation by $(x_0, y_0) \in \mathbb{Z}^2$

$$\xi_{\text{trans}}^{x_0,y_0}(x,y) := (x + x_0, y + y_0).$$

The application of the operation $\xi^{\varphi,\theta,x_0,y_0}$ transforms the world R and gives us two new probabilistic mappings $R \to \xi R$ and $\xi R \to Y^{\xi}$ (Fig. 8). The first mapping applies the operation $\xi^{\varphi,\theta,x_0,y_0}$ on R by

$$\Pr(\xi R = r' | R = r) := \delta_{r',\xi(r)} \tag{13}$$
$$= \begin{cases} 1 & r' = \xi(r) \\ 0 & \text{else} \end{cases}. \tag{14}$$

The second mapping $\xi R \to Y^{\xi}$ is chosen as a "copy" of $R \to Y$:

$$\Pr\left(Y^{\xi} = y | \xi R = r\right) := \Pr(Y = y | R = r). \tag{15}$$

We define the *utility for extrinsic symmetry operation* $\xi^{\varphi,\theta,x_0,y_0}$ for the concept (R, Y) as

$$I\left(Y; Y^{\xi}\right), \tag{16}$$

where a higher value means "higher symmetry".

Informally, this utility measures "how much a rotated/mirrored/translated concept has in common with the original one". Note that because of the use of information theory, possible symbol permutations are ignored. With this method we are also able to interpret a sensor mapping $R_t \to S_t$ as a concept and calculate its symmetries.

Intrinsic Symmetry

We define a *permuted embodiment* for an agent as shown in Fig. 9. In comparison to Fig. 1, the original sensor S is replaced by $S^{\pi} \to S^{orig}$ and the original actuator A by $A^{orig} \to A^{\pi}$. Each pair of permutation of sensor an actuator (π_S, π_A) with

$$\pi_S : S^{\pi} \to S^{orig} \tag{17}$$
$$\pi_A : A^{orig} \to A^{\pi}. \tag{18}$$

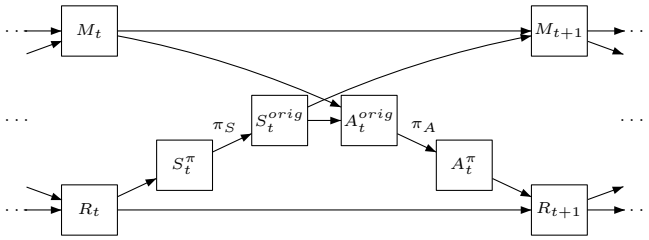


Figure 9: Permuted embodiment for the perception-action loop as CBN

defines an *intrinsic symmetry operation*. To evaluate an intrinsic symmetry operation (π_S, π_A) on an initial position capturing agent we first generate a whole set of concepts $\left\{\left(R_0, M_{15}^{(1)}\right), \ldots, \left(R_0, M_{15}^{(n)}\right)\right\}$ from other good initial position capturing agents with an equivalent setup (both, evaluated agent and the other agents are still without the modifications from Fig. 9 at this point). For this set of concepts is an objective common concept $\left(R_0, M_*^{obj}\right)$ is computed. To evaluate a specific intrinsic symmetry operation (π_S, π_A), we apply it on the PAL[3] and then calculate the resulting concept (R_0, M_{15}^{π}). We define the quality of this operation as

$$I\left(M_*^{obj}; M_{15}^{\pi}\right), \tag{19}$$

where a higher value means "higher symmetry". Informally spoken, "we are shuffling perceptions and actions of the agent and investigating if he is still able to 'conform' the common concept".

Results Symmetry

Figure 11 shows because of lack of space in extremely compact format some of our symmetry results for comparison. The upper half of the figure is for setup s+, the lower half for setup sq.

The Extrinsic Symmetry of the Setup is shown in ① and ②. ① shows the used setup as map of possible sensor outcomes (understood as concept) $p(r|s)$. The marked box inside of each symbol denotes the places belonging to R_0. This region visualizes the area of the concept what will be compared with the transformed concept by the mutual information. For setup s+ is $\mathcal{R}_0 = (-5, \ldots, 5)^2$ and for setup sq $\mathcal{R}_0 = (-10, \ldots, 10)^2$. Of the translation operations, only those are tested which maps R_0 on a subset of the shown positions in the concept, consequently only translations are tested with $\max(|x_0|, |y_0|) \leq 5$ for setup s+ resp ≤ 10 for setup sq. The corresponding extrinsic *symmetry spectrum* in ② shows the number of symmetries (y-axis) for a given value of $x = \frac{I\left(R;Y^{transformed}\right)}{\max I(R;Y^{transformed})}$. Additional to these

[3]Without changing on the controller, i.e. not optimizing the utility from Eq. 8 again.

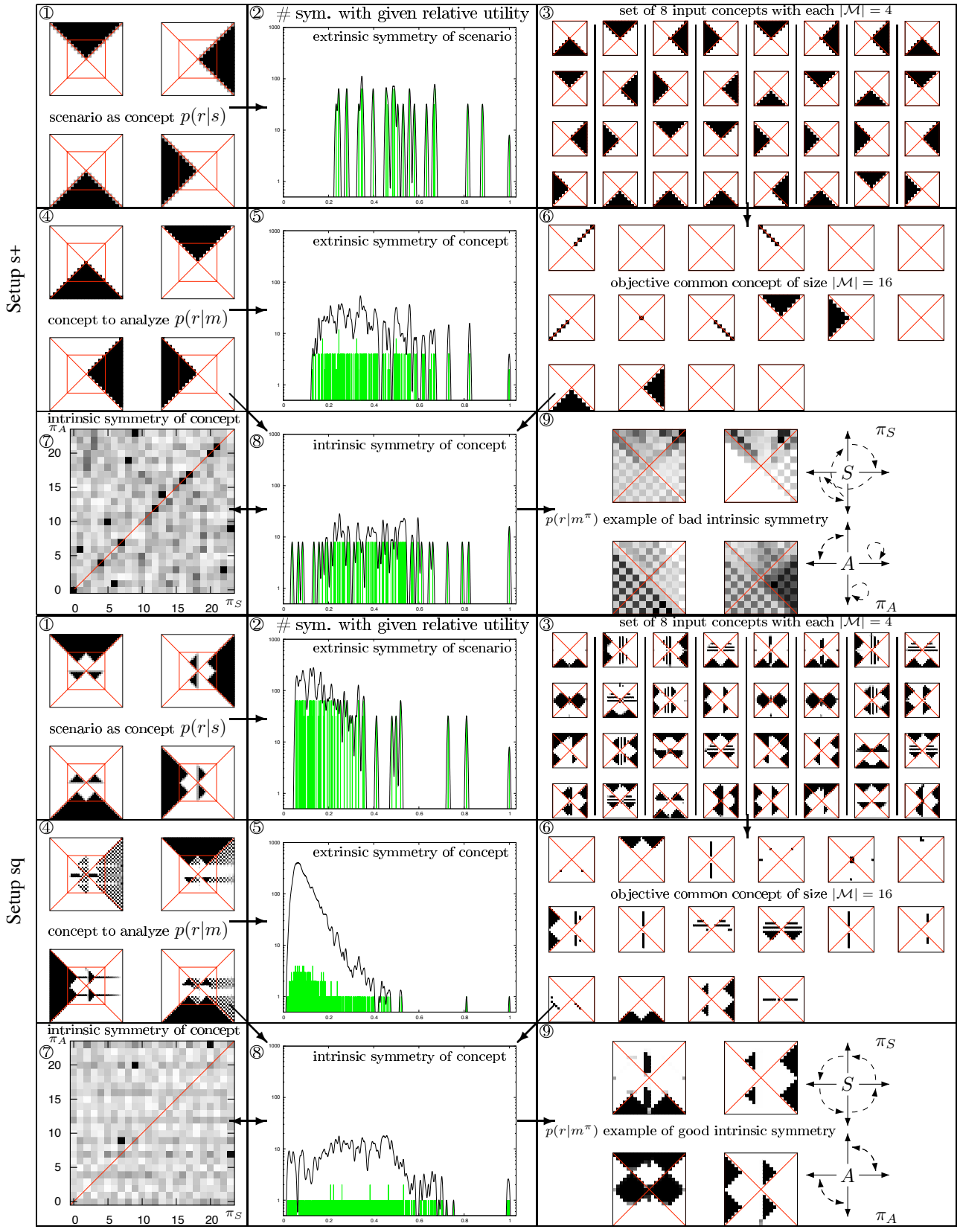


Figure 11: Symmetry - see text for details

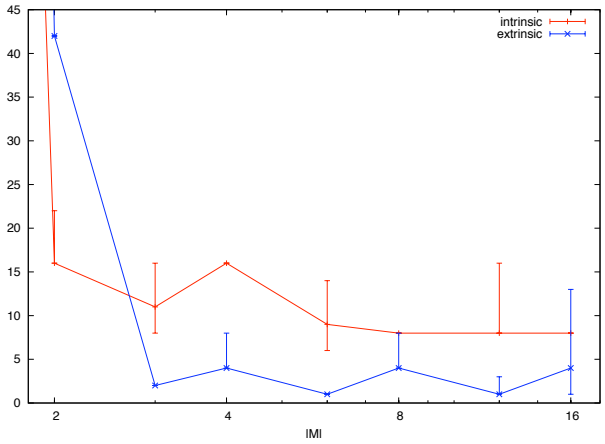


Figure 10: Number of good intrinsic and extrinsic symmetries depending on memory size

peaks, we added a smoothing curve into the spectrum. The rightmost peak with the 8 best symmetries includes all operations with translation $x_0 = y_0 = 0$. The next best peak (second rightmost) covers translation of length 1. The peaks are mostly ordered by their translation length $\sqrt{x_0^2 + y_0^2}$.

The Extrinsic Symmetry of the Concept is shown in ④ and ⑤. ④ shows a concept derived from an initial position capturing agent (R_0, M_{15}) with $|\mathcal{M}| = 4$ and its extrinsic symmetry spectrum similar to the setup in ① and ②. Also here the peaks are mainly ordered by translation distance. For setup s+ the peak with the best 4 operations contains the mirroring about the x resp. y axis. If we mirror around the x-axis we have additionally to translate the concept by 1 in y direction to get it perfectly matching with the original one. For setup sq we see that there is only one best symmetry, the identity. The next best symmetry, a mirroring around the x-axis, is much weaker.

An Objective Common Concept (R_0, M_*) which is used for the intrinsic symmetry is derived from 8 other solutions for the initial position capturing (shown in ③: each column of 4 symbols forms one input concept). The common concept has a size of $|\mathcal{M}_*| = 16$ symbols and is shown in ⑥.

The Intrinsic Symmetry of the Concept is shown once as spectrum in ⑧. ⑦ also shows these intrinsic symmetries but in a different way. The x-axis resp. y-axis enumerates the different possibilities for the permutations for π_S resp. π_A whereby 0 stands for the identity. The gray values are according to $\frac{I(R;M_{15}^{\pi})}{\max_{\pi} I(R;M_{15}^{\pi})}$ with an enlarged contrast for values near to 1 resp. black. The diagonal in this map stands for "synchronized" embodiment permutations with $\pi_S = \pi_A$. The introduction of those synchronized permutation makes only sense if, like in our case, sensor and actuator values can be associated and ordered in the same way.

⑨ shows on of the best (for setup sq) resp. of the worst (setup s+) concepts $(R_0; M_{15}^{\pi})$ after applying an intrinsic symmetry operation (π_S, π_A). This operation resp. its two permutations are shown to the right of the concept. The 4 possible values for S resp. A are shown as solid arrows and their permutation mappings with dashed arrows. In case of setup s+ the 16 best symmetries are similar to the 8 rotation and mirroring of the most right two input concepts shown in ③. In case of setup sq the 4 best symmetries are similar to the shown example but mirrored around the x- and/or y-axis.

Symmetry Dependence on Memory size $|\mathcal{M}|$ is shown in Fig. 10. It shows the number (y-axis) of good extrinsic resp. intrinsic symmetries (at least 85 % of maximal symmetry utility) for an initial position capturing agent with setup s+ according to memory size $|\mathcal{M}|$ (x-axis). The error-bars show the number of symmetries with at least 82.5% resp. 87.5% of maximal symmetry utility.

Discussion

We have shown how to extract common perspectives out of a group of agents with individual perspectives. There is evidence that both objective and subjective methods are almost similar if, as in our case, the input concepts are mostly deterministic. So one can save computation resources by calculating only the objective one. Both methods are based on the fact that for some locations in the world, the agents have a disagreement about how to group them to symbols (in our example e.g. the 4 diagonals in setup s+). Additionally to assigning the "original" symbols for indisputable areas, the common concept methods are able to identify the disputed areas and assign new symbols to them. If we would enlarge the memory size for the individual agents, they would find some of these new symbols as well. In our example, an agent with a bigger memory size would also find the diagonals but with much lower accuracy. Especially the symbol for the "center of the world" (Fig. 11 setup s+ ⑥) was never found by an individual agent in our experiments. So a new level for structuring the world emerged by considering a whole group of agents instead of individuals.

We also have evidence that "good" agents' concepts and especially common concepts have a higher degree of "symmetry". We developed two methods to study the strength of symmetry. With the extrinsic symmetry method, rotation and mirroring symmetries were found but the translations were not. Some "long distance" similarities in the sq setup we expected to appear were too weak and vanished in the "noise" of other translations. But, as expected, small translations were not completely asymmetrical. In general, the degree of symmetry is vaguely ordered by its translation length.

As opposed to the extrinsic, the intrinsic symmetry just observes which changes in the agent's interaction with the

environment (actuator/sensor permutations) have no (bad) effect on its concept. This method is additionally improved in that we do not compare a permuted concept with the individual (original) concept but with a common one. This common concept is "free" of special decisions of individual agents and gives a more universal representation for a task than any individual solution. The intrinsic operation forces the agents to "conform" to the common concept without optimizing them again by "transplanting their brain into another body". Searching for the best intrinsic operations is in fact partly a re-optimization of the controller. But in total, we only test in the shown example a vanishingly small ($3.1 \cdot 10^{-17}$-th) part of the search space. The meaning of the intrinsic symmetry method is not yet fully understood. Partly the intrinsic symmetries are identical to the extrinsic symmetries (rotation, mirroring) but they include many more operations.

Increasing the agent's memory size, the number of best extrinsic symmetries drops to 1 which means that identity is the only remaining symmetry operation. The number of best intrinsic symmetries behaves differently which means that intrinsic symmetries are not too sensitive to variations of the concept due to symmetry operations. This raises another interesting idea: The intrinsic symmetry may give us a hint for an optimal memory size of an agent. With growing memory size, the agents begin to "realize" that not every symmetry they "see" is really in the world. But this process stops at a certain $|\mathcal{M}|$ which might be a good choice for an agents memory size in the considered environment.

Conclusion and Outlook

We discussed two techniques to generate a common perspective by conflating the individual perspectives of a group of agents. Through this common perspective, we were able to analyze the similarity of individual agent representations and find common classifications of the environment. Additionally, some features of the world are only (or at least much more easily) detectable in the common perspective. We did not model the process of agreeing between these agents and only used very general information-theoretical principles which make them applicable to other scenarios as well.

We found evidence that good classifications of the environment capture many of its symmetries. While individual concepts may suffer some symmetry breaking, common concepts will reveal these symmetries. To analyze these symmetries, we developed two information-theoretical approaches. In the extrinsic approach, we measure for every symmetry transformation of the environment the degree to which the concept is respected. This approach abstracts away from how we achieved the classification. In contrast, the intrinsic approach is only suitable for agents interacting with an environment through a PAL. Here we analyze which modifications of the embodiment lead to agents who are "similar" to the original one. Since we measure this similarity indirectly by comparing the transformed concept to a common concept, the individual concept's symmetry breaks do not influence this method. The intrinsic method provides insight into the agent and its perspective on the environment. It identifies symmetries beyond the geometrical symmetries of the world found in the extrinsic case. The intrinsic symmetries accord to changes of the agent's embodiment which can not be detected by the agent.

Especially the role of the intrinsic symmetry and its meaning is not fully understood. In the future, it could help to extract structural regularities in the environment by the agent.

Acknowledgments

We want to thank Prof. Barbara Drossel for many ideas and suggestions. The first author was supported by scholarship from the Studienstiftung des deutschen Volkes.

References

Barlow, H. B. (1959). Possible principles underlying the transformations of sensory messages. In Rosenblith, W. A., editor, *Sensory Communication: Contributions to the Symposium on Principles of Sensory Communication*, pages 217 – 234. The M.I.T. Press.

Crutchfield, J. (1990). Information and its Metric. In Lam, L. and Morris, H., editors, *Nonlinear Structures in Physical Systems – Pattern Formation, Chaos and Waves*, pages 119–130. Springer Verlag.

i Cancho, R. F. and Solé, R. V. (2003). Least effort and the origins of scaling in human language. *PNAS 788-791*, 100(3):788–791.

Klyubin, A. S., Polani, D., and Nehaniv, C. L. (2007). Representations of space and time in the maximization of information flow in the perception-action loop. *Neural Computation*, 19(9):2387–2432.

Laughlin, S. B. (2001). Energy as a constraint on the coding and processing of sensory information. *Current Opinion in Neurobiology*, (11):475–480.

Linsker, R. (1988). Self-organization in a perceptual network. *Computer*, 21(3):105–117.

Philipona, D. and O'Regan, J. (2006). Color naming, unique hues, and hue cancellation predicted from singularities in reflection properties. *Visual Neuroscience*, 23(3-4):331–339.

Steels, L. (1997). The synthetic modeling of language origins. *Evolution of Communication*, 1(1):1–34.

Tishby, N., Pereira, F., and Bialek, W. (1999). The information bottleneck method. In *Proceedings of the 37-th Annual Allerton Conference on Communication, Control and Computing*, pages 368–377.

Wennekers, T. and Ay, N. (2005). Finite State Automata Resulting from Temporal Information Maximization and a Temporal Learning Rule. *Neural Comp.*, 17(10):2258–2290.

KohonAnts: A Self-Organizing Ant Algorithm for Clustering and Pattern Classification

A.M. Mora[1], C.M. Fernandes[1,2], J.J. Merelo[1], V. Ramos[2], J.L.J. Laredo[1] and A.C. Rosa[2]

[1]Departamento de Arquitectura y Tecnología de Computadores. University of Granada (Spain)
[2]LASEEB-ISR/IST. University of Lisbon (Portugal)
amorag@geneura.ugr.es, cfernandes@laseeb.org

Abstract

In this paper we introduce a new ant-based method that takes advantage of the cooperative self-organization of Ant Colony Systems to create a naturally inspired clustering and pattern recognition method. The approach considers each data item as an ant, which moves inside a grid changing the cells it goes through, in a fashion similar to Kohonen's Self-Organizing Maps. The resulting algorithm is conceptually more simple, takes less free parameters than other ant-based clustering algorithms, and, after some parameter tuning, yields very good results on some benchmark problems.

Introduction and State of the Art

Clustering is performed naturally by some types of ants at least in two different ways. First, ant colonies recognize by odour other member of their colony (as mentioned in the paper by Labroche et al. (2003)) leading to a natural clustering of ants belonging to the same nest, which is a consequence of nurturing and also has some genetic support; second, ants do physically cluster their larvae and dead bodies, putting them in piles whose position and size is completely self-organizing, as described by Deneubourg et al. (1991). Ant algorithms inspired by these models such as those proposed by Bonabeau et al. (1998); Abraham and Ramos (2003); Labroche et al. (2003); Ramos and Merelo (2002) have been applied to clustering and classification. In general, these methods follow the second clustering behavior: data for training the clusters is represented as *dead bodies*, which ants have to pick up (with a certain probability, and following some rule) and drop (also following some rule), while at the same time dropping and following pheromones. This results in the introduction of a few artifacts in the method: while the number of *dead bodies* (data items) to sort is *natural*, grid size, number of ants, pheromone following behavior and the rest is not. This results in a certain amount of parameter tuning for obtaining good results, but in any case is farther away from natural inspiration.

In this paper we present KohonAnts, an Ant algorithm that merges the biologically inspired concepts in Kohonen's Self-Organizing Map (proposed and described in Kohonen (1988, 2001)) and Chialvo and Millonas (1995) ant algorithm (both will be introduced in next section). It is based in several new ideas. First, as in the above-mentioned Labroche et al. model, every ant represents a data item. Ants move in a grid dropping *vectorial* pheromones. The grid is filled with initially random vector pheromones (of the same dimension as the data), and every time an ant falls in a cell, it changes the pheromone following a method similar to that used in Kohonen Self-Organizing Map, making the cell pheromone closer to the data item stored in the ant itself.

Since ants move around in the grid, ant position and pheromone content co-adapt, so that eventually ants with similar data items are close together in the grid (a *nesting* behavior), and the grid itself contains vectors similar to those stored in the ants on top of them. The grid can then be used to classify in the same way as Kohonen's Self-Organizing Map (but with better results), while ants can be used to visually identify the position of the clusters.

The interesting part of this method is that self-organization comes through stigmergy: ants change their environment (pheromones stored on the grid), and that influences the behavior of the rest of the ants (that follow a path changed by their cluster-siblings). There are less non-natural parameters (grid size is one of them), and, finally, results obtained are quite competitive with other methods tested.

In this paper, after presenting all concepts used in our method in section *Preliminary Concepts*, after it, we will describe the KohonAnts model itself in section *Self-Organizing Ants Model*, followed by the experiments in section *Experiments and Results*. Finally, we will conclude our description in section *Conclusions and Future Works* with a discussion of the obtained results and future lines of work.

Preliminary Concepts

Before describing KohonAnts, we would like to introduce the algorithms in which it is based on for the unfamiliar reader. First, Ant Colony Optimization (ACO) algorithms are presented in subsection *ACO*, followed by Kohonen's

Self-Organizing Map in subsection *SOM*. Finally, Chialvo and Millonas' model is presented in subsection *Ant System Model*.

ACO

The ACO is a meta-heuristic inspired by the behavior of some species of ants that are able to find the shortest path from nest to food sources in a short time. The method is based in the concept of *stigmergy*, that is, communication between agents using the environment. Every ant, while walking, deposits a substance called *pheromone* which other ants can sense. The ants tends to follow pheromone (it evaporates after some time) so, in intersections between several trails, an ant moves with high probability following the highest pheromone level. This metaheuristic was introduced by Dorigo et al. in 1991 (see Dorigo and Caro (1999) and Dorigo and Stützle (2002) for more details).

ACO algorithms take this behavior as inspiration to solve combinatorial optimization problems, using a colony of artificial ants as computational agents that communicate each other using *pheromones*. The problem to be solved using ACO must be transformed into a graph with weighted edges. In every iteration, each ant builds a complete path (solution), by travelling through the graph. At the end of this construction (and in some versions, during it), each ant leaves a trail in the visited edges depending on the fitness of the solution it has found. This is a measure of desirability for that edge and it will be considered by the following ants. In order to guide its movement, each ant uses two kinds of information that will be combined: *pheromone trails*, which correspond to 'learnt information' changed during the algorithm run, denoted by τ; and *heuristic knowledge*, which is a measure of the desirability of moving to the next node, based in previous knowledge about the problem (does not change during the algorithm run), denoted by η. The ants usually choose edges with better values in both properties, but sometimes they may 'explore' new zones in the graph because the algorithm has a stochastic component, that broadens the search space to regions not previously explored. Due to all these properties, all ants cooperate in order to find the best solution for the problem (the best path in the graph), resulting in an global emergent behavior. There are lots of variants and new methods, but we introduce Ant Colony System (ACS) because our model takes some features of it.

The building of solutions is strongly based in the *state transition rule* (called *pseudo-random proportional state transition rule* in ACS), since every ant uses it to decide which node *j* is the next in the construction of a solution (path), when the ant is at the node *i*. This formula calculates the probability associated to every node in the neighbourhood of *i*, and is as follows:
If ($q \leq q_0$)

$$j = \arg\max_{j \in N_i} \left\{ \sum_{u \in N_i} \tau(i,u)^{\alpha} \cdot \eta(i,u)^{\beta} \right\} \quad (1)$$

Else

$$P(i,j) = \begin{cases} \dfrac{\tau(i,j)^{\alpha} \cdot \eta(i,j)^{\beta}}{\sum_{u \in N_i} \tau(i,u)^{\alpha} \cdot \eta(i,u)^{\beta}} & if\ j \in N_i \\ 0 & otherwise \end{cases} \quad (2)$$

Where q is a random number in [0,1] and q_0 is a parameter which set the balance between exploration and exploitation. If $q \leq q_0$, the best node is chosen as next (exploitation), on the other hand one of the feasible neighbours is selected, considering different probabilities for each one (exploration). α and β are weighting parameters to set the relative importance of pheromone and heuristic information respectively, and N_i is the current feasible neighbourhood for the node *i*.

There is a *global pheromone updating*, which is only performed for the edges of the global best solution, so for every edge (i,j) in $S_{GlobalBest}$ is:

$$\tau^t(i,j) = (1-\rho) \cdot \tau^{t-1}(i,j) + \rho \cdot \Delta\tau(i,j)_{GlobalBest} \quad (3)$$

t marks the new pheromone value and *t-1* the old one. ρ in [0,1] is the common evaporation factor and $\Delta\tau$ is the amount of pheromone deposited depending on the quality of the best solution.

There is also a *local pheromone updating*, which is performed by each ant, every time that a node *j* is added to the path which it is building. This formula is:

$$\tau^t(i,j) = (1-\varphi) \cdot \tau^{t-1}(i,j) + \varphi \cdot \tau_0 \quad (4)$$

Where φ in [0,1] is the local evaporation factor and τ_0 is the initial amount of pheromone (it corresponds to a lower trail limit). This formula results in an additional exploration technique, because it makes the edges traversed by an ant less attractive to the following ants and helps to avoid that many ants follow the same path.

SOM

The Self-Organizing Map (SOM) was introduced by Teuvo Kohonen in 1982 (see Kohonen (2001) for details). It is a non-supervised neural network that tries to imitate the self-organization done in the sensory cortex of the human brain, where neighbouring neurons are activated by similar stimulus. It is usually used either as a clustering/classification tool or as a method to find unknown relationships between a set of variables that describe a problem. The main property of the SOM is that it makes a nonlinear projection from a high-dimensional data space (one dimension per variable) on a regular, low-dimensional (usually 2D) grid of neurons (see Figure 1).

Since this type of network is distributed in a plane (2-dimensional structure) it can be concluded that the projections preserve the topologic relations while simultaneously

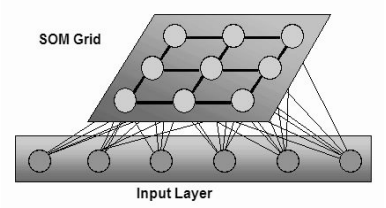


Figure 1: SOM Grid structure. There is an input layer (with the input samples) and a process layer (where the neurons of the network are) which takes a grid shape.

creating a dimensional reduction of the representation space (the transformation is made in a topologically ordered way).

The SOM processes a set of input vectors (samples or patterns), which are composed by variables (features) typifying each sample, and creates an output topological network where each neuron is associated also to a vector of variables (model vector) which is representative of a group of the input vectors. Note in Figure 1 that each neuron of the network is completely connected to all the nodes (each node is a sample) of the input layer. So, the network represents a feed-forward structure with only one computational layer formed by neurons or model vectors.

There are four main steps in the processing of the SOM. Excepting the first one, the others are repeated until a stop criteria is reached:

- **Initialization of model vectors**. Usually it is made by assigning small random values to their variables, but there are some other possibilities as an initialization using random input samples.

- **Competitive process**. For each input pattern X, all the neurons (model vectors) V competes using a *similarity function* in order to identify the most similar or close to the sample vector. The most usual function is a distance measure (as Euclidean distance). The winner neuron is called the best matching unit (BMU).

- **Cooperative process**. The BMU determines the centre of a topological neighbourhood where those neurons inside it will be updated (the model vectors) to be even more similar to the input pattern. There is a *neighbourhood function* used to determine the neurons to consider. If the lattice where the neurons are is rectangular or hexagonal, it is possible to consider as neighbourhood rectangles or hexagons with the BMU as centre. Although it is more usual to use a Gaussian function to assure that the farther the neighbour neuron is, the smaller the updating to its associated vector is. In this process, the neurons inside a vicinity cooperate all of them to learn.

- **Learning process**. In this step the variables of the model vectors inside the neighbourhood are updated to be closer to those of the input vector. It means doing the neuron more similar to the sample. The *learning rule* used to update the vector (V) for every neuron i in the neighbourhood of the BMU is:

$$V_i^t = V_i^{t-1} + \alpha^t \cdot N_{BMU}^t(i) \cdot (X - V_i^{t-1}) \qquad (5)$$

Where t is the current iteration of the whole process, X is the input vector, N_{BMU} is the neighbourhood function for the BMU, which returns a high value (in [0,1]) if the neuron i is in the neighbourhood and close to the BMU (1 if $i = BMU$), and a small value in the other case (0 if i is not located inside the neighbourhood). α is the *learning rate* (also in (0,1]). Both (neighbourhood and learning rate) depends on t, since it is usual to decrease the radius of the first one and the value of the second in order to make higher updating at the beginning of the process and almost none in the latter.

The consecutive application of Equation 5 and the update of the neighbourhood function, has the effect of 'moving' the model vectors, V_j from the winning neuron towards the input vector X_i. It is, the model vectors tend to follow the distribution of the input vectors. Consequently, the algorithm leads to a topological arrangement of the characteristic map of the input space, in the sense that adjacent neurons in the network tend to have similar weights vectors.

As a consequence, looking at the display of a SOM, it is possible to recognize some clusters as well as the metric-topological relations of the data items (vectors of variables of the problem) and the outstanding variables.

Ant System Model

In Chialvo and Millonas (1995), the authors presented a simple ant model where trails and networks of ant traffic emerge without impositions by any special boundary conditions, lattice topology, or additional behavioral rules. In this model, the state of an ant can be expressed by its position r and orientation θ. Since the response at a given time is assumed to be independent of the previous history of the individual, it is sufficient to specify a transition probability from one place and orientation (r, θ) to the next (r^*, θ^*) an instant later. Initial papers by Millonas (1992, 1994) transition rules were derived and generalized from noisy response functions, which in turn were found to reproduce a number of experimental results with real ants. The response function can effectively be translated into a two-parameter transition rule between the cells by using the pheromone weighting function showed in Equation 6:

$$W(\sigma) = \left(1 + \frac{\delta}{1 + \sigma \cdot \delta}\right)^{\beta} \qquad (6)$$

This equation measures the relative probabilities of moving to a cell r with pheromone density $\sigma(r)$. The parameter β is associated with the osmotropotaxic sensitivity proposed in Wilson (1971). In practical terms, this parameter controls the degree of randomness with which each ant follows the gradient of pheromone: for low values of β, pheromone concentration does not greatly affect its choice, while high

values cause it to follow pheromone gradient with more certainty, as proved in Chialvo and Millonas (1995). The sensory capacity $1/\delta$ describes the fact that each ant's ability to sense pheromone decreases somewhat at high concentrations. In addition to the former equation, there is a weighting factor $w(\Delta\theta)$, where $\Delta\theta$ is the change in direction at each time step, i.e. measures the magnitude of the difference in orientation. This weighting factor ensures that very sharp turns are much less likely than turns through smaller angles; thus each ant in the colony have a probabilistic bias in the forward direction. A discretization of the model is necessary in order to perform simulations and test some assumptions: Chialvo and Millonas created a square lattice where ants can move around, taking one step at every iteration. The decision (where to go) is made according to the pheromone concentration in all eight neighboring cells (Von Neumann neighborhood) and the weighting factor $w(\Delta\theta)$, using Equation 6, and computing the transition probabilities via Equation 7:

$$P_{ik} = \frac{W(\sigma_i) \cdot w(\Delta_i)}{\sum_{j/k} W(\sigma_j) \cdot w(\Delta_j)} \tag{7}$$

This equation represents the transition probabilities on the lattice to go from cell k to cell i and notation j/k indicates the sum over all the cells j which are in the local (Von Neumann) neighborhood of k. Δ_i measures the magnitude of the difference in orientation for the previous direction at time $t-1$. As an additional condition, each individual leaves a constant amount η of pheromone at the cell where it is located at every time step t. This pheromone decays at each time step at a rate k. Toroidal boundary conditions are imposed on the lattice to avoid boundary effects. Please note that there is no direct communication between the organisms but a type of indirect communication through the pheromone field. In fact, ants are not allowed to have any memory and the individual's spatial knowledge is restricted to local information about the whole colony pheromone density.

This model has been applied in many different works, for instance in Ramos and Almeida (1994), the authors adapted it by placing the ants 'over' a gray-scale image. So, they evolve reinforcing pheromone levels around pixels with different gray levels yielding pheromone maps that may be a suitable support for edge detection and image segmentation. This last model was improved in Fernandes et al. (2005a) by introducing a mechanism to eliminate and create ants along the evolution process, which means a self-regulated population size and it results faster and also more effective in creating pheromone trails around the edges of the images.

Self-Organizing Ants Model

The algorithm presented in this paper is an ant algorithm with some common features with the Ant System of Chialvo et al., nevertheless it also includes some other features inspired by the Kohonen's SOM. It is called, for this reason, *KohonAnts* (or KANTS).

KANTS has been designed as a clustering and classification algorithm, so it is capable to group a set of input samples (training dataset) into clusters with similar features. In addition it behaves as a good classification algorithm. It works in a non-supervised (self-organizing) way, without considering the class of the input patterns during the process.

The main idea is to assign each input sample (which is a vector) to an ant, and put them into an habitat which is a toroidal $X \cdot Y$ grid. Then, they move around in the lattice changing the environment, which is a stigmergic mechanism. Every cell of the grid that constitutes the environment also contains a vector of the same dimension and range as the training set. The factor of change of the environment) depends on the values of the ant's vector, and, since every ant tends to move towards those zones in the grid which are more similar to themselves (to their associated vectors), ant position and pheromone content co-adapt. This means that eventually, ants with similar data items will be close together in the grid, and the grid itself will contain similar vectors to those stored in the ants on top of them.

Then, the grid can be used as a classification tool (in the same way as the resulting map after training using Kohonen's SOM), while ants will be grouped in clusters of similar individuals.

In the following paragraphs we present the most important features of the algorithm.

Decide Where to Go Rule

This is the most important function in the algorithm. It is used by every ant placed at cell i to decide which is the next cell j to move.

This function is based in Chialvo's Ants System pheromone weighting function and pseudo-random proportional rule of ACS, so it is:
If ($q \leq q_0$)

$$j = \arg\max_{j \in N_i} W(\sigma_{ij}) \tag{8}$$

Else

$$P_{ij} = \begin{cases} \dfrac{W(\sigma_{ij})}{\sum_{u \in N_i^t} W(\sigma_{iu})} & if\ j \in N_i^t \\ 0 & otherwise \end{cases} \tag{9}$$

In that rule, $q_0 \in$ [0,1] is the standard ACS parameter and q is a random value in [0,1]. N_i is the neighbourhood of the cell i, which is a function similar to the one used in SOM. It also has associated a *neighbourhood radius,* nr which diminish along the running, so the neighbourhood is different at every iteration t. This function returns '1' if the cell is included in the neighbourhood and '0' otherwise.

σ is defined by the following equation:

$$\sigma_{ij} = \sqrt{V_i(v)^2 - CTR_j(v)^2} \qquad \forall v = 1..nvars \tag{10}$$

Where V_i is the vector associated to the cell i and CTR_j is the centroid of a zone centered in the cell j. It is a vector where each value takes the arithmetic mean of the correspondent values of the vectors associated to the cells in-

cluded within a *centroid radius, cr*. The formula is equivalent to calculate the Euclidean distance between the vector associated to the cell i and the centroid vector for the cell j, both vectors have a number of variables $nvars$.

Finally, in the *decide where to go* rule, $W(\sigma)$ is the Ant System pheromone weighting function (Equation 6).

The rule works as follows: when an ant is building a solution path and is placed at one node i, a random number q in [0,1] is generated, if $q \leq q_0$ the best neighbour j is selected as the next node in the path (Equation 8). Otherwise, the algorithm decides which node is the next by using a roulette wheel considering P_{ij} as probability for every feasible neighbour j (Equation 9).

Notice that the second part of the rule (Equation 9) is similar to the transition probability defined by Chialvo et al. (Equation 7), but considering a weighting factor $w(\Delta\theta) = 1$, so, all the neighbour cells have the same probability in advance (before considering the σ value). We tested the algorithm with some other weighting values, but the results are not clearly improved. Further research will be focused on this issue.

In addition, there is an important factor to mark, which is that the ants are capable to move to cells far more than one hop from the cell where they are currently located. It means that they can 'jump' or 'fly' as some real-world ant species are able. This property is vanishing along the algorithm running because the neighbourhood radius is decreased until it takes a value of '1' (ants only move from one cell to a one hop distance neighbour).

The Updating Function

This process is usually performed in classical ant algorithms as a pheromone trail deposition. At every step, each ant k updates the cell i where is placed, using an updating formula similar to the learning function of SOMs (see Equation 5). Bearing in mind that every sample/ant and cell in the grid is a vector of $nvars$ variables, the formula is as follows:

$$V_i^t(v) = V_i^{t-1}(v) + R \cdot [a_k(v) - V_i^{t-1}(v)] \quad \forall v = 1..nvars \tag{11}$$

Where V_i is the vector associated to the cell i, t is the current iteration, and a_k is the vector associated to the ant k. R is the reinforce of the update, which is described as:

$$R = \alpha \cdot (1 - \overline{D}(a_k, CTR_i)) \tag{12}$$

α is the learning rate factor typical in SOM (which is constant in this algorithm), CTR_i is again the centroid of a zone centered in the cell i. Finally, $\overline{D}$ is the mean Euclidean distance between the ant's vector and the centroid vector. It is:

$$\overline{D} = \sum_{v=1}^{nvars} \frac{\sqrt{a_k(v)^2 - C_i(v)^2}}{nvars} \tag{13}$$

The Evaporation Function

As in all the ant algorithms, it is a very important process in which the environment reverts to its previous (or initial) state. This process is performed,for every cell i, once all the ants have moved and updated the environment in the current iteration.

$$V_i(v) = V_i(v) - \rho \cdot V_{i0}(v) \qquad \forall v = 1..nvars \tag{14}$$

Where ρ is the usual evaporation factor and V_{i0} is the initial vector associated to the cell i. It means that the function changes the values of the vector in order to be similar to the initial, which can be interpreted as an evaporation of the trails in the environment.

Pseudocode

The pseudocode of our model is presented in Algorithms 1 and 2. Here we consider each cell as a pair of coordinates, because the algorithm works using a grid.

Algorithm 1 KANTS Algorithm

```
initialize_randomly_grid_vectors
place_randomly_ants_in_grid
for N_iterations do
   for each ant a at cell (x, y) do
      j = decide_where_to_go(a,(x, y))
   end for
   update_grid    // Using Equation 11
   evaporate_grid    // Using Equation 14
   update_neighbourhood_radio
end for
```

Algorithm 2 Decide_Where_To_Go $(a,(i,j))$

```
for all cells (x, y) in neighbourhood of (i, j) do
   // Probability = Euclidean Distance to centroid
   σ_{ij,xy} = E_D((i, j),centroid((x, y)))
   compute W(σ_{ij,xy}) and P_{ij,xy}    // Using Equations 6 and 9
end for
// Ant Colony System/Ant System. Equations 8 and 9
q = random(0,1)
if q ≤ q0 then
   // selected cell = the one with maximum probability
   (k, l) = MAX(P_{ij,xy})
else
   // selected cell = roulette_wheel
   (k, l) = roulette_wheel(P_{ij,xy})
end if
```

Experiments and Results

This section presents the data sets used to train and test KANTS algorithm (Subsection *The Datasets*), followed by the results obtained in clustering (Subsection *Clustering*) and classification (Subsection *Classification*).

The Datasets

The datasets used to test and validate the model are some well-known real world databases:

- **IRIS** contains data of 3 species of iris plant (Iris Setosa, Versicolor and Virginica), 50 samples of each one and 4 numerical attributes (the sepal and petal lengths and widths in cms.). The first class is linearly separable from the others while the other two are not.

- **GLASS** contains data from different types of glasses studied in criminology. There are 6 classes, 214 samples (unevenly distributed in classes) and 9 numerical features related to the chemical composition of the glass. This database is difficult to classify (and depending on the algorithm, also difficult to cluster), since some classes are represented by just a few samples (3-10), and some other classes not being linearly separable.

- **PIMA**. This is the Pima Indians Diabetes database which contains data related to some patients (indians of that tribe) and a class label representing their diabetes diagnostic according to the world-wide health organization's criterion. There are 768 samples with 8 numerical features (medical data). Again, this is a hard to process database, because many samples of the two classes takes close values for the same variables.

In each of the three databases, we have consider 3 sets built by transforming the original into 3 disjoint sets of equal size. The original class distribution (before partitioning) is maintained within each set. Then we consider 3 pair of datasets 'training-test' by splitting the 3 previous into half size ones, they are named including the text *50tra-50tst*. In addition, 3 other pairs are created, but considering a distribution of 90% of samples for training and 10% for test. These sets are named including *90tra-10tst*.

Clustering

In Chialvo and Millonas (1995), the authors performed a study on the distribution of ants with different configurations in the β-δ parameter space. Three types of behavior were observed when looking at the snapshots of the system after 1000 iterations: disorder, patches and trails.

The results obtained with their method follow theoretical prediction: a second order phase transition is observed, when a region of the parameter space which gives rise to disorder regimes "turns into" a region where trails are formed. Moving away from the order-disorder line, the system loses its ability to evolve lines/trails of ants and patches gradually appear. In addition, another experiment was conducted: the system was tuned to a region in the parameter space were trails emerge. After the traffic network was formed, β was decreased in order to tune the system bellow the transition line; then, the ants started executing random walks and left their previously formed trails. Once β was set again to the initial value, the ants self-organized again on a similar traffic network.

A similar test was performed with KANTS, but since Iris dataset was used (and due to it is not very complex), we have run the algorithm only a few iterations.

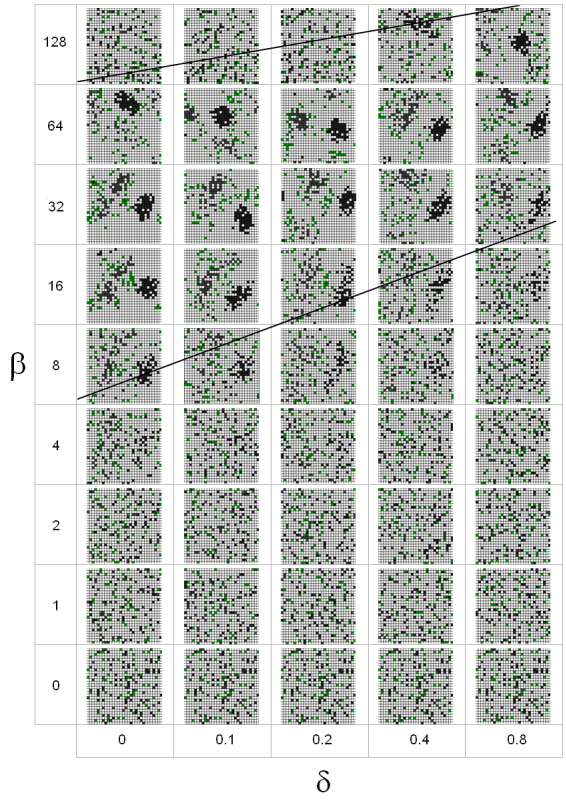


Figure 2: Snapshots of the ants in the system after 100 iterations for different β and δ values. The straight lines roughly delimit the region where clusters emerge.

Parameters β and δ were varied, and the resulting ants' distribution after 100 iterations is depicted in Figure 2. Parameters α, *neighbourhood radius (nr)* and *centroid radius (cr)*, were set to 1, 1 and 3, respectively. From the figures it is not possible to distinguish three different types of behavior, as in Chialvo and Millonas' experiments with the original model, but it is clear that there is a transition line from a disordered state, where ants/data do not cluster, and a ordered state where cluster start to emerge. Further away from the transition line, the model's ability to form clusters gradually starts to decay (again). In the same way as in the original model, there is only a small region of the parameter space that gives rise to a self-organized behavior, but while Ant System forms trails, KANTS emerge clusters of ants that are actually data samples.

Considering this results, KANTS appear to be a promising tool for data clustering. With a simple mechanism and proper tuning of β and δ, data represented by (and behaving as) ants form clusters that are easily distinguishable in the grid. Even if some kind of local search is eventually neces-

sary in order to tackle real-world problems, KANTS by now come forward as a core model where hybridization may be performed and the resulting algorithms applied to hard problems.

In Figure 3 an example of the ants evolution (movement during the run) in the grid is showed.

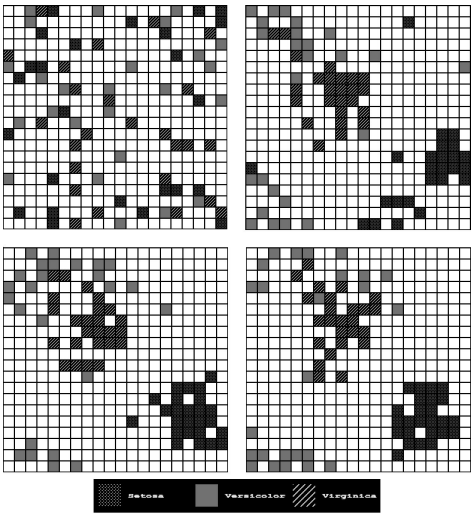


Figure 3: Evolution of position of ants in the grid for the IRIS problem. It shows the situation at the beginning (top-left), at step 50 (top-right) and 100 (bottom-left) and at step 150 (bottom-right).

Looking at the snapshots of the grid at different iterations, it is possible to notice that every ant tends to move to a group of ants of the same class (they have similar values for the features). So, starting from a random initial configuration, in a few steps, the ants forms visible clusters.

Classification

In order to classify with KANTS, we introduce a parameter: the number of neighbours to compare with the test sample. So, the algorithm searches for the K nearest vectors in the grid (using the Euclidean distance) to the vector correspondent to the sample which it wants to classify. It assigns the class of the majority.

It is similar to the one used in K-Nearest Neighbours method (see Fix and J. L. Hodges (1989) for details), but we use it once the grid has been trained (using the training dataset) and many times the algorithm works very well even considering $K = 1$.

Since KANTS is a stochastic approach, 10 runs were made considering each pair of datasets (training and test). Results are presented in Table 1, where mean, standard deviation, and best of the resulting percentages in classification are given. We compare the results with those yielded using the traditional deterministic method K-Nearest Neighbours (KNN).

IRIS	KANTS		KNN	
Dataset	Best	Mean	Best	Mean
50tra-50tst-Set1	98.67	98.00 $\pm$0.67	97.30	-
50tra-50tst-Set2	98.67	97.60 $\pm$0.53	96.00	-
50tra-50tst-Set3	100.00	98.80 $\pm$0.40	94.60	-
90tra-10tst-Set1	100.00	100.00 $\pm$0.00	100.00	-
90tra-10tst-Set2	100.00	99.33 $\pm$2.00	93.33	-
90tra-10tst-Set3	100.00	100.00 $\pm$0.00	93.33	-
GLASS	KANTS		KNN	
Dataset	Best	Mean	Best	Mean
50tra-50tst-Set1	68.22	65.42 $\pm$1.62	62.60	-
50tra-50tst-Set2	67.29	64.86 $\pm$1.52	64.40	-
50tra-50tst-Set3	74.77	71.03 $\pm$2.17	64.40	-
90tra-10tst-Set1	69.57	65.65 $\pm$1.30	47.80	-
90tra-10tst-Set2	73.91	73.48 $\pm$1.30	60.80	-
90tra-10tst-Set3	91.30	83.48 $\pm$3.25	82.60	-
PIMA	KANTS		KNN	
Dataset	Best	Mean	Best	Mean
50tra-50tst-Set1	75.52	74.32 $\pm$0.61	70.03	-
50tra-50tst-Set2	77.34	76.61 $\pm$0.58	71.80	-
50tra-50tst-Set3	77.60	75.13 $\pm$0.85	72.90	-
90tra-10tst-Set1	83.12	80.52 $\pm$1.42	64.90	-
90tra-10tst-Set2	79.22	75.32 $\pm$1.42	73.60	-
90tra-10tst-Set3	84.42	80.65 $\pm$2.05	70.10	-

Table 1: Classification results with Iris, Glass and Pima databases (6 different datasets each time).

The results are very good when comparing them with a traditional clustering and classification method such as KNN, even yielding 100% in many cases. We would like to enphasize the fact that the Glass and Pima datasets usually obtain a low classification rate (both are difficult databases, as we previously commented), while KANTS achieves in some cases a rate 10% higher than KNN. The results are even more encouraging considering that KANTS is a non-supervised algorithm.

In addition, it is important to comment that the algorithm's running time is just a few seconds, depending on the dataset size, so for these results it takes 8 seconds in Iris, 10 seconds in Glass and 20 seconds in Pima. All the experiments have been performed in a Pentium 1.6 GHz.

Conclusions and Future Work

This paper presents KohonAnts, a new method for clustering and data classification, based on an hybridization of Ant Algorithms and Kohonen Self-Organizing Maps. The new model turns n-variable data samples into artificial ants that evolve in a 2D toroidal grid *paved* with n-dimensional vectors. Data/Ants act on the habitat vectors by pushing the values towards their own. In addition, ants are attracted by regions were the vector values are closer to their own data. In this way, similar ants tend to aggregate in common regions of the grid. There is indirect communication between ants through the grid (stigmergy) leading, with a proper setting of the model's parameters, to the emergence of data clusters. In addition, ants' actions (pheromone deposition) over the grid and pheromone evaporation creates a kind of cognitive field which has turned out be very effective for classification purposes.

It has been demonstrated that KANTS model is useful for

clustering and classification tasks, yielding very good results in both kind of problems. The concept it is based on is quite simple and naturally inspired, but even so results obtained are quite good compared with traditional clustering methods (such as KNN). It is also a fast method, not needing a lot of computation time for obtaining the results mentioned above. As should be the spirit of publicly-funded research, we maintain all sources for the project as well as data used in experiments in the public repository `https://forja.rediris.es/websvn/wsvn/geneura/KohonAnts/`, under a GPL license[1].

As future short-term lines of work, we will perform further tests on the algorithm, comparing it with more specific clustering and classification methods. We will also try to streamline ant movement rules, and compare among different options.

In addition, a lot of enhancements are still possible in the original KANTS model presented in this paper. A neighbourhood function may be considered, similar to the one used in Self-Organizing Maps for updating the environment in a radius. As in Fernandes et al. (2005a) and in Fernandes et al. (2005b), reproduction may improve speed and accurateness of the algorithm. Chialvo and Millonas probability equation was not fully explored since weights $w(\Delta\theta)$ different of '1' yield worse solutions, so an in-depth study in this issue will be performed. Finally, a stopping criteria is needed in order to avoid unnecessary iterations in the process.

Acknowledgements

This work has been supported by NOHNES project from the Spanish Ministry of Science and Education (TIN2007-68083-C02-01). C. Fernandes also wishes to thank FCT, Ministério da Ciência e Tecnologia, his Research Fellowship SFRH/BD/18868/2004, also partially supported by Fundação para a Ciência e a Tecnologia (ISR/IST pluriannual funding) through the POS_Conhecimento Program that includes FEDER funds.

References

Abraham, A. and Ramos, V. (2003). Web usage mining using artificial ant colony clustering and linear genetic programming. *Evolutionary Computation, 2003. CEC'03. The 2003 Congress on*, 2.

Bonabeau, E., Theraulaz, G., Fourcassié, V., and Deneubourg, J. (1998). Phase-ordering kinetics of cemetery organization in ants. *Physical Review E*, 57(4):4568–4571.

Chialvo, D. and Millonas, M. (1995). *The Biology and Technology of Intelligent Autonomous Agents*, volume 144 of *NATO ASI Series*, chapter How swarms build cognitive maps, pages 439–450.

Deneubourg, J., Goss, S., Franks, N., Sendova-Franks, A., Detrain, C., and Chrétien, L. (1991). The dynamics of collective sorting robot-like ants and ant-like robots. *Proceedings of the first international conference on simulation of adaptive behavior on From animals to animats table of contents*, pages 356–363.

Dorigo, M. and Caro, G. D. (1999). The ant colony optimization meta-heuristic. In Corne, D., Dorigo, M., and Glover, F., editors, *New Ideas in Optimization*, pages 11–32. McGraw-Hill.

Dorigo, M. and Stützle, T. (2002). The ant colony optimization metaheuristic: Algorithms, applications, and advances. In F. Glover, G. K., editor, *Handbook of Metaheuristics*, pages 251–285. Kluwer.

Fernandes, C., Ramos, V., and Rosa, A. C. (2005a). Self-regulated artificial ant colonies on digital image habitats. *International Journal of Lateral Computing*, 2(1):1–8.

Fernandes, C., Ramos, V., and Rosa, A. C. (2005b). Varying the population size of artificial foraging swarms on time varying landscapes. In Duch, W., Kacprzyk, J., Oja, E., and Zadrozny, S., editors, *15th International Conference on Artificial Neural Networks, ICANN2005*, volume 3696 of *LNCS*, pages 311–316. Springer-Verlag.

Fix, E. and J. L. Hodges, J. (1989). Discriminatory analysis: Nonparametric discrimination: Consistency properties. In *International Statistical Review*, volume 57, pages 238–247.

Kohonen, T. (1988). Representations of sensory information in self-organizing feature maps, and the relation of these maps to distributed memory networks. In Cotterill, R. M. J., editor, *Computer Simulation in Brain Science*, pages 12–25. Cambridge University Press, Cambridge, UK.

Kohonen, T. (2001). *The Self-Organizing Maps*. Springer.

Labroche, N., Monmarche, N., and Venturini, G. (2003). AntClust: Ant Clustering and Web Usage Mining. *Proc. of GECCO-2003, Springer, LNCS*, 2723:25–36.

Millonas, M. (1992). A connectionist-type model of self-organized foraging and emergent behavior in ant swarms. *Journal Theor. Biology*, (159):529.

Millonas, M. (1994). Swarms, phase transitions, and collective intelligence. In Langton, C., editor, *Artificial Life III*, volume XVII, pages 417–445, Massachussetts. Addison-Wesley Reading.

Ramos, V. and Almeida, F. (1994). Artificial ant colonies in digital image habitats - a mass behavior effect study on pattern recognition. In Dorigo, M., Middendorf, M., and Stüzle, T., editors, *ANTS´2000 - 2nd International Workshop on Ant Algorithms*, pages 113–116, Brussels, Belgium.

Ramos, V. and Merelo, J. J. (2002). Self-organized stigmergic document maps: Environment as a mechanism for context learning. In Alba, E., Fernández, F., Gómez, J. A., Herrera, F., Hidalgo, J. I., Merelo-Guervós, J.-J., and Sánchez, J. M., editors, *Actas primer congreso español algoritmos evolutivos, AEB02*, pages 284–293. Universidad de Extremadura. `http://citeseer.nj.nec.com/ramos02selforganized.html`.

Wilson, E. (1971). *The Insect Societies*. Belknam Press, Cambridge.

[1] It is only requested that this paper (or another by the same authors) is referenced in published research.

Emergence of Cooperation in N-player games on small world networks

Colm O'Riordan[1], Alan Cunningham[1] and Humphrey Sorensen[2]

[1]NUI, Galway, Ireland
[2]University College Cork, Ireland.
colm.oriordan@nuigalway.ie

Abstract

The emergence of cooperation in social dilemmas has been addressed in a number of fields. In this paper, we illustrate how robust cooperation can emerge among a population of agents participating in a N-player dilemma when the agents are spatially arranged on a graph exhibiting small world properties. We present a graph structure with a high level of community structure, small diameter and a variance in the node degree distribution. We show that with simple learning rules, robust cooperation emerges. We also show that a population of agents whose interactions are constrained by such a graph can adapt to dramatic environmental changes.

Introduction

Questions regarding cooperation and its emergence, particularly in environments inhabited by self interested individuals, have been addressed in a many domains. The include, among many others, computer science (Chiba and Hiraishi, 1998), biology (Boyd and Richerson, 1988), robotics (Birk, 1999) and social science (Hardin, 1968).

Social dilemma games have been commonly adopted to capture and represent the salient features of interactions in these environments; in particular the conflict between the individually rational actions and the collectively rational group actions and outcomes. The prisoner's dilemma (and variations) is the most oft studied game. Most previous work has focussed on the case involving two participants. The extended N-player version is less studied but it has been argued by Davis et al. (1976) to have "greater generality and applicability to real life situations".

In N-player dilemma games defection is the rational choice for all individuals which in turn leads to a suboptimal outcome for the group. Many researchers have investigated the effect of spatial constraints on agent interactions in both the 2-player and N-player game (Hauert, 2006), (Wu et al., 2005), (Santos and Pacheo, 2005). In these spatially organised games, agents are more likely to interact with a smaller subset of agents than would be expected in simulations where are agents are not spatially organised, e.g. randomly organised or round robin type simulations. This factor has been shown to have a dramatic impact of the likelihood of cooperation emerging.

One form of spatial arrangement or topology that has generated much attention recently is that of a *small world graph* (Watts, 1999). Small world graphs are typified by the fact that most nodes are reachable from all other nodes in a short number of steps. These graphs also tend to have a high clustering coefficient with a high presence of cliques or near-cliques. Another property often associated with small world graphs is that the node degree distribution follows a power law distribution.

One key property that we have explored in previous work is that of *community structure* (O'Riordan and Sorensen, 2008b). This property has also been explored in recent work (Lozano et al., 2006). A graph is said to have a community structure if collections of nodes are joined together in tightly knit groups between which there are only looser connections. This property has been shown to exist in many real-world social networks (Newman and Girvan, 2004).

In our previous work, we have shown that by enforcing a high level of community structure robust cooperation can emerge among agents participating in N-player social dilemma games. The topologies explored in our previous work, however, are quite unrealistic and do not possess the other properties found in many naturally occurring graphs, i.e. small world properties including a variance in node degree.

This paper investigates whether it is possible to build graphs that exhibit the properties of small world graphs which induce the emergence of cooperation. We present two different extensions to our previous representations and illustrate that by constructing the small world graph while maintaining a high level of community structure that cooperation can indeed still emerge.

The following sections discuss some background material, particularly in N-player social dilemmas, graphs with community structure and some of our previous findings. We then discuss the particular graph model and agent interaction models used in this work. The experimental set up is then explained with our two algorithms for creating small world

graphs explained. We present results obtained from simulations with these two different topologies. Finally we present some conclusions and briefly outline some intended future work.

Background

N-player social dilemmas

N-player dilemmas are characterised by having many participants, each of whom may choose to cooperate or defect. These choices are made autonomously without any communication between participants. Any benefit or payoff is received by all participants; any cost is borne by the cooperators only. A well-known example is the *Tragedy of the Commons* (Hardin, 1968). In this dilemma, land (the commons) is freely available for farmers to use for grazing cattle. For any individual farmer, it is advantageous to use this resource rather than their own land. However, if all farmers adopt the same reasoning, the commons will be over-used and soon will be of no use to any of the participants, resulting in an outcome that is sub-optimal for all farmers.

In the N-player dilemma game there are N participants. Each player is confronted with a choice: to either cooperate or defect. We represent the payoff obtained by a strategy which defects given i cooperators as $D(i)$ and the payoff obtained by a cooperative strategy given i cooperators as $C(i)$.

Defection represents a dominant strategy, i.e. for any individual, moving from cooperation to defection is beneficial for that player (they still receive a benefit without the cost):

$$D(i) > D(i-1) \quad 0 < i \leq N-1 \tag{1}$$

$$C(i) > C(i-1) \quad 0 < i \leq N-1 \tag{2}$$

$$D(i) > C(i) \quad 0 < i \leq N-1 \tag{3}$$

However, if all participants adopt this dominant strategy, the resulting scenario is sub-optimal and, from a group point of view, an irrational outcome ensues:

$$C(N) > D(0) \tag{4}$$

If any player changes from defection to cooperation, the performance of the society improves, i.e. a society with $i+1$ cooperators attains a greater payoff than a society with i cooperators:

$$(i+1)C(i+1)+(N-i-1)D(i+1) > (i)C(i)+(N-i)D(i) \tag{5}$$

Small world Graphs

As mentioned in the introduction, small world graphs are a class of graphs or topologies such that nearly all nodes are reachable from all other nodes in a few steps. Watts and Strogatz (Watts, 1999) demonstrated that a regular lattice can be transformed into a small world network by making a small fraction of the connections random. The algorithm involves taking a regular lattice (ring, grid) and repeatedly removing some edge (a, b) and replacing it with an edge (a, c). If the node c is selected with probability based on its degree, then the notion of preferential attachment is present which results in a graph with node degree distribution following a power law.

The property of community structure has been reported in several real world networks (Newman and Girvan, 2004) and many algorithms have been proposed to measure the level of community structure present in the graph (Donetti and Munoz, 2004) (Zhang et al., 2007).

Such graphs have been used to constrain agent interactions in social dilemma games in interesting work (Wu et al., 2005), (Santos and Pacheo, 2005) which show that cooperation can be induced in 2-player games. Our work differs by addressing the N-player version which has been shown to be more challenging to induce cooperation in evolutionary settings (Yao and Darwen, 1994). We also show that the maintenance of one key property, that of community structure is of importance.

N-player dilemmas and Community Structure

In previous work, we have created a range of lattices which can be tuned to exhibit different levels of community structure (O'Riordan and Sorensen, 2008b). These graphs do not exhibit node degree distribution according to power laws (in fact, the degree is constant throughout the graph) and they also do not exhibit other small world properties. In the model previously adopted we created graphs with strongly connected clusters of agents who were loosely connected to neighbouring clusters. We varied the degree of community structure by simply varying the ratio of the weights on intra-community edges to intra-community edges. Agents were chosen to interact based on the strength of the edge weights. We allowed agents learn from their immediate neighbours; agents effectively imitated their more successful neighbours. If all immediate neighbours perform similarly, agents were allowed to learn from neighbouring clusters. We showed that cooperation emerged. Our initial model is discussed more in the following section.

Model

Initial Graph Topology

In the simulations described in this paper, agents are located on nodes of a graph. The graph is an undirected weighted graph. The weight associated with any edge between nodes represents the strength of the connection between the two agents located at the nodes. This determines the likelihood of these agents participating together in games.

The graph is static throughout the simulation: no nodes are added or removed and the edge weights remain constant.

We use a regular graph: all nodes have the same degree. In the initial topology, nodes have four neighbours. We use two

different edge weight values in each graph: one (a higher value) associated with the edges within a community and another (a lower value) associated with the edges joining agents in adjacent communities. All weights used in the this work are in range [0,1].

The graph is depicted in Fig. 1, where the thicker lines represent intra-community links (larger value as edge weight) and the thinner lines indicate inter-community links between neighbouring communities. The rectangles of thicker lines represent a community; the vertices represent agents.

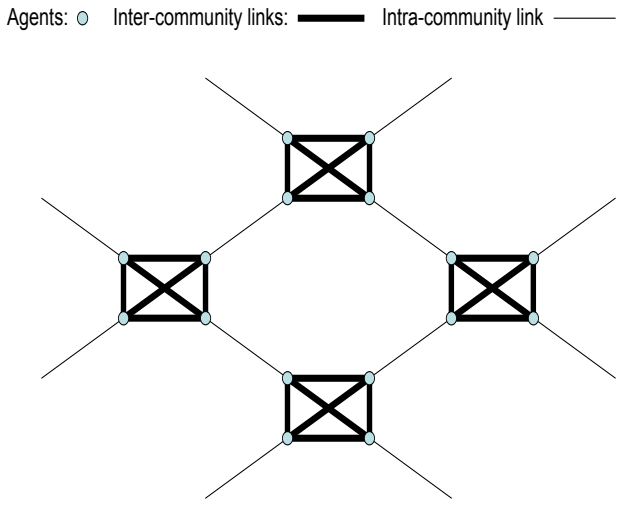


Figure 1: Graph with community structure

Agent Interactions

Interaction Model Agents in this model can have a strategy of either cooperation (C) or defection (D). Agents interact with their neighbours in a N-player prisoner's dilemma. The payoffs received by the agents are calculated according to the formula proposed by Boyd and Richerson (Boyd and Richerson, 1988), i.e. cooperators receive $Bi/N - c$ and defectors receive Bi/N, where B is a constant (in this paper, B is set to 5), i is the number of cooperators involved in the game, N is the number of participants and c is another constant (in this paper, c is set to 3).

Each agent may participate in several games. The algorithm proceeds as follows: for each agent a in the population, agents are selected from the immediate neighbourhood of agent a to participate in the game. Neighbouring agents are chosen to participate with a probability equal to the edge of the weight between the nodes. This means that, for a population with a high community structure, most games involve an agent's local community members. This allows a high degree of insulation from agents in neighbouring communities. An agent's fitness is calculated as the average payoff received in the interactions during a generation.

Learning

Agents may change their behaviours by comparing their payoff and that of neighbouring agents. We adopt a simple update rule whereby an agent updates their strategy to those used by more successful strategies. Following each round of games, agents are allowed to learn from their neighbours. Again these neighbours are chosen stochastically; the neighbours are chosen according to the weight of the edge between agent and neighbour.

We incorporate a second update mechanism. The motivation for its inclusion is as follows. Following several iterations of learning from local neighbours, each community is likely to be in a state of equilibrium—either total cooperation or total defection. Agents in these groups are receiving the same reward as their immediate neighbours. However, neighbouring communities may be receiving different payoffs. An agent that is equally fit as its immediate neighbours may look further afield to identify more successful strategies.

In the first update rule, agents consider other agents who are immediate neighbours. Let $s_adj(x)$ denote the immediate neighbours of agents x chosen stochastically according to edge weight. The probability of an agent x updating their strategy to be that of a neighbouring agent y is given by:

$$\frac{w(x,y).f(y)}{\Sigma_{z \in s_adj(x)} w(x,z).f(z)} \tag{6}$$

where $f(y)$ is the fitness of an agent y and $w(x,y)$ is the weight of the edge between x and y.

The second update rule allows agents to look further afield from their own location and consider the strategies and payoffs received by agents in this larger set, i.e. agents update to a strategy y according to:

$$\frac{w(x,y).f(y)}{\Sigma_{z \in adj(adj(x))} w(x,z).f(z)} \tag{7}$$

where again $f(y)$ is the fitness of agent y and now $w(x,z)$ refers to the weight of the path between x and z. We use the product of the edge weights as the path weight. Note that in the second rule, we don't choose the agents in proportion to their edge weight values; we instead consider the complete set of potential in the extended neighbourhood. In this way all agents in a community can be influenced by a neighbouring cooperative community.

Small World version of Graph

In order to create a graph topology more reflective of naturally occurring graphs, the basic graph topology must be changed. This is achieved by adopting the approach proposed by Watts (1999).

Our first approach involves taking our existing graph structure and re-attaching edges i.e. the following procedure is repeated: edge (a, b) is randomly selected from the set of edges present and replaced with the edge (a, c) where node c is selected in proportion to its degree. The new edge will have a weight equal to the deleted one. This approach, while introducing the small world property and the desired node degree distribution seriously damages the community structure. We hypothesise that this should negatively impact on the emergence of cooperation.

Our second approach begins with another regular graph structure; we place the communities of agents on a ring (depicted in Fig. 2). We again re-attach edges in the graph, but with the following constraint; only inter-community edges are deleted and re-attached. Thus, we choose an edge (a, b) randomly such that both a and b are on the circumference of the ring; this edge is deleted and re-attached as (a, c) such that c is selected randomly from those nodes positioned on the circumference. Again, the new edge will have a weight equal to the deleted one. This approach maintains the community structure in the graph while introducing the desired small world graph properties.

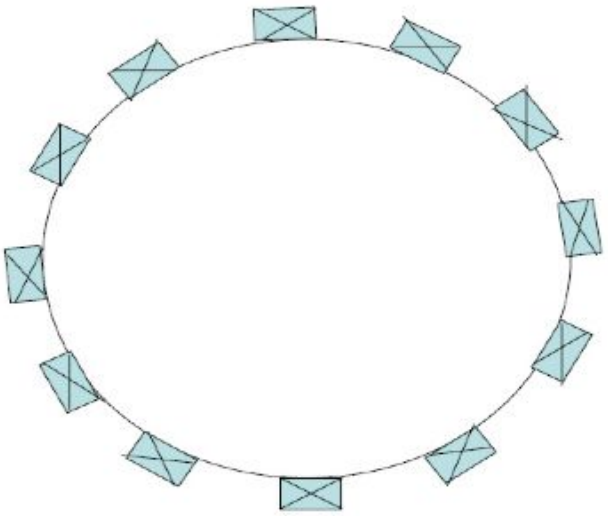


Figure 2: Ring structure with communities

The following tables present some data to illustrate some of the properties of the resulting graphs for the two different algorithms.

Experiment Setup

A population of 800 agents is used. Strategies are assigned to agents randomly. We allow simulations to run for 200 generations.

Following each generation, the first learning rule is applied. Following every four generations (sufficient for community to reach an equilibrium), the second learning rule is applied. This is not necessary in most cases; we merely choose to let the local interactions stabilise prior to applying the second rule. This eases analysis in some cases where fluctuations can occur if community structure levels are not sufficiently high. We include another plot in a later section in this paper where we use the ring graph with re-attachment and modify the rates of application of learning rules such that they both occur every generation. The outcome is similar.

In all the simulations we initially enforce a high level of community structure; the intra-community links are held constant with a value of one. The value of the inter-community are set to 0.1. The lower the value, the more insulated clusters are and hence should promote cooperation.

We vary the level of re-attachment and measure the resulting levels of cooperation.

Re-attachment Rate	Dev. in Node Degree	Av. Diameter
0%	0	15.59
1%	0.261	12.14
5%	0.620	8.52
10%	0.835	7.49

Table 1: Properties of resulting small world graph using first algorithm (initial lattice structure, all edges considered for re-attachment) for different levels of re-attachment

Re-attachment Rate	Dev. in Node Degree	Av. Diameter
0%	0.5	132.9
1%	0.512	41.17
5%	0.532	5.655
10%	0.589	2.136

Table 2: Properties of resulting small world graph using second algorithm (initial ring structure, inter-community edges considered for re-attachment) for different levels of re-attachment

Results

Emergence of Cooperation

For the first graph structure (regular lattice) with different levels of edge re-attachment, we see that the levels of cooperation is dependent on the degree of re-attachment present (see Fig. 3). For a regular graph with high community structure and no other small world properties, we see that the population quickly converges to cooperation. Introducing 1% re-attachment reduces the diameter of the graph and increases the node degree deviation but also damages the level of community structure. We see that the levels of cooperation reached fall to roughly 700 cooperators in the population. As the level of re-attachment increases, the ef-

fect becomes even more pronounced with a big decrease in the number of cooperators for re-attachment level of 5% and a large collapse in the number of cooperators for re-attachment levels of 10%.

It is worth commenting on the nature of the fluctuations in the separate runs. Consider, as an example, the line indicating re-attachment levels of 5% where the levels of cooperation fluctuate considerably. This is due to the effect of the two learning rules and the frequency with which they are applied. Following a few generations, each community converges to total cooperation or total defection. Following every fourth generation, the second rule is applied with leads to an immediate increase in the number of cooperators as members of non-cooperating clusters imitate more successful clusters. These new cooperators are in most cases interacting with non-cooperators and hence are exploited by their immediate neighbours. These immediate neighbours are then imitated leading to emergence of defection in these clusters.

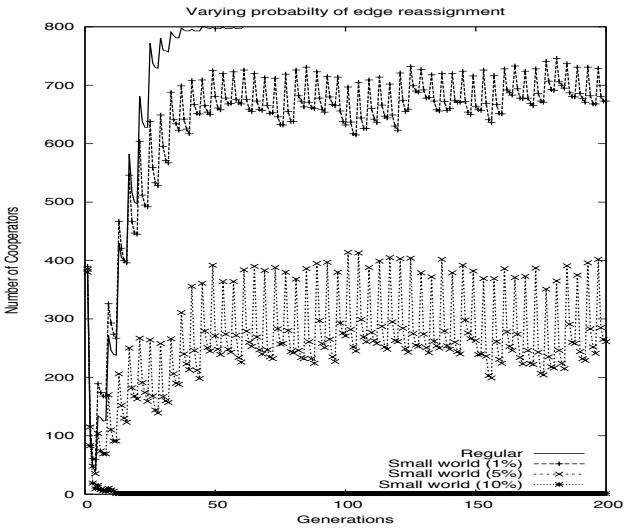


Figure 3: Levels of Cooperation present in population placed on small world graph created from orginal lattice

Fig. 4 shows the levels of cooperation attained given a graph with small world properties that also has initially a high level of community structure created by re-attaching inter-community edges only.

We see that for levels of re-attachment up to 10%, cooperation still emerges. These results illustrate that we can have small world properties (e.g. small diameter) and still maintain community structure and hence maintain high levels of cooperation.

An interesting point to note is that cooperation reaches the maximum possible level most quickly for re-attachment levels of 5%. This is due to the reduction in the diameter which causes cooperation to spread more quickly as non-cooperative clusters are more likely to be close to cooperative clusters. However, increasing the level of re-attachment further slows down the spread of cooperation. This is because, despite the potential gain caused by the decrease in diameter, the increased probability of having a number of nodes with a high degree which can be influenced more readily by non-cooperating strategies.

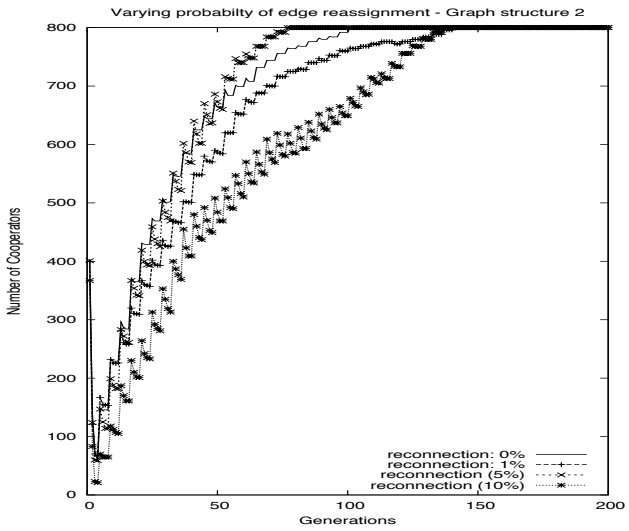


Figure 4: Levels of Cooperation present in population placed on small world graph created from ring; only inter-community links re-attached.

Robustness

In many scenarios that we may wish to model, it is possible for uncertainty or noise to exist—agents may perform their acts incorrectly or imperfectly, their acts may be misinterpreted, agents may learn or imitate others and change their behaviour accordingly, and agents may exit or join the group thereby changing the environment or others. Alternatively, the environment may change dramatically thereby requiring agents to explore and learn new suitable behaviours.

In previous work O'Riordan and Sorensen (2008a), we showed that these graph structures allowed populations to be robust to noise and to dramatically changing environments for a population of generalised *tit-for-tat* strategies. In this experiment we explore again if the population of agents can survive and track dramatic environmental change. We introduced 1% noise to ensure some exploration of the strategy space. At every generation, each agent has a 1% probability of changing strategy. We also introduce dramatic environmental change during the simulation. This involves reversing the payoffs of the game which causes 'cooperation' now to be viewed as individually rational and collectively suboptimal and renders 'defection' the new socially beneficial and cooperative act.

In our simulation, we count and plot the number of agents choosing the socially beneficial action. We see that following the change in environment at generation 350, the population recovers to high levels of cooperation (Fig. 5).

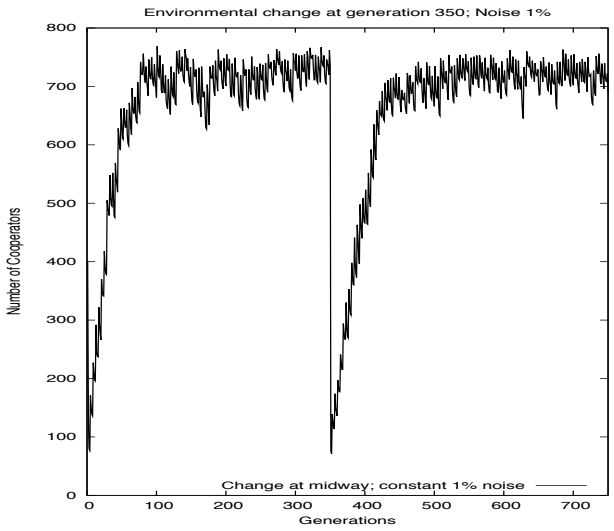


Figure 5: Levels of cooperation present on ring with re-attachment probability 5%. Noise is set to 1%. Dramatic environment reversal occurs at generation 350

Conclusions

In this paper, we wished to explore if cooperation can emerge among self-interested agents participating in N-player social dilemmas where the agents are placed on a small world network exhibiting community structure. Our previous work illustrated the emergence of cooperation given a community structure on regular lattices. In this paper, we showed that by converting the graph to a small world network by re-attaching edges in such a manner that damaged the community structure, cooperation collapses and defection emerges as the norm. We also showed that by converting a regular graph to a small work graph while taking care to preserve the community structure, cooperation can emerge as the norm. The speed of the emergence can also be improved by having small world properties (e.g. reduced diameter).

We have shown that the notion of community structure is a key feature in the emergence of cooperation. In order for cooperative clusters to survive, the agents must be able to protect themselves from non-cooperative agents by insulating themselves and playing mainly among themselves. However, communities or clusters cannot be totally isolated; they must have some link to other communities so as to provide an opportunity to learn more beneficial strategies if possible. A balance must be struck between the risk of exploitation and the potential to learn a better strategy if one exists.

Discussion

In the experiments in this paper, we utilise two learning rules—one involving an agent's immediate neighbours, the other involving an extended neighbourhood. We allow the agents to learn from the immediate neighbours first and then upon reaching an equilibrium we allow them to learn from the neighbouring communities. We achieve this by allowing the first learning rule every generation and the second learning rule every four generations.

The motivations were primarily to allow local communities reach an equilibrium prior to learning from others as otherwise, in some cases, this causes fluctuations in levels of cooperation and convergence is never reached. This occurs when there are insufficient levels of community structure. In these cases, a local community may be heading towards to defection and then learn from neighbours and heads towards cooperation again etc. We wished to ease the complexity of the interactions for these specific cases.

However, it should be noted for the results presented for the main graph of interest (the ring transformed into a small world graph while maintaining the level of community structure), the rates of application of the learning rules does not dramatically interfere with the results. The same trends are noticed (Fig. 6) where we apply both learning rules every generation. The agents reached the state of total cooperation much more quickly due to the application of the second learning rule every generation.

Future Work

There are several directions for future work. One future direction would be to explore the generalizability of these results. We wish to explore different updating schemes and other social dilemma games to explore if the same effects are detected. We also wish to explore these graph structures in uncertain environments. Another track which we will pursue is to investigate under which conditions these graphs can emerge based on agent interactions. In this paper, agent interactions are constrained by the graph properties. It would be interesting to show that such graphs can emerge based on interactions between agents.

References

Birk, A. (1999). Evolution of continuous degrees of cooperation in an n-player iterated prisoner's dilemma. *Technical Report, Vrije Universiteit, Brussel, AI-Laboratory, 1999*.

Boyd, R. and Richerson, P. (1988). The Evolution of Reciprocity in Sizable Groups. *Journal of Theoretical Biology*, 132:337–356.

Chiba, K. and Hiraishi, K. (1998). Iterated continuous prisoner's dilemma game and its usefulness in analyzing multi-agent systems. *IEEE International Conference on Systems, Man, and Cybernetics*, 1:644– 649.

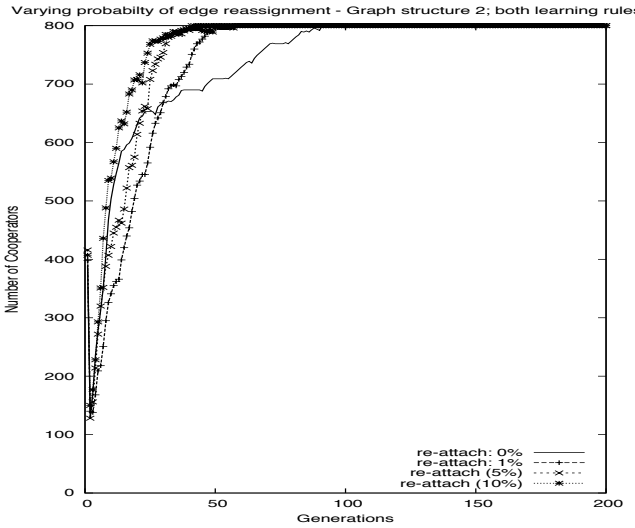


Figure 6: Small world ring; both learning rules applied each generation

Davis, J., Laughlin, P., and Komorita, S. (1976). The social psychology of small groups: Cooperative and Mixed-Motive Interaction. *Annual review of Psychology*, 27:501–541.

Donetti, L. and Munoz, M. (2004). Detecting network communities: a new systematic and powerful algorithm. *Journal of Statistical Mechanics, P10012*.

Hardin, G. (1968). The tragedy of the commons. *Science*, 162(3859):1243–1248.

Hauert, C. (2006). Spatial Effects in Social Dilemmas. *Journal of Theoretical Biology*, 240(4):627–636.

Lozano, S., Arenas, A., and Sanchez, A. (2006). Mesoscopic structure conditions the emergence of cooperation on social networks. *arXiv:physics/0612124v1*.

Newman, M. E. J. and Girvan, M. (2004). Finding and evaluating community structure in networks. *Physics Review E*, 69.

O'Riordan, C. and Sorensen, H. (2008a). Stable cooperation in changing environments (short paper). *7th International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)*.

O'Riordan, C. and Sorensen, H. (2008b). Stable cooperation in the n-player prisoner's dilemma: the importance of community structure. *Lecture Notes in Artificial Intelligence LNAI 4865, Adaptive Agents and Multi-Agent Systems III*.

Santos, F. and Pacheo, J. (2005). Scale-free Networks Provide a Unifying Framework for the Emergence of Cooperation. *Physical Review Letters*, 95(9).

Watts, D. J. (1999). *Small Worlds: The Dynamics of Networks Between Order and Randomness*. Princeton University Press.

Wu, Z., Xu, X., Chen, Y., and Wang, Y. (2005). Spatial prisoner's dilemma game with volunteering in newman-watts small-world networks. *Physical Review E*, 71:37103.

Yao, X. and Darwen, P. J. (1994). An experimental study of N-person iterated prisoner's dilemma games. *Informatica*, 18:435–450.

Zhang, S., Wang, R., and Zhang, X. (2007). Identification of overlapping community structure in complex networks using fuzzy *c*-means clustering. *Physica A*, 374:483–490.

Optimal noise in spiking neural networks for the detection of chemicals by simulated agents

Nicolas Oros, Volker Steuber, Neil Davey, Lola Cañamero, Rod Adams

Science and Technology Research Institute
University of Hertfordshire
AL10 9AB
United Kingdom
{N.Oros, V.Steuber, N.Davey, L.Canamero, R.G.Adams}@herts.ac.uk

Abstract

We created a spiking neural controller for an agent that could use two different types of information encoding strategies depending on the level of chemical concentration present in the environment. The first goal of this research was to create a simulated agent that could react and stay within a region where there were two different overlapping chemicals having uniform concentrations. The agent was controlled by a spiking neural network that encoded sensory information using temporal coincidence of incoming spikes when the level of chemical concentration was low, and as firing rates at high level of concentration. With this architecture, we could study synchronization of firing in a simple manner and see its effect on the agent's behaviour. The next experiment we did was to use a more realistic model by having an environment composed of concentration gradients and by adding input current noise to all neurons. We used a realistic model of diffusive noise and showed that it could improve the agent's behaviour if used within a certain range. Therefore, an agent with neuronal noise was better able to stay within the chemical concentration than an agent without.

Introduction

Animals are able to detect and react to chemicals (odours, pheromones…) present in the environment. The key sense to detect these chemical cues is smell rather than taste (Wyatt, 2003). Almost all animals have a similar olfactory system including olfactory sensory neurons (OSN) that are exposed to the outside world and linked directly to the brain. Pheromones and other odour molecules present in the environment are converted into signals in the brain by first binding to the olfactory receptor protein situated in the cell membrane of the OSN. Spikes are then sent down the axon of the OSN (Kandel et al., 2000). A chemical blend is composed of many molecules that can be detected with tuned odour receptors and therefore, activates a large range of olfactory sensory neurons. Odours are coded by which neurons emit spikes and also by the firing patterns of those neurons sending spikes to others during and after the stimulus. In many vertebrates and insects, oscillations of the neural activity have been recorded in the olfactory systems (Wyatt, 2003). Therefore, the synchronization of firing between different sensory neurons seems to be very important for odour perception and interpretation. The firing rate and the number of sensory neurons are also important in odour recognition when stronger stimuli increase the frequency of firing of individual sensory neurons but also stimulate a larger number of them.

Different studies have been done on the perception of simulated chemicals using artificial neural networks where neural synchronization occurs (Brody & Hopfield, 2003; Hopfield, 1999; Hoshino et al., 1998) and also using robots (Kanzaki et al., 2005; Kuwana & Shimoyama, 1998; Payton et al., 2001; Pyk et al., 2006; Webb, 1998). We were interested in studying the perception and the behaviour of an agent in response to changes of its environment. The primary research question is how two encoding strategies can be used to integrate sensory information in order to control a simulated agent. To the best of our knowledge, no neural architecture, controlling a simulated agent, has been created that encodes the sensory information onto both the firing rate and the synchronization of firing (temporal coincidence of incoming spikes) depending on the environment. As the interaction between the two encoding strategies is complex, we decided to create a simple architecture using a spiking neural network. This model could encode the sensory information onto both the firing rate and the synchronization of firing depending on the environment. The neural network controlled the agent by encoding the sensory information onto temporal coincidences in a low concentration environment, and firing rates at high concentration.

It is well known that real neuronal systems contain noise (Kandel et al., 2000) which may improve the brain's ability to process information, a phenomenon also called stochastic resonance (Hänggi, 2002; Mori & Kai, 2002; Moss et al., 2004; Wiesenfeld & Moss, 1995). Researchers in robotics and artificial life have already implemented simple models of neural noise (Di Paolo, 2003; Florian, 2006; Jacobi et al., 1995). Here we study the effect of a more realistic noise model based on a diffusive OU (Ornstein-Uhlenbeck) process (Uhlenbeck & Ornstein, 1930). We added this noise in the neural network and studied its effect on the behaviour of the agent. Our results suggest a potential function for noise in real

biological systems, and highlight that features of biological systems can be used to construct better agents.

Environment

We created a simulation of a continuous world including an agent and a maximum of two chemicals. We decided to use a simple model of chemicals that are not diffused and evaporated but with concentrations that can be calculated directly at any given point. Our agent was equipped with two antennae and a differential steering system using two wheels. The two antennae were separated widely enough to detect the presence of the chemical concentration (Fig. 1). The left and right wheels were situated on the sides of the agents. To control the agent, we had to decide which neurons' model to use in order to study firing synchronization of the sensors.

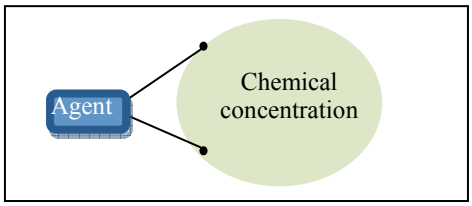


Figure 1. An agent equipped with two wheels and two antennae used to detect chemicals.

Neural Network

There are three main ways to encode the intensity of sensory information into spiking neurons based on biological evidences (Floreano & Mattiussi, 2001; Florian, 2003; Gerstner & Kistler, 2002; Izhikevich, 2003, 2004; Koch, 1999) . The most commonly used method consists of mapping the stimulus intensity to the firing rate of the neuron (firing rate encoding). Another method encodes the intensity of the stimulation into the number of spikes sent by different neurons arriving at a pre-synaptic neuron at the same time (firing synchronization or temporal coincidence encoding). The last main encoding scheme maps the strength of the stimulation in the firing delay of the neuron (delay encoding). As we saw earlier, spatial configuration is an important feature in odour recognition of neurons as is the synchronization of firing between neurons (Kandel et al., 2000; Laurent et al., 1996; Wyatt, 2003). J. Hopfield and C. Brody (Brody & Hopfield, 2003; Hopfield, 1999) created simple neural networks using spiking neurons to simulate an olfactory process. In their system, the recognition of an odour was signalled by spike synchronization in artificial glomeruli. In our system, the neural network was supposed to detect the blend of two different chemicals and modify the agent's behaviour. We used a model of neural network that allowed us to study synchronization of firing in a simple manner. The neural network could control the agent by encoding the sensory information onto temporal coincidences in a low concentration environment, and firing rates at high concentration.

Models of Spiking Neurons

It is well known that compared to the complex and computationally slow Hodgkin and Huxley model, simple spiking models like integrate-and-fire neurons can run quickly enough and have a more realistic behaviour than firing rate ones (Floreano & Mattiussi, 2001; Florian, 2003; Gerstner & Kistler, 2002; Izhikevich, 2003, 2004; Koch, 1999). This is why more and more researchers are implementing spiking neurons in robots and simulated agents. Therefore, we decided to use a simple model of a spiking neuron. Our model is based on a leaky-integrator model which includes synaptic integration and conduction delays. The idea is that a spike sent by a neuron will take some time to arrive at another neuron. This time delay depends on the distance between the sender and the receiver. All the spikes arriving at a neuron are summed to calculate the neuron's input current density (in Amperes per Farad) and membrane potential (in Volts) after every time step ($\Delta t = 0.1ms$). Once the membrane potential reaches a certain threshold θ, the neuron will fire and then will be set to 0 for a certain time (refractory period). During this time, the neuron cannot fire another spike even if it is highly stimulated.

Many real neurons' membrane potential is around -70mV during resting state. When a neuron fires, its membrane potential will increase rapidly to about 30mV, so the height of a typical spike is approximately 100mV (Kandel et al., 2000). We set the resting potential to 0 and the potential of a spike to 100mV. It is reasonable to set the neuron's threshold at 20mV, the refractory period to 3ms and the membrane time constant τ_m to 50ms (Kandel et al., 2000). We also decided to set a synaptic time constant τ_s to 2ms: a spike that arrives at a synapse triggers a current given by:

$$I_j(t) = \left(\frac{t-(t_{spike}+delay)}{\tau_s}\right) exp\left(1 - \frac{t-(t_{spike}+delay)}{\tau_s}\right) \quad (1)$$

where $I_j(t)$ is the synaptic input current, t_{spike} corresponds to the time a spike has been sent to the neuron, $delay$ is the time delay in seconds before the spike arrives to the neuron (*delay* = *coeff_delay* * *distance*) with *coeff_delay* $= 5.10^{-5}$.

The change of membrane potential is given by:

$$\frac{dV}{dT} = -\left(\frac{V}{\tau_m}\right) + \sum(I_j W_j) \quad (2)$$

where V is the membrane potential, τ_m is the membrane time constant and $W_j$the synaptic weight.

Sensory Neurons

We created a model of a spiking sensory neuron in which the chemical concentration is processed so that a quasi-linear relationship between the concentration and the firing rate of the sensor is produced (Oros et al., 2008). Such relationships exist in biological systems. For example in humans, the relationship between the frequency of firing and pressure on the skin is linear (Kandel et al., 2000). We used a two step process where two biologically realistic non-linear mappings between sensory information and input current and between

input current and firing rate results in a linear relationship. Researchers in robotics and artificial life use a linear direct mapping between the sensory information and the firing rate (Di Paolo, 2002, 2003; Florian, 2006). The sensory neurons used in our model are able to encode the stimulus intensity, measured at the tip of the antenna, into sensory input current using a biologically plausible sigmoid function (Oros et al., 2008). This current is injected to the sensor's membrane potential that increases, making the sensor fire into appropriate firing rates. Therefore, the sensory neurons encode the concentration value onto the appropriate firing rate. The sensors were configured in order to distinguish a large range of concentrations between 1 and 300. Over 300, they were saturating.

Motor Neurons

We decided that, in order to move, the agent should be driven by two wheels each controlled by two motor neurons: one to go forward, one to go backward. We created sensors able to detect a chemical gradient. But an agent equipped with such sensors will not move without any stimulus. So we decided for simplicity that an agent should always move forward in the absence of any external input. We performed this by adding a small baseline input current (0.5 A/F) in the motor neurons responsible to go forward. The final velocity of the wheels was calculated by subtracting the firing rate of the motor neurons, responsible for moving the agent forward and backward, running over a certain period of time. The agent was moved by calculating the velocity every 10ms.

Temporal Coincidence

We used the agent and world described above. The environment contained either one or two chemicals denoted by A or B. In this experiment, each chemical source had a circular shape and the same fixed value all over its surface. One agent, placed in the world, was controlled by a simple spiking neural network implementing the neurons described in the previous section.

The neural controller was based on a Braitenberg vehicle (anger behaviour) (Braitenberg, 1984) where an agent moves faster toward a stimulus when it detects it (Fig. 2).

Our hypothesis was that by using this architecture, the sensory neurons needed to encode the sensory information onto the firing rates, and also onto temporal coincidences between spikes sent by sensors. To verify this hypothesis, we performed three series of tests to study the effect of the starting positions, the sensory delays and the value of the concentrations on the agent's behaviour.

Experiment I

The first test was to study the effect of the agent's starting position on its behaviour. Both concentration values for the chemicals A and B were set to be low. In all the experiments described in this paper, the concentration range was from 1 to 300. In this instance, A and B concentrations were set to 1 or 2. We tried ten different starting positions and five different settings for the environment: with one chemical A, one chemical B, and finally one concentration of the chemical A overlapping with one concentration of the chemical B. Each run lasted 600 seconds and the neural network was updated every 0.1ms (so the run lasted 6,000,000 time steps). Every 10ms, the agent was moved and the sensory inputs updated.

In these experiments, the agent could detect double concentrations of one chemical (A or B) but did not react to it. However, the agent was able to react only to the blend of both chemicals A and B, where it stayed inside the overlapping concentrations. We recorded the current density and membrane potential of the neuron N0 during a small interval of time when the agent was inside the blend of chemicals A and B (Fig. 3, top). The input current of the neuron N0 was increasing when spikes coming from both S2 and S3 arrived at the same time. Then, the membrane potential also increased and reached the threshold θ (0.0046 Volts) making the neuron N0 fire. The potential was then set to 0 during the refractory period. As the sensors were synchronized and the delay between them and the neurons were the same, the spikes arrived at the same time to the neuron allowing it to detect them and fire.

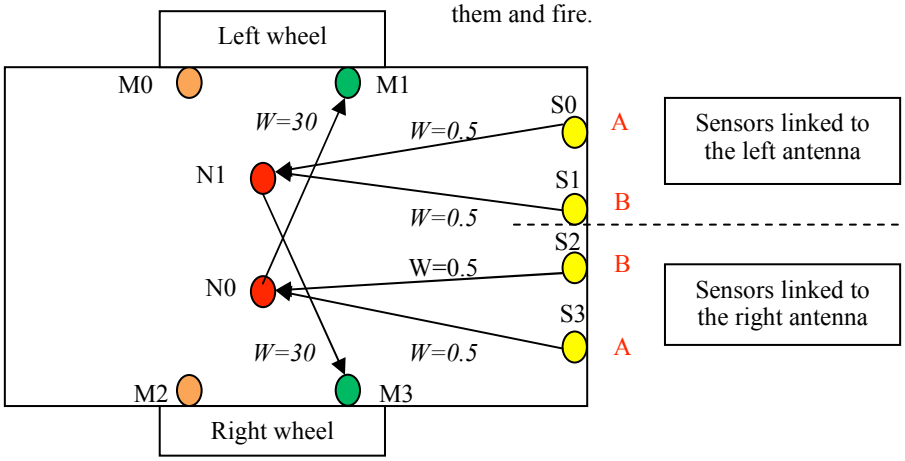


Figure 2. Agent's neural controller. The sensors S0 and S3 detect the chemical A and the sensors S1 and S2 detect the chemical B. The sensory axons' lengths are all similar (delays = 2.5ms). The motor neurons M1 and M3 are responsible to move the agent forward. The threshold of the neurons (N0 and N1) was set to 4.6 mV. *W* is the synaptic weight.

Experiment II

The second experiment was to test our hypothesis by modifying the sensory response delays to verify that our architecture necessarily needed to encode the sensory information onto temporal coincidence. We changed the delays by modifying the position of the sensors therefore modifying the length of their axons linked to the neurons. We only changed the delays of the sensors detecting the chemical B (S1 and S2).

We used one of the Experiment I 's setups where the agent was staying in the chemical blend of the chemicals A and B having a concentration of 1 each. We tried different values of delays (from 1ms to 50ms) and we noticed that a small change (up to 7.5ms) did not modify the agent's behaviour. But a further change in the delays (from 7.5ms) made the agent unable to react to the blend of chemicals A and B so it could not stay inside the concentrations.

As in the Experiment I, we recorded the current density and membrane potential of the neuron N0 during 0.5s when the agent was inside the chemical blend.

In Figure 3 (bottom), we can see that the current of the neuron N0 increases when a spike coming from both S2 and S3 arrive but as the delay has been changed, the spikes do not arrive at the same time so the current is lower than in Experiment I. Therefore, the neuron's potential increases but never reaches the threshold so the neuron does not fire (Fig. 3, bottom).

Experiment III

In order to investigate the use of firing rate encoding, we used only one concentration of either A or B and increased it. When the concentration was augmented from 1 to above 50, the agent was then able to react to it. Therefore, the neural network showed much more sensitivity to two chemicals than to one. We also realized when using two overlapping chemicals A and B, as the concentration value increased, modifying the delays had a minor effect and the agent was still able to react to the chemicals. The firing rates were increasing too so the agent was moving faster. In these experiments, the temporal coincidence encoding was not necessary. The sensory information was encoded onto the firing rates of the sensors.

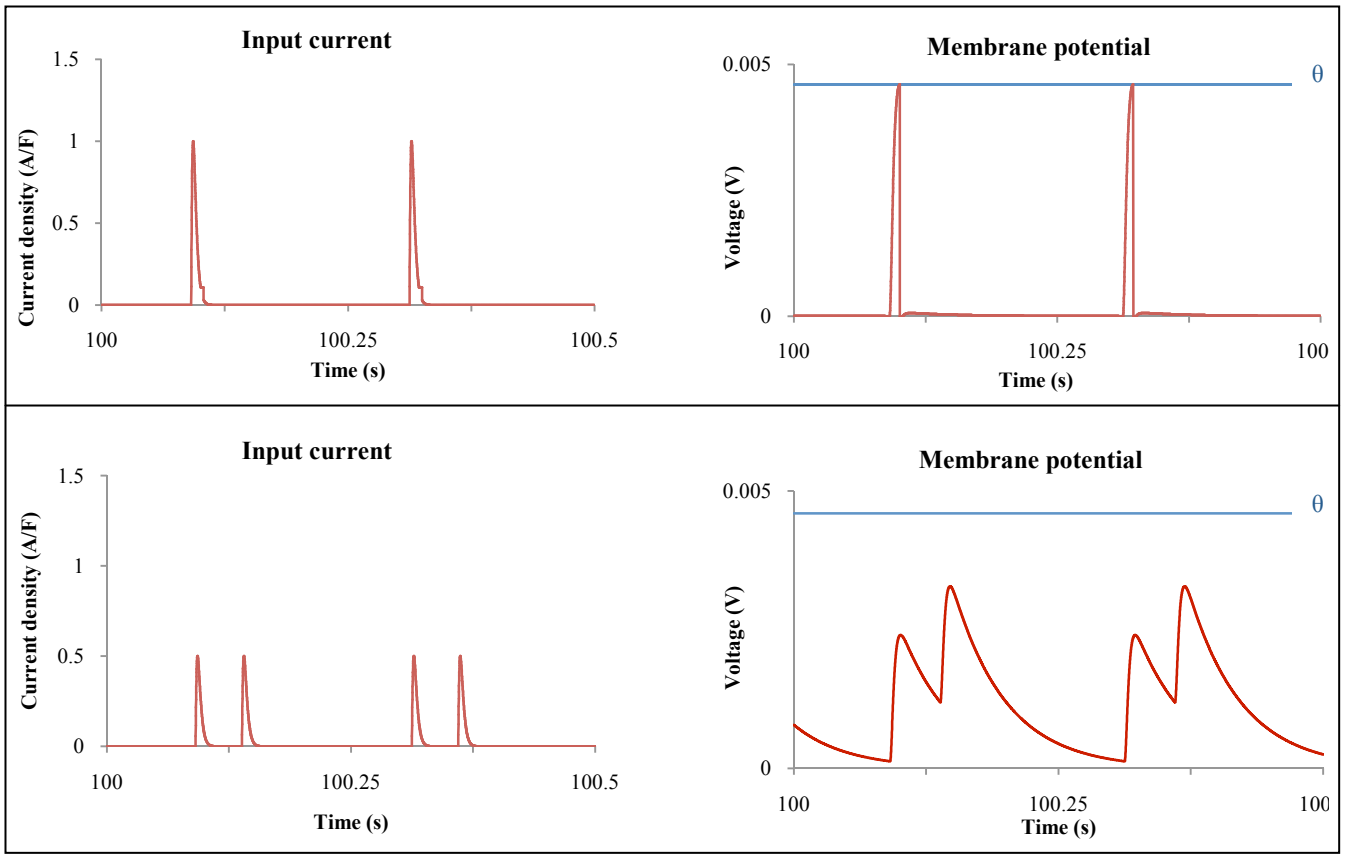


Figure 3. Current density (in Amperes per Farad) and membrane potential (in Volts) of the neuron N0 recorded between 100s and 100.5s. On the top panel (Experiment I), the spikes sent by the sensors arrived at the same time increasing the current density to 1 A/F. The membrane potential was then increased and reached the threshold making the neuron N0 fire. On the bottom panel (Experiment II), the spikes sent by the sensors were not coincident as the delays between the sensors (S1 and S2) and the neurons (N0 and N1) were changed (to 50ms in this case). Therefore the current was never above 0.5 A/F so the membrane potential could not reach the threshold to make the neuron N0 fire.

Diffusive noise

In the previous experiments, we presented a simple neural architecture where temporal coincidence and firing rate encoding strategies were both important mechanisms used in different environmental settings. In a low concentration setting, synchronization of spikes sent by the sensors was essential to allow the agent to detect the blend of two chemicals. We changed the sensory delays and noticed that the agent was then not able to react to the chemicals anymore. In a high concentration setting, the temporal coincidence between the firing of the sensors was not a necessary condition and the agent was able to stay inside the chemical concentration using just a firing rate encoding strategy. Interestingly, the model showed much more sensitivity to the presence of two chemicals than a single chemical. To this point, we have used uniform concentrations to simplify the study of the different encoding strategies. However, this model of chemical concentration was not realistic, so we decided to use an environment comprising two non uniform chemical concentration gradients. We tested our architecture in the new environment and noticed that the agent moved outside the concentration when its trajectory was along the direction of the gradient since both of its antennae where instantaneously outside the chemical concentrations. For this reason, we decided to add noise to the neural network.

We used a realistic model of noise in the form of an diffusive OU current (Uhlenbeck & Ornstein, 1930). This form of colored noise characterizes the subthreshold voltage fluctuations in real neuronal membranes (Rudolph & Destexhe, 2003). We added this noise to the total current calculated in Equation (2) in each neuron. The noise is described by:

$$\frac{dI(t)}{dt} = -\frac{1}{\tau_I}\left(I(t) - I_0\right) + \sqrt{\frac{2\sigma^2}{\tau_I}}\xi(t) \qquad (3)$$

where τ_I denotes the current noise time constant (2ms in our case), I_0 is the mean synaptic current (0 in our case), σ is the noise diffusion coefficient and $\xi(t)$ is a white Gaussian noise (with mean = 0 and standard deviation = 1).

We performed different series of tests to find appropriate level of noise, by modifying σ, in order to have an agent that stays in the gradient chemical blend. We placed the agent at three different positions (Fig. 6) and tried eight different levels of noise (Fig. 4 and 5). For each level, we performed 100 runs per position. Each run lasted 300s and we recorded the fitness of an agent during the last 100s. The fitness function was very simple and consisted of the sum of the distance between the agent and the centre of the concentrations measured every time the agent moved. The maximum value of both concentrations was set to 25.

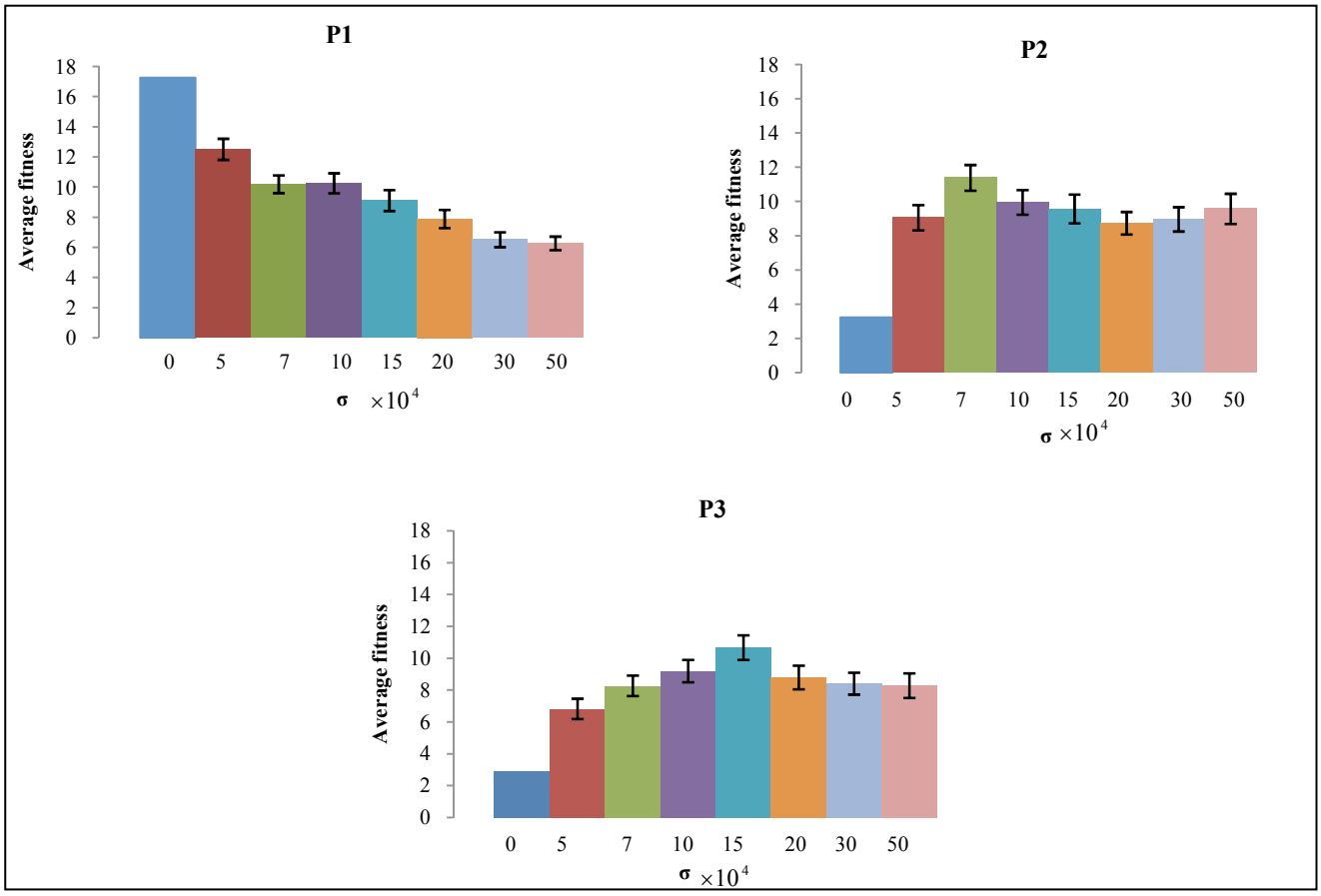


Figure 4. Mean fitness values recorded during 100s for an agent starting at the positions P1, P2 and P3 using different levels of noise ($\sigma \times 10^4$). The error bars represent standard errors.

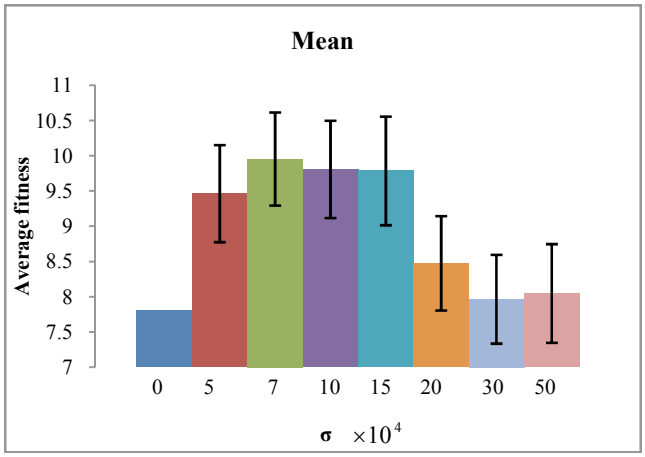


Figure 5. Mean of the fitness values displayed in Figure 4. ($\sigma \times 10^4$).

By looking at Figures 4 and 5, we can see that when the agent was starting from P2 or P3, an appropriate level of noise allowed it to stay within the concentration having a higher fitness than an agent without neural noise. We also note that the level of noise needed to be within a certain range as a low value did not improve the agent's behaviour and a high value disturbed it. We noticed as well that the agent was more sensitive to noise in low concentration areas than in high concentration areas.

Conclusion

We first presented in this paper a simple neural architecture where temporal coincidence and firing rate encoding strategies were both important mechanisms used in different environmental settings. In a low concentration setting, synchronization of spikes sent by the sensors was essential to allow the agent to detect the blend of two chemicals. We changed the sensory delays and noticed that the agent was then not able to react to the chemicals anymore. In a high concentration setting, the temporal coincidence between sensors firing was not a necessary condition and the agent was able to stay inside the chemical concentration using just the firing rate encoding strategy. Interestingly, the model showed much more sensitivity to the presence of two chemicals than a single chemical. Our results showed that a spiking neural network could be used to control an agent and could encode external stimuli in more than one way. The second study was on the effect of noise on the agent's behaviour using the same neural architecture. We used a more complex environment using chemical gradients and a realistic model of neural noise. We found that the overall fitness of the agent was better when a certain amount of noise was added in the neural network. Our results suggest that a realistic model of noise can improve an agent's behaviour. This is further evidence that adding biologically realistic features can be beneficial for certain engineering tasks, and suggests a potential function of noise in real biological systems. The effect of biologically realistic noise should be an interesting topic of research in other artificial life scenarios.

Our future work will be to see if we can evolve such architecture using a developmental model (evolving the number of neurons and their connections, the synaptic weights, and delays of the neural network).

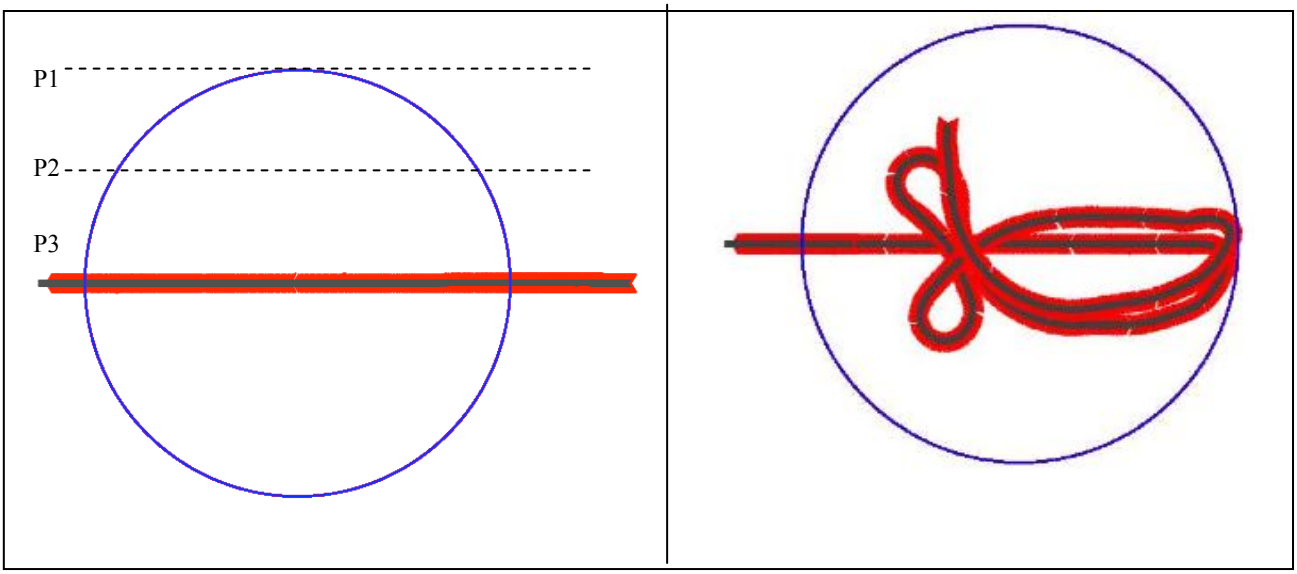


Figure 6. Left panel: path of an agent moving across the blend of chemicals A and B. The agent's neural controller doesn't have any noise so the agent goes straight as both of its antennae arrived at the same time outside the concentration. Right panel: path of an agent running over 300s. The agent's neural controller has noise so the agent does not go exactly in a straight line and therefore, can react to the absence of the chemical concentration to stay inside.

References

Braitenberg, V. (1984). *Vehicles: Experiments in Synthetic Psychology*: MIT Press, Cambridge, MA.

Brody, C. D., & Hopfield, J. J. (2003). Simple Networks for Spike-Timing-Based Computation, with Application to Olfactory Processing. *Neuron, 37*, 843-852.

Di Paolo, E. A. (2002). *Evolving spike-timing dependent plasticity for robot control.* Paper presented at the EPSRC/BBSRC International Workshop: Biologically-inspired Robotics, The Legacy of W. Grey Walter, WGW'2002, Bristol.

Di Paolo, E. A. (2003). Spike-Timing Dependent Plasticity for Evolved Robots. *Adaptive Behavior, 10*, 243-263.

Floreano, D., & Mattiussi, C. (2001). Evolution of Spiking Neural Controllers for Autonomous Vision-Based Robots. *Proceedings of the international Symposium on Evolutionary Robotics From intelligent Robotics To Artificial Life, 2217, T. Gomi, Ed. Lecture Notes In Computer Science Springer-Verlag, London, 38-61.*

Florian, R. V. (2003). Biologically inspired neural networks for the control of embodied agents. *Technical report Coneural-03-03 Version 1.0.*

Florian, R. V. (2006). *Spiking neural controllers for pushing objects around.* Paper presented at the Proceedings of the Ninth International Conference on the Simulation of Adaptive Behavior (SAB'06), Rome.

Gerstner, W., & Kistler, W. M. (2002). *Spiking Neuron Models. Single Neurons, Populations, Plasticity*: Cambridge University Press.

Hänggi, P. (2002). Stochastic resonance in biology: how noise can enhance detection of weak signals and help improve biological information processing. *Chemphyschem 3*(3), 285-290.

Hopfield, J. J. (1999). Odor space and olfactory processing: Collective algorithms and neural implementation. *PNAS., 96*(22), 12506-12511.

Hoshino, O., Kashimori, Y., & Kambara, T. (1998). An olfactory recognition model based on spatio-temporal encoding of odor quality in the olfactory bulb. *Biological Cybernetics, 79*, 109-120.

Izhikevich, E. M. (2003). Simple model of spiking neurons. *IEEE Transactions on Neural Networks*.

Izhikevich, E. M. (2004). Which model to use for cortical spiking neurons. *IEEE Transactions on Neural Networks*.

Jacobi, N., Husbands, P., & Harvey, I. (1995). Noise and the Reality Gap: The Use of Simulation in Evolutionary Robotics, *Proceedings of the Third European Conference on Advances in Artificial Life*: Springer-Verlag.

Kandel, E. R., Schwartz, J. H., & Jessell, T. M. (2000). *Principles of Neural Science* (4th ed.): McGraw-Hill.

Kanzaki, R., Nagasawa, S., & Shimoyama, I. (2005). Neural Basis of Odor-source Searching Behavior in Insect Brain Systems Evaluated with a Mobile Robot. *Chemical Senses, 30*(1), 285-286.

Koch, C. (1999). *Biophysics of Computation: Information Processing in Single Neurons*: Oxford University Press: New York, New York.

Kuwana, Y., & Shimoyama, I. (1998). A Pheromone-Guided Mobile Robot that Behaves like a Silkworm Moth with Living Antennae as Pheromone Sensors. *The International Journal of Robotics Research, 17*, 924-933.

Laurent, G., Wehr, M., & Davidowitz, H. (1996). Temporal Representations of Odors in an Olfactory Network *Journal of Neuroscience, 16*(12), 3837-3847.

Mori, T., & Kai, S. (2002). Noise-Induced Entrainment and Stochastic Resonance in Human Brain Waves. *Physical Review Letters, 88*(21), 218101.

Moss, F., Ward, L., & Sannita, W. (2004). Stochastic resonance and sensory information processing: a tutorial and review of application. *Clin Neurophys, 27*, 677– 682.

Oros, N., Steuber, V., Davey, N., Cañamero, L., & Adams, R. G. (2008). *Optimal receptor response functions for the detection of pheromones by agents driven by spiking neural networks.* Paper presented at the 19th European Meetings on Cybernetics and Systems Research, Vienna.

Payton, D., Daily, M., Estowski, R., Howard, M., & Lee, C. (2001). Pheromone Robotics. *Auton. Robots, 11*(3), 319-324.

Pyk, P., Bermúdez i Badia, S., Bernardet, U., Knüsel, P., Carlsson, M., Gu, J., et al. (2006). An artificial moth: Chemical source localization using a robot based neuronal model of moth optomotor anemotactic search. *Autonomous Robots, 20*(3), 197-213.

Rudolph, M., & Destexhe, A. (2003). Characterization of subthreshold voltage fluctuations in neuronal membranes. *Neural Computation, 15*(11), 2577-2618.

Uhlenbeck, G. E., & Ornstein, L. S. (1930). On the theory of Brownian Motion. *Physical Review, 36*, 823-841.

Webb, B. (1998). Robots crickets and ants: models of neural control of chemotaxis and phonotaxis. *Neural Networks, 11*, 1479-1496.

Wiesenfeld, K., & Moss, F. (1995). Stochastic resonance and the benefits of noise: from ice ages to crayfish and SQUIDs. *Nature, 373*, 33–36.

Wyatt, T. D. (2003). *Pheromones and Animal Behaviour, Communication by Smell and Taste*: Cambridge University Press.

Holey Fitness Landscapes and the Maintenance of Evolutionary Diversity

Greg Paperin, Suzanne Sadedin, David Green, Alan Dorin

Faculty of Information Technology, Monash University
Monash University Clayton Campus, Building 63, Wellington Road
Clayton, 3800 Victoria, Australia
gpaperin@infotech.monash.edu.au

Abstract

Analytical models show that high-dimensional fitness landscapes form "holey" rather than "rugged" topographies, but the implications of this finding for biological and artificial life systems remain largely unexplored. One of the reasons for this gap can be attributed to serious difficulties in the implementation of individual-based holey fitness landscape (HFL) models. Here, we introduce a method for simulating HFLs in spatially explicit individual-based models that overcomes these difficulties. We examine how the HFL changes predictions for the maintenance of genetic diversity in the face of migration. Previous models suggest that ecologically-based reproductive isolation will rapidly collapse under migration. Our results indicate that an underlying HFL can often maintain diversity in this situation. Hybrid species emerge frequently when HFL genetics are simulated, but are usually doomed to extinction because of small population sizes. However, hybridisation can also lead to novel adaptations and potentially the exploitation of new ecological niches. More generally, the results imply that HFL genetics should not be neglected in studies of adaptation and diversity.

Introduction

The processes underlying the emergence and persistence of diversity form a key topic in evolutionary theory. Analytical models have provided considerable insight into these issues, but integrating the findings from different theoretical approaches remains a formidable challenge. In particular, the relationship between genetic diversity and reproductive isolation – widely considered the defining feature of biological species [3, 4] – remains controversial [5-9]. Here, we explore the dynamics of reproductive isolation (RI) in a genetically realistic fitness landscape within an individual-based, spatially explicit model.

Reproductive isolation (RI) is often seen as a requirement for biological diversification because it permits the coexistence of different lineages with co-adapted genomes. However, the origin and persistence of RI requires special circumstances. A mutant individual that is reproductively isolated from the surrounding population will rarely be successful. For this reason, speciation is usually thought to occur between spatially separated populations that acquire incompatible alleles through drift or selection [10-12]. However, even in this scenario, the maintenance of RI presents a theoretical challenge: even moderate migration between the two populations leads to selection against incompatible alleles, and the extinction or merging of incipient species is likely. Likewise, when RI is based on ecological divergence or mating barriers, it is often transient, collapsing when selection pressures change.

Recent theoretical advances suggest that assumptions about the relative fitness of different combinations of traits have profound implications for our understanding of these problems [11]. In particular, Gavrilets and Gravner [13] showed that when fitness landscapes have high dimensionality (as is likely for real organisms), the topology of the landscape changes from "rugged" to "holey". Several implications of this insight for speciation theory are explored by [11]. However, integrating the HFL into simulation models that incorporate spatially and ecologically plausible assumptions remains a challenge. In the sections that follow, we examine the notion of the fitness landscape and its implications for genetic diversity. We then present a method for integrating HFL genetics into a spatially explicit, individual-based model. Using this model, we explore conditions for maintenance of RI, genetic variation, and for the emergence of hybrid species.

Fitness Landscapes and Speciation

The term "fitness landscape" (FL) was coined by Wright [14] to represent the fitness of all conceivable individuals relative to their traits. He envisaged a rugged landscape, where peaks represented combinations of traits with high fitness separated by valleys of low-fitness trait combinations. On this landscape, selection drives populations uphill. Since Wright's work, several critiques of the FL concept have been made. Fitness landscapes are usually treated as static networks, but in reality, fitness is the ability to survive and reproduce in a dynamic environment that is constantly changing through co-evolutionary dynamics and external disturbances [2]. Some models account for this by using a FL that changes with time to reflect changes in the environment (e.g. [15]). However, the effects of genes underlying species differences and RI are, in general, not strongly affected by the environment and FLs are, therefore, widely accepted as a useful abstraction in theoretical biology [11].

In terms of FLs, the problem of speciation is that part of a population located at a fitness peak must cross the fitness valley surrounding the peak in order for the diverged genes not to be selected out. Stochastic factors such as genetic drift may act against natural selection and help overcoming fitness valleys, particularly for small populations, however, such factors can only account for selected types of speciation. It has been shown that speciation due to stochastic crossing of fitness valleys is, in general, extremely unlikely [10, 11].

Peaks in low-dimensional spaces become saddle points in higher-dimensional spaces. This led to the suggestion that highly multi-dimensional biological FLs may actually possess a single global maximum that can be reached by hill climbing from (almost) any point [16]. Although this model is useful in some cases, it does not apply in general: the local-maxima-to-saddle-point transformations are outnumbered by the appearance of new peaks in higher dimensions [11].

On a biochemical level, most genetic changes are fitness-neutral. This led to the suggestion that the fitness landscapes may be largely flat [17] and that the main force behind speciation is stochastic genetic divergence, i.e. genetic drift. However, an overwhelming proportion of biochemically conceivable genotypes are, in fact, inviable because they contain deleterious genes or groups of incompatible genes. Neutral fitness landscapes fail to account for this fact.

Holey Fitness Landscapes

A genetic model that accounts for the above limitations is the holey fitness landscape (HFL) introduced by Gavrilets [10, 11, 13]. Generally, a HFL is "an adaptive landscape where relatively infrequent high-fitness genotypes form a contiguous set that expands throughout the genotype space" [10].

To build some intuition for this model, we first recall a few results from the percolation theory which plays an important role in the analytical treatment of HFLs. Consider a 2-dimensional lattice of cells which can assume one of two states: "black" or "white" (figure 1). Let every cell be black with some probability p independently of all other cells, or white with probability $1 - p$. If p is small, the lattice will contain a few black cells, which may be grouped in a number of small, isolated clusters. As p increases, these clusters grow and merge. Once p crosses a certain threshold p_c, most of the black cells merge together into a single giant cluster that percolates the whole lattice (see figure 1). For a 2-dimensional square lattice this percolation threshold is known to be $p_c \approx 0.5927$ [18]. However, for lattices of higher dimensions the percolation threshold lies around the reciprocal of the lattice dimension [19], meaning that for a high dimension lattice a small proportion of black cells is sufficient for the emergence of a giant percolating cluster of connected black cells.

For the HFL model, we assume that a genotype is viable with probability p independent of all other genotypes, and inviable with probability $1 - p$. For the purpose of this discussion, the exact fitness of a genotype is irrelevant and we generalise to set the fitness of all viable and inviable genotypes to 1 and 0 respectively. Assume that all possible genotypes are ordered in an abstract genotype space in which the distance between the genotypes describes the probability or ease of transformation from one genotype to another. Distance 1 means that two genotypes can be transferred into each other through a single one-point mutation. Consider the space of all possible haploid genotypes with L loci and A alleles at each locus (note that for the purposes of this model, a diploid genotype with L loci can be represented as a haploid genotype with $2L$ loci [20]; for simplicity, we will therefore only consider haploid genotypes). The dimensionality of this genotype space is $D = L \times (A - 1)$, and the corresponding percolation threshold is $p_c = 1/D$. Note that even for short (on biological scales) genotypes a relatively small value of p will result in an extensive network of high-fitness ridges extending through the genotype space (e.g. for $L = 10^5$ and $A = 5$, $p_c \approx 20 \times 10^{-7}$). The traditional picture of rugged highly-dimensional FLs is therefore misleading, as these landscapes are characterised by the existence of percolating nearly neutral networks. It can be shown [11, chap. 4] that if the fitness of the genotypes is not restricted to 1 or 0, a large number of such networks emerges, each containing genotypes from a narrow fitness band. Among these networks, those with high fitness are particularly important as adaptive walks along such networks can proceed very far without any substantial loss to fitness.

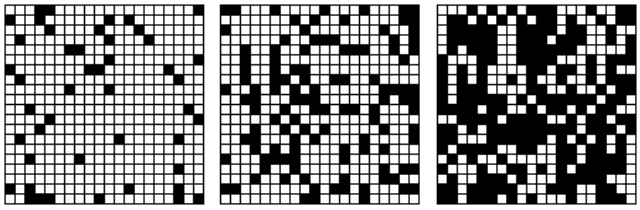


Figure 1. Percolation on a square lattice. The cells are black with probability $p = 0.1$ (left), $p = 0.3$ (middle) and $p = 0.6$ (right).

Holey Fitness Landscape in Simulations

There are a number of analytic models of adaptive radiation based on HFLs (e.g. see [11, part 1]), however they do not incorporate ecological selection and are not explicitly spatial. Other models treat disruptive (diversifying) selection while ignoring the viability and genomic compatibility issues introduced by the HFL (e.g. [21]) or make strong simplifying assumptions about such incompatibilities (e.g. [12]). It is known that diversification occurs easily in large spatial environments with disruptive ecological selection, and there will often be restricted gene flow between the resultant ecotypes, but how enduring RI occurs remains unclear. Gene flow barriers induced by mating barriers – even with strong ecological selection – appear to be transient. Models of adaptive radiation and ecological speciation in general deal with this simply by setting a threshold level of gene flow that they regard as acceptable, but this is unsatisfactory in that such species can merge back together as soon as selective pressures change. HFLs are thought to underlie the evolution of lasting, effective barriers to gene flow that appear during adaptive radiation, however this has not been further explored

in silico. The reason for this gap is that difficulties arise when realising a HFL in a computer model.

Recall that according to the HFL model, the majority of viable genotypes $G \in \mathbb{V}$ belong to a single largest connected cluster $\mathbb{V}' \subseteq \mathbb{V}$, where $\mathbb{V} \subset \mathbb{G}$ is the set of all viable genotypes and $\mathbb{G}$ is the set of all genotypes. The size of $\mathbb{V}'$ is of the order of $2^L \times p$, and $\mathbb{V}'$ percolates $\mathbb{G}$. The details of the proof can be found in [13]. The proof uses the idea of a surviving branching process to estimate the size of $\mathbb{V}'$. Assume that $p = p_c = 1/(L-1)$. The probability that the branching process dies at any specific branching point is given by $(1-p)^{L-1} = (1-1/(L-1))^{L-1} \approx (1-1/L)^L$. This means that the above statement holds with a probability 1 when $L \to \infty$. For finite but large L this probability is close to 1, however, for smaller L, the probability of the emergence of the giant connected cluster is smaller.

In natural populations, L is very large, but in an individual-based simulation, the genotype of each individual must be modelled explicitly, held in computer memory and processed by various operations. In practice, this limits the number of loci L to relatively low values. If L is small, $\mathbb{V}'$ can be expected to be small, i.e. $\mathbb{V}$ can be expected to consist of a large number of small clusters that are not connected to each other. Thus, an adaptive walk starting at some $G \in \mathbb{V}$ cannot proceed far in this case and evolution cannot occur.

Note, however, that for small L, the probability that $\mathbb{V}'$ contains most of $\mathbb{V}$ is not large, but positive. For any given small L and $p \geq p_c$, consider all possibilities for selecting $\mathbb{V}$ from $\mathbb{G}$. For most of such possibilities, $\mathbb{V}'$ is small, but there are some choices for which $\mathbb{V}'$ is large.

Any selection of $\mathbb{V}$ from $\mathbb{G}$, for which the giant connected cluster $\mathbb{V}'$ emerges, is an approximation of a HFL for large L. For any such selection, all crucial properties of $\mathbb{V}'$, $\mathbb{V}$ and $\mathbb{G}$ hold and no assumptions are violated. The results about HFL obtained in [11, 13] hold in these cases. If a way to select $\mathbb{V}$ from $\mathbb{G}$ such that the giant connected cluster $\mathbb{V}'$ emerges can be found, the resulting set of genotypes can be used as a basis for individual-based simulations exploring HFL genetics.

One of the challenges in creating an appropriate set $\mathbb{V}' \subset \mathbb{G}$ that is connected and uniformly distributed in $\mathbb{G}$ is related to the fact that the size of $\mathbb{V}'$ grows exponentially with L. We have developed a number of algorithms that allow creating $\mathbb{V}'$ for relatively large values of L (up to 30) within a few minutes on a common desktop computer. In [20] we give an overview of our approach and provide a numerical analysis of the evolutionary properties of the resulting FL.

In short, we create a set $\mathbb{V}'$ of haploid genotypes with a number of diallelic loci represented as bit-strings. This set adheres to the properties described above. The bit-strings are stored in a manner that allows an efficient implementation of a function *viable(G)* that takes an arbitrary bit-string and returns true iff $G \in \mathbb{V}'$.

In the remainder of the paper we introduce an individual-based simulation model designed to investigate to what extent can HFL-genetics sustain RI between spatially separated sub-population in face of migration.

Simulation model

Our objective is to investigate the extent to which HFL can sustain existing RI between spatially isolated populations under different levels of migration. For this we created an individual-based simulation model in which the individuals are located on a homogeneous landscape consisting of cells. Individuals, whose fitness (viability) is defined by the HFL, mate with other individuals within the same cell and then migrate to a neighbouring cell with a certain probability. As common in biological models (e.g. [21]), we use a number of neutral loci to measure the level of gene flow between the populations in different cells for different migration rates.

Methods

In this model the individuals are represented by their genotype, which consists of two sections: a *coding* section and a *neutral* section. The coding section consists of a number of diallelic loci that are assumed to code for vital traits. The coding section of a genotype is used as a parameter to the *viable* function of the HFL in order to determine whether an individual is viable. We experimented with 20 to 28 coding loci (not shown here) and found that the particular number does not affect the results significantly. In the experiments reported here we use L=26, which represents a trade-off between richer genotypes and computational resources required to complete a large number of simulation runs. The neutral genotype section consists of 5 loci with 128 different alleles possible at each locus. The neutral loci do not affect the fitness (viability) of an individual and are used to measure the genetic divergence between individuals (see figure 2).

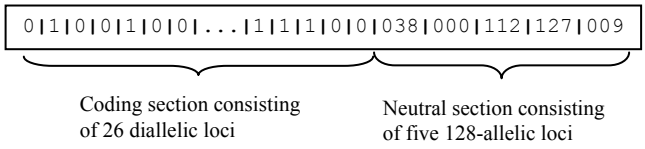


Figure 2. An example of a model genotype.

The lifecycle of a model individual is reproduction – selection – migration. Generations are non-overlapping.

Reproduction. Individuals mate only with other individuals within the same cell of the spatial landscape. Each individual in a cell is selected once as a mother. For each mother, a partner is uniformly randomly selected from the same cell (selfing is permitted). The number of offspring for each pair is drawn from a Poisson distribution with a parameter λ=4 (values in range λ=2..10 did not affect the results significantly). The genotype of each offspring is determined through free recombination of the parents' genotypes (i.e. the allele at each locus is inherited from each parent with equal probability independent of other loci). Each locus of the offspring is mutated with a probability 10^{-4} (values in the range 10^{-3} to 10^{-5} are commonly used in biological models of this kind, e.g. [2, 21]). If a coding locus is mutated, its binary value is flipped. The neutral loci are subject to a circular stepwise mutation model [22]. If the

coding section of an offspring's genotype is determined to be viable by the HFL model, the offspring is added to the new generation, otherwise it is discarded immediately. After all offspring for all pairs of parents have been determined, the old generation is discarded and replaced with the new population.

Selection. All individuals within a single cell of the spatial landscape compete to survive to the age of reproduction. Note that this approach is different from the approach commonly used in genetic algorithms, where all individuals survive and then compete to be selected for reproduction. Here all surviving individuals reproduce and their progeny compete to reach a mature age, which normally requires acquiring environmental resources. Each landscape cell is assumed to have a certain maximum carrying capacity C_{mc}, i.e. to provide enough resources for the survival of C_{mc} mature individuals. If a cell is inhabited by no more than C_{mc} individuals, all survive. Otherwise C_{mc} individuals are selected with equal probability and the rest are discarded (as in this HFL model a particular individual is either fit or inviable).

Dispersal. Individuals that reach maturity have a certain probability of migrating to one of the neighbouring spatial landscape cells. To avoid edge artefacts the landscape is represented as a torus. The effect of different migration rates is discussed in the results section.

In order to investigate how spatial distance affects the results we consider different grid layouts. We start with the simplest case (a 1×2 grid) and then gradually increase the grid size (2×2 cells and 3×3 cells). The results (discussed below) imply how the dynamics of RI will behave on larger landscapes. Each cell is initialised with a random viable individual with alleles at neutral loci all set to 0. Initially we disable any migration between the cells and iterate the model for 100 thousand generations in order to allow the allele distribution to reach equilibrium. We then turn on migration at a specific rate (see results section) and iterate the model for 300 thousand further generations. Measurements are taken every 1000 generations.

A quantity of prime interest in this model is the number of reproductively isolated groups (RI groups) present in the model at any one time as well as various attributes of such groups. We are interested in groups of genotypes that could mate successfully, not in groups of individuals who actually do so. Finding such groups is difficult as the groups may be partially overlapping (a genotype can successfully mate with two genotypes that cannot mate with each other) and the genetic distance between groups is initially unknown and may vary. In order to cluster the genotypes of a population into RI groups we employ the Markov Clustering algorithm (MCL) [23], as it does not require a distance threshold parameter and because it has been successfully applied to a similar task – clustering protein sequences into families [24] (we essentially cluster gene sequences into families). For that we first calculate a reproductive success probability matrix for all genotypes in the population. The probability of reproductive success of two genotypes is estimated by simulating a large number of crossovers between the genotypes and considering the proportion of crossovers that result in viable offspring. The matrix is then used as input to the clustering algorithm. To further verify the applicability of MCL to our model we apply this algorithm to a previous model of adaptive radiation that uses the same genetic setup [2]. There we investigated adaptive radiation under disruptive selection caused by ecological niches and RI groups could be determined simply by asserting to which niche genotypes were best adapted. Tests show that the RI groups determined by the clustering correspond to the groups determined by assigning the genotypes to niches (see figure 3).

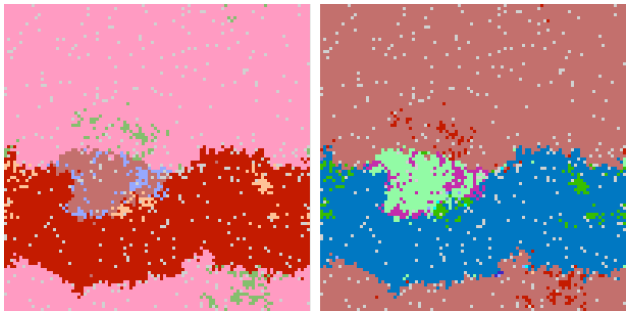


Figure 3. Using Markov Clustering (MCL) for determining RI groups. Depicted is a snapshot of a spatial landscape (100×100 grid) from [2]. Each cell is coloured according to the cluster to which the majority of the genotypes of the individuals inhibiting the cell belong. Left: the genotypes were assigned to RI groups using the MCL algorithm. Right: the genotypes were assigned to RI groups according to the ecological niche to which they are best adapted. Although represented by different colours, both groupings are largely the same.

On the basis of the RI groups we measure the average genetic divergence in neutral loci between the groups using the fixation index F_{st}. A number of slightly different approaches to calculating F_{st} have been proposed. Here we follow the approach taken in [25]: for every pair of genotypes within a group C, we measure the stepwise genetic distance – the minimum number of stepwise mutations necessary to obtain one genotype from the other – and calculate the average genetic distance $d_W(C)$ within the group C. We then measure the pair-wise distances between all genotypes that belong to C and all genotypes that do not belong to C in order to obtain the average genetic distance $d_B(C)$ between C and all other groups. Then, $F_{st}(C) = 1 - d_W(C) / d_B(C)$ and the overall fixation index F_{st} is the average of $F_{st}(C_i)$ for all groups C_i. Note that groups of different sizes are treated equally in this approach.

For each of the scenarios discussed below we have performed 10 independent model runs and averaged the results.

Simulation Results

Consider first the 2×2 layout. As a basis for comparison we performed a set of runs with a migration rate of 0%. As expected, the number of RI groups corresponds to the number of cells (4), the divergence at neutral loci grows (F_{st} approaches 1) and the number of distinct coding genotype sections in the population fluctuates around a value slightly higher than the number of RI groups – due to viable mutants

and drift (see figure 4). In one of the runs, two of the cells appeared not to be reproductively isolated as the random founder individuals were genetically similar by chance.

In the next scenario we increased the migration rate to 1% after the first 100,000 generations. This lead to a slight increase in the number of distinct coding sections in the population which is due to viable hybrids resulting from breeding with immigrants. Some of these hybrids spontaneously form RI groups, however such groups cannot persist due to low population numbers in comparison to native populations. These viable hybrids facilitate a limited gene flow between the populations: after 300,000 generations F_{st} has decreased to ca. 0.8 (figure 4.D).

The turnover of viable coding genotype sections in the population over time (the number of distinct viable coding sections that have been present in the model population from the start until a given time) can be used to describe the rate at which novel adaptive phenotypes are evolved. In the first 100,000 generations, when migration rate is 0%, the turnover increases at a small rate due to genetic drift. Once migration is enabled, the turnover grows at a higher rate which suggests that more new viable genotypes are discovered through hybridisation than through generic drift. While this result is sensitive to the mutation rate, it is even more pronounced for higher migration rates (figure 4.C).

In the next scenario the migration rate was set to 5% after the first 100,000 generations. Qualitatively, the results are similar to the 1% scenario. Quantitatively, the gene flow between the populations is higher (F_{st} falls to ca. 0.7, not shown). The higher migration rate leads to an increased probability for formation of RI hybrid groups (figure 4.A). Genetic drift within a larger number of RI groups as well as hybridisation between more diverse individuals leads to a larger number of coding genotype sections in the population (figure 4.B) and to a higher rate of discovering new viable adaptations (figure 4.C). Further rises in the migration rate to 10% (not shown), 15% and 20% (figure 4) increase the strength of the above effects.

When the migration rate is set to 25% or more the RI can be no longer sustained. A large number of reproduction events that lead to inviable offspring have a destabilising effect on the population size. Under such conditions, there is a high chance of extinction for any native cell population. Once an immigrant population has become established in a cell, a positive feedback loop is created: For individuals of the native population the chance of having viable offspring is decreased by the presence of the invaders, as they may be selected as mating partners. At the same time, the chance of new invaders to successfully reproduce is increased. As seen in figure 4.A the number of RI groups collapses to 1 under 25% migration. Sporadically small RI groups arise due to drift, but do not persist long enough to achieve a significant divergence in neutral loci (figure 4.D). The main population evolves as a single RI group. As a consequence, the number of distinct coding sections in the population is very small (figures 4.B & 4.C).

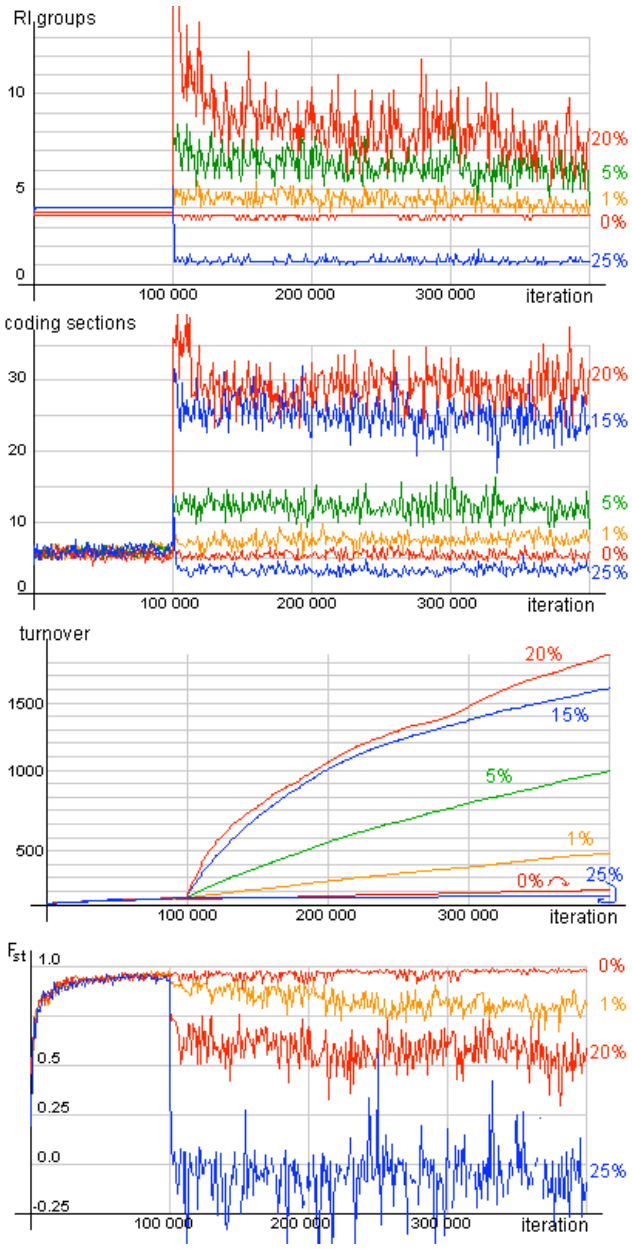


Figure 4. Evolution on a 2×2 grid for the migration rates 0% (red), 1% (orange), 5% (green), 15% (blue), 20% (red) and 25% (blue). Data averaged over 10 runs. Some values omitted for clarity.
A (top): The number of RI groups increases when the migration rate is higher. For very high migration rates the whole model population collapses into a single reproductive group.
B (2nd from top): The number of distinct coding genotype sections in the population increases when the migration rate is high. As the population collapses to a single reproductive group at very high migration rates, the number of coding sequences falls.
C (3rd from top): The rate of evolving new viable coding genotype sections increases when migration rate is higher due to drift in a larger number of IR groups and due to hybridisation between more RI groups. As the population collapses into a single reproductive group at very high migration rates, the number of coding sequences falls.
D (bottom): Genetic divergence between RI groups measured using the fixation index. Higher migration rates lead to increased gene flow and this lower genetic divergence.

In order to investigate how spatial distance affects the above results we have repeated the experiments on a 1×2 grid. In large the model behaviour is similar, however the migration rate has a larger impact on the smaller landscape.

Readily a migration rate of 1% causes F_{st} to decrease to ca. 0.5 after 300,000 generations of migration (figure 5.C). A migration rate of 10% causes the generic divergence of the two RI groups to decrease to insignificant levels within 50,000 generations of migration. However, RI can be sustained at 10% and 15% migration – the number of RI groups stays around 2 which shows that the significant gene flow is not sufficient to break RI and must occur through viable hybrids, who, however, cannot establish a separate RI population. This can also be seen in that the number of distinct coding sections in the population remains small (not shown) suggesting that hybrids occur between the same genotypes. This conclusion is further supported by the turnover rate of the coding sections (figure 5.B): After an initial increase similar to the 2×2 scenarios, the turnover rate slows down to a level close to the rate before migration was turned on, showing that the two populations have reached an equilibrium and that further genetic innovation is due to drift. At 20% migration RI collapses rapidly and the entire model population evolves as a single reproductive group (figure 5).

Next, we repeated the experiments on a 3×3 grid. As expected, larger grid makes it possible to sustain RI at higher migration rates. At 30% migration RI is sustained and the number of RI groups lies above 40. At 35% migration, RI collapses in a way similar to the previous scenarios (not shown).

In order to give our HFL simulations a basis for comparison, we simulated all of the above scenarios without the HFL. In these control runs all individuals are viable and selection is thus random. In this context, RI cannot be defined and the number of RI groups and F_{st} cannot be measured. However, a related measure is the average genetic divergence D_B at neutral loci between all individuals of the entire model population. In the presence of groups without gene flow between them, D_B is expected to grow as the neutral loci in such groups will diverge. We measured D_B for all grid sizes discussed earlier. As expected, for a migration rate of 0%, D_B steadily grows. However, for all grid sizes, a migration rate of 1% is sufficient to cause D_B to drop sharply and to remain low for the rest of the simulation (figure 6). This indicates that without HFL-genetics (or other reproductive barriers occurring in nature) RI cannot be sustained even for small migration rates.

Conclusions

The role of spatial separation in facilitating RI is well known [14]. The difference between the three spatial scenarios demonstrates this effect. In order for an allele to pass from one cell to another non-adjacent cell it must first become established in the intermediate locations. Strong RI induced by the HFL enhances this effect. Thus, hybrid zones and divergent satellite populations may provide a stronger barrier to gene flow than often assumed.

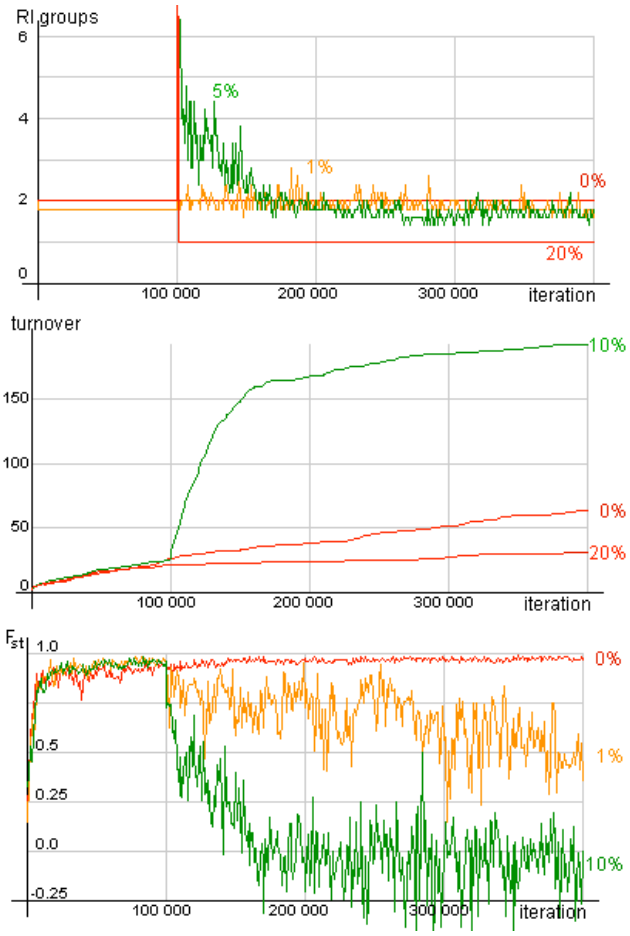


Figure 5. Evolution on a 1×2 grid for the migration rates 0% (red), 1% (orange), 5% (green), 10% (green) and 20% (red).
Data averaged over 10 runs. Some values omitted for clarity.
A (top): Number of RI groups.
B (middle): Turnover of viable coding genotype sections.
C (bottom): Genetic divergence measured using the F_{st}.
(The graphs in this paper were created and processed using the LiveGraph exploratory data analysis and visualisation framework [1].)

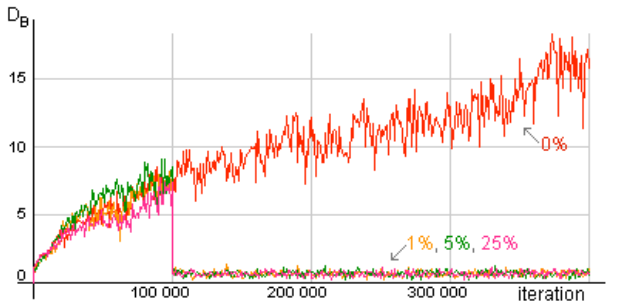


Figure 6. Average genetic divergence D_B at neutral loci between the individuals of the entire population in neutral evolution (without HFL). The average genetic distance grows when migration rate is 0%. The average distance quickly collapses to a small value above 0 (due to drift) for all other migration rates (1%, 5%, 25% are shown). This behaviour is largely the same for all grid sizes considered.

The effect of higher mutation rates on the number of distinct viable coding sections in the population is stronger than on the number of RI groups. This suggests that despite HFL, a small proportion of hybrids is viable and does not exhibit RI from the main population. It is these viable hybrids that facilitate the gene flow between RI populations. However, the small effect of an increasing migration rate on the number of RI groups implies that some hybrid populations exhibit real RI and are not simply fuelled by repeated hybridisation with immigrants.

The common assumption is that hybrid zones are maintained by an interaction between continuous hybridisation and selection against hybrids. RI between the hybrids and the main population is often attributed to ecological preferences to a specific environment within the hybrid zone and not to genetic incompatibility. If such a specific ecological environment is altered, the hybrids become disadvantaged. As a result they become extinct either through selection against them or by adapting to the main environment thus removing RI between the hybrids and the main population. However, hybrid populations that have strong genetic incompatibilities with the main population caused by HFL-genetics are more likely to persist. In our simulations such populations are short-lived because their small initial population size and the absence of prezygotic isolation (RI caused by not mating with members of other groups rather than by offspring inviability) make it unlikely that they successfully reproduce for a large number of consecutive generations. However, in the presence of a free ecological environment niche within the hybrid zone, hybrid groups can multiply in numbers and persist. These populations, once numerous, are less likely to be affected by a disturbance of their specific ecological niche due to the strong genetic RI between the hybrids and the main population. This can allow the hybrid population to further diverge eventually forming prezygotic RI and thus to speciate. Although further data are required, this observation provides potential support for the analogy of novel species to point mutations implicit in some recent ecological [26] and macro-evolutionary [27] theory.

As discussed earlier, for relatively high migration rates, an immigrant (not hybrid) population that became established in a new environment is likely to induce a positive feedback loop leading to the extinction of the native population: A large number of immigrants who act as potential mating partners in the absence of prezygotic RI decreases the chance of native inhabitants to have viable offspring and increases the chance of further invaders to successfully reproduce. This may lead to reinforcement – the evolution of sexual selection and thus prezygotic mating barriers in response to selection against hybrids. Reinforcement is a controversial topic in speciation theory [28, 29]. However, as argued in the previous paragraph and supported by our results, RI generated by the HFL is often resistant to mutations reducing hybrid disadvantage. Thus, reinforcement may be more likely in the context of HFL-genetics than previous models indicate [28, 29].

The notion of holey fitness landscapes, while largely unchallenged, has arguably received insufficient attention from theorists. The current model shows that simulating plausible fitness landscapes can considerably change predictions about the maintenance of diversity and the emergence of new adaptations and species. The approach described here may be useful in further exploring these issues and related problems of adaptive radiation, evolvability and evolutionary search. From the perspective of artificial life research, representing fitness landscapes in a biologically plausible way may facilitate ongoing adaptive exploration and the continuous generation of novelty in evolving populations.

References

[1] (2007). *LiveGraph - a framework for real-time data visualisation, analysis and logging*. Retrieved on 01.03.2008 from: http://www.live-graph.org.

[2] G. Paperin, D. G. Green, S. Sadedin and T. G. Leishman (2007). A Dual Phase Evolution model of adaptive radiation in landscapes. In M. Randall, H. A. Abbass and J. Wiles (eds.), *The Third Australian Conference on Artificial Life (ACAL'07)*, pp. 131-143 Springer.

[3] T. G. Dobzhansky (1937). *Genetics and the origin of species*. Columbia University Press, New York.

[4] E. Mayr (1942). *Systematics and the origin of species*. Columbia University Press, New York.

[5] K. De Queiroz (1998). *The general lineage concept of species, species criteria, and the process of speciation: a conceptual unification and terminological recommendations*. In *Endless forms: Species and speciation*. D. J. Howard and S. H. Berlocher (eds.), pp. 57-75. Oxford University Press, New York.

[6] R. G. Harrisson (1998). *Linking evolutionary pattern and process: the relevance of species concepts for the study of evolution*. In *Endless forms: Species and speciatiopn*. D. J. Howard and S. H. Berlocher (eds.), pp. 19-31. Oxford University Press, New York.

[7] J. Mallet (1995). A species definition for the modern synthesis. *Trends in Ecology and Evolution*. 10 (7): pp. 294-299.

[8] K. L. Shaw (1998). *Species and the diversity of natural groups*. In *Endless forms: Species and speciation*. D. J. Howard and S. H. Berlocher (eds.), pp. 44-56. Oxford University Press, New York.

[9] A. R. Templeton (1998). *Species and speciation: geography, population structure, ecology and gene trees*. In *Endless forms: Species and speciation*. D. J. Howard and S. H. Berlocher (eds.), pp. 32-43. Oxford University Press, New York.

[10] S. Gavrilets (2003). Models of Speciation: What have we learned in 40 years? *Evolution*. 57 (10): pp. 2197-2215.

[11] S. Gavrilets (2004). *Fitness Landscapes and the Origin of Species*. In *Monographs in Population Biology*. Princeton University Press, Princeton / Oxford.

[12] H. A. Orr (1995). The Population Genetics of Speciation: The Evolution of Hybrid Incompatibilities. *Genetics*. 139 (4): pp. 1805-1813.

[13] S. Gavrilets and J. Gravner (1997). Percolation on the Fitness Hypercube and the Evolution of Reproductive Isolation. *Journal of Theoretical Biology*. 184 (1): pp. 51-64.

[14] S. Wright (1932). The roles of mutation, inbreeding, crossbreeding and selection in evolution. In D. F. Jones (ed.), *6th International Congress of Genetics*, pp. 256-366.

[15] S. A. Kauffman (1993). *The Origins of Order: Self-Organization and Selection in Evolution*. Oxford University Press, USA.

[16] W. B. Provine (1986). *Sewall Wright and Evolutionary Biology* University of Chicago Press.

[17] M. Kimura (1983). *The neutral theory of molecular evolution*. Cambridge University Press, Cambridge, New York.

[18] M. E. J. Newman and R. M. Ziff (2000). Efficient Monte Carlo Algorithm and High-Precision Results for Percolation. *Physical Review Letters*. 85 (19): pp. 4104-4107.

[19] G. R. Grimmett (1999). *Percolation*. Springer.

[20] G. Paperin, D. G. Green and A. Dorin (2007). Fitness Landscapes in Individual-Based Simulation Models of Adaptive Radiation. In T. D. Pham and X. Zhou (eds.), *2007 International Symposium on Computational Models for Life Science (CMLS'07)*, pp. 268-278 American Institute of Physics.

[21] S. Gavrilets and A. Vose (2005). Dynamic patterns of adaptive radiation. *Proceedings of the National Academy of Sciences USA*. 102 (50): pp. 18040-18045.

[22] T. Ohta and M. Kimura (1973). A model of mutation appropriate to estimate the number of electrophoretically detectable alleles in a finite population. *Genetical Research*. 22 (2): pp. 201-204.

[23] S. Van Dongen (2000): "Graph Clustering by Flow Simulation". University of Utrecht: Utrecht.

[24] A. J. Enright, S. Van Dongen and C. A. Ouzounis (2002). An efficient algorithm for large-scale detection of protein families. *Nucleic Acids Research*. 30 (7): pp. 1575-1584.

[25] R. R. Hudson, M. Slatkin and W. P. Maddison (1992). Estimation of Levels of Gene Flow From DNA Sequence Data. *Genetics*. 132 (2): pp. 583-589.

[26] S. P. Hubbell (2001). *The Unified Neutral Theory of Biodiversity and Biogeography*. Princeton University Press.

[27] S. J. Gould (2002). *The Structure of Evolutionary Theory*. Belknap Press, Harvard.

[28] H. G. Spencer, B. H. Mcardle and D. M. Lambert (1986). A Theoretical Investigation of Speciation by Reinforcement. *The American naturalist*. 128 (2): pp. 241-262.

[29] R. Butlin (1987). Speciation by reinforcement. *Trends in Ecology & Evolution*. 2 (1): pp. 8-13.

Conformist Transmission and the Evolution of Cooperation

Jorge Peña

Institute of Applied Mathematics, University of Lausanne,
UNIL-Dorigny, CH-1015 Lausanne, Switzerland
jorge.pena@unil.ch

Abstract

We study the effects of conformist transmission on the evolutionary dynamics of the Prisoner's Dilemma, the Snowdrift and the Stag Hunt games in both well-mixed and spatially structured populations. The addition of conformism introduces a transformation of the payoff matrix that favours the stability of pure equilibria and reduces the basin of attraction of risk dominant equilibria. When both conformism and local interactions are present, the system can exhibit higher levels of cooperation than those obtained in the absence of either of the two mechanisms.

Introduction and Related Work

Evolutionary game theory (Hofbauer and Sigmund, 1998; Gintis, 2000) is the theory of evolutionary dynamics when selection is frequency-dependent, i.e. when the success of an individual is conditioned not only by the strategy he or she follows but also by the strategies followed by other individuals in the population. Although originally developed as an application of game theory to the study of genetic evolution (Maynard Smith, 1982), evolutionary game theory has also been used to investigate cultural evolutionary processes, that is the way ideas or beliefs spread through a population of individuals capable of imitation.

In *cultural* evolutionary game-theoretic models, ideas are transmitted via biased imitation. Most of these models posit that the only important psychological bias underlying imitation is *prestige* or *payoff-based bias*, defined as the predisposition to imitate successful individuals. Assuming a very large and well-mixed population, payoff-based biased transmission can be shown to generate a famous differential equation, named the *replicator dynamics* (Taylor and Jonker, 1978; Gintis, 2000). In the context of evolutionary game theory, the equilibrium points and other characteristics of the dynamics of different games are studied in order to better understand the evolutionary processes involved.

The Prisoner's Dilemma (PD), Snowdrift[1] (SD) and the Stag Hunt (SH) are among the most studied two-person, symmetric games in the literature. They are used for investigating under which circumstances altruistic traits can become fixed in a population of "selfish" individuals. In social dilemmas of cooperation, individuals' behaviours are of two types: cooperative and non-cooperative. Cooperators are willing to engage in cooperative tasks, while non-cooperators (usually called *defectors*) prefer not to. The success resulting from the interaction of cooperators and defectors is given by the payoff matrix:

	C	D
C	R	S
D	T	P

where C denotes cooperators and D denotes defectors. R is the *reward* for mutual cooperation, P is the *punishment* for mutual defection, T is the *temptation* to defect and S is the *sucker's payoff*.

In all three social dilemmas, mutual cooperation is favoured over both mutual defection ($R > P$) and an equal probability of unilateral cooperation and defection ($2R > T + S$). The three dilemmas however differ in their ordering of payoffs. In the PD, $T > R > P > S$; in SD, $T > R > S > P$, and in the SH, $R > T > P > S$.

The evolution of cooperation can be studied by looking at the stable equilibria of the replicator dynamics for each of these games. In the PD, the only stable equilibrium occurs when the population is entirely comprised of defectors. In the SD game cooperators and defectors coexist in equilibrium. In the SH there are two equilibria: when all individuals cooperate and when all individuals defect. This last equilibrium is however risk dominant, i.e. it has the largest basin of attraction.

The replicator dynamics is a rough approximation of actual cultural evolutionary dynamics as it assumes that populations are very large and well-mixed, and that payoff-based bias is the sole psychological mechanism guiding cultural transmission processes. More realistic models of cultural evolutionary processes correct at least one of these assumptions and arrive at different results from those predicted by the standard replicator dynamics.

[1] Also known as Hawks-Doves or Chicken.

Evolutionary graph-theoretical models (Lieberman et al., 2005; Szabo and Fath, 2007), for instance, go beyond the assumption of large, well-mixed populations by restricting interaction and imitation to near neighbours in a graph representing spatial locality or a social network. In many cases, this graph structure has been shown to promote cooperation beyond the limits of the replicator dynamics in a well-mixed population (Nowak and May, 1992; Nowak et al., 1994; Skyrms, 2003; Santos and Pacheco, 2005).

Other researchers have augmented cultural evolution models by including different psychological biases that, together with payoff-based bias, could influence the way people imitate. In particular, *conformism* or *conformist bias* (Boyd and Richerson, 1985), which is the propensity for preferentially imitating common behaviours, has been suggested to be an important component of our social learning psychology (Asch, 1951; Coultas, 2004).[2] When conformist transmission is introduced in cultural evolution models, the result (in the case of large, well-mixed populations) is a *modified* replicator dynamics that can lead to different equilibrium points and different dynamics from those predicted by the standard replicator dynamics (Henrich, 2001; Skyrms, 2005). By making use of such equation, Henrich and Boyd (2001) have shown how even limited levels of conformism are able to stabilise cooperative behaviour in a public goods game if punishment is also included in the model. In related work, Skyrms (2005) has explored the effect of conformist bias in a number of symmetric two-by-two games. Analyses in that work were however restricted to some specific numerical cases and no general conclusions were formally drawn.

The aim of this paper is to study the effects of conformist transmission on the evolution of cooperation when considering two-person symmetric games such as the PD, SD and the SH. We propose an evolutionary graph-theoretical model in which cultural transmission is guided by both payoff-based and conformist biases, and study it both analytically and by means of simulation.

The paper is organised as follows. The next section gives the agent-based level specifications of the model. It is then shown how to recover the modified replicator dynamics in the limiting case of a large and well-mixed population, and the equation is studied by means of equilibrium analysis. This is followed by a simulation study of the particular case of a population organised into a regular 2D lattice. Finally, conclusions are drawn.

[2]From an evolutionary psychology perspective, conformist bias could have evolved because it is adaptive in the face of costly information. Boyd and Richerson (1985) and Henrich and Boyd (1998) have theoretically shown that conformist transmission is adaptive in spatially and/or temporally varying habitats since it provides a simple heuristic rule that increases the probability of acquiring locally adaptive beliefs and behaviours.

The Model

Our model considers a population of n individuals, where the i-th individual is represented by the vertex v_i of an undirected graph $G(V, E)$ with $v_i \in V\ \forall i$. The open neighbourhood of i, $N(i)$, is the set of all individuals j such that there is an edge $e_{ij} \in E$. The number of neighbours of individual i is thus the degree k_i of vertex v_i. The closed neighbourhood $N[i]$ is the set of i's neighbours plus i itself.

Each individual is characterised by its *cultural trait* or *strategy* $s_i \in \{A, B\}$. Social interaction is modelled by means of a two-person, symmetric game with a payoff matrix M given by[3]:

	A	B
A	a	b
B	c	d

Each time step t, individuals simultaneously engage in social interactions. As a result of these interactions, individual i collects an *average payoff* given by:

$$u_i(t) = \frac{1}{k_i} \sum_{j \in N(i)} M\left(s_i(t), s_j(t)\right).$$

After interactions are completed, individual i randomly chooses one of its neighbours $j \in N(i)$ as its *model* for cultural transmission. Imitation is assumed to be conformist-biased with probability α and payoff-biased with probability $1 - \alpha$. Parameter α thus weighs the importance of conformism relative to payoff-biased transmission.

The adoption of individual j's strategy by the focal individual i depends on j's *cultural fitness* w_{ij}. Cultural fitness (the direct analogue to biological fitness in genetic evolution) is a measure of the attractiveness or the transmissibility of a model's strategy. If transmission is payoff-biased, j's cultural fitness is given by the difference of average payoffs between j and i:

$$w_{ij}(t) = u_j(t) - u_i(t).$$

If transmission is conformist, j's cultural fitness is given by

$$w_{ij}(t) = q_{ij}(t) - \frac{1}{2},$$

where q_{ij} is the proportion of agents in $N[i]$ having the same strategy as j. Notice that w_{ij} is positive whenever $u_j > u_i$ (payoff-biased transmission) or j follows the strategy followed by the majority of i's neighbours (conformist transmission).

Agent i copies j's strategy with a probability proportional to w_{ij}. Formally:

$$Pr\left(s_i(t+1) = s_j(t)\right) = f\left(w_{ij}\right),$$

[3]Without loss of generality, payoffs are assumed to be non-negative values.

where f is assumed to be a monotonically increasing function, in order for models with high cultural fitness to propagate their strategies more often than models with low cultural fitness. Three alternative definitions of f are considered in this paper, each one specifying a different imitation rule: (i) imitate-if-better (IIB); (ii) replicator dynamics 1 (RD1); and (iii) replicator dynamics 2 (RD2).[4]

The IIB rule is given by:

$$f_{IIB}(w_{ij}) = \begin{cases} 0 & \text{if } w_{ij} \leq 0 \\ 1 & \text{if } w_{ij} > 0 \end{cases},$$

whereas RD1 and RD2 are respectively defined by:

$$f_{RD1}(w_{ij}) = \begin{cases} 0 & \text{if } w_{ij} \leq 0 \\ \beta w_{ij} & \text{if } w_{ij} > 0 \end{cases},$$

and

$$f_{RD2}(w_{ij}) = \frac{1}{2}\left(1 + \beta w_{ij}\right).$$

Parameter β normalises w_{ij} such that $0 \leq Pr\left(s_i(t+1) = s_j(t)\right) \leq 1$. Thus, $\beta = 2$ in the case of conformist transmission and

$$\beta = \frac{1}{\max\{a,b,c,d\} - \min\{a,b,c,d\}} \quad (1)$$

in the case of payoff-biased transmission. Fig. 1 depicts f for each imitation rule.

The three imitation rules described above have been traditionally used in the literature, either directly in evolutionary graph-theoretical models (e.g. RD1 by Hauert and Doebeli (2004) and Santos and Pacheco (2005)) or in order to derive population-level analytical models (e.g. RD2 by Henrich (2001) and Boyd and Richerson (2002)).

From the previous definitions it is possible to derive $Pr\left(s_i(t+1) = A\right)$, which is the probability of individual i following strategy A at time step $t+1$ after having chosen a neighbour j as a model. Individual i's strategy will become or remain A whenever: *a*) A is the current strategy of both i and j ; *b*) i's current strategy is A, j's current strategy is B, but i does not imitate j; or *c*) i's current strategy is B, j's current strategy is A, and i imitates j. The formal equation is shown in Fig. 2.

Exact analysis for the case of large, well-mixed populations

General games

Here we analyse the limiting case of a complete graph with large n, which is equivalent to having the large, well-mixed population that is traditionally assumed in standard evolutionary game theory.

[4]We give RD1 and RD2 these names because both imitation rules can be shown to recover the replicator dynamics in the well-mixed, 100% payoff-biased transmission case (Gintis, 2000; McElreath and Boyd, 2007).

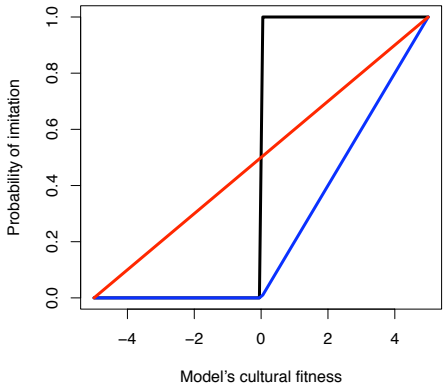


Figure 1: Imitation rules. IIB is shown in *black*, RD1 ($\beta = 0.2$) in *blue* and RD2 ($\beta = 0.2$) in *red*.

Let p_t denote the frequency of individuals with strategy A at time step t. For a complete graph with $n \to \infty$, $k_i = n - 1 \approx n\ \forall i$, and

$$u_i(t) = \begin{cases} u_A(t) & \text{if } s_i(t) = A \\ u_B(t) & \text{if } s_i(t) = B \end{cases}$$

$\forall i$, where $u_A(t)$ and $u_B(t)$ are the average payoffs collected by individuals with strategies A and B at time step t, respectively given by

$$u_A(t) = ap_t + b(1 - p_t), \quad (2)$$

and

$$u_B(t) = cp_t + d(1 - p_t). \quad (3)$$

Additionally, since $N[i] = V\ \forall i$:

$$q_{i,j}(t) = \begin{cases} p_t & \text{if } s_j(t) = A \\ 1 - p_t & \text{if } s_j(t) = B \end{cases} \quad \forall i, j.$$

Using these relations and RD2 as imitation rule, the equation of Fig. 2 can be shown to reduce to:

$$\begin{aligned} \Delta p \quad = \quad & p_t(1 - p_t)\{(1 - \alpha)\beta\left[u_A(t) - u_B(t)\right] \\ & + \alpha(2p_t - 1)\}, \end{aligned} \quad (4)$$

where $\Delta p = p_{t+1} - p_t$ is the change in the proportion of individuals with behaviour A between time steps t and $t+1$. The recursion of Eq. 4 is a modified replicator dynamics that had been already derived in related work on cultural transmission processes including both payoff-biased and conformist imitation (Henrich and Boyd, 2001; Henrich, 2001; Carpenter, 2004; Skyrms, 2005).

Let us first analyse the particular case when cultural transmission is payoff-biased only. Making $\alpha = 0$, Eq. 4 reduces to:

$$\Delta p = p_t(1 - p_t)\beta\left\{u_A(t) - u_B(t)\right\},$$

$$
\begin{aligned}
Pr\left(s_i(t+1)=A\right) &= Pr\left(s_i(t)=A, s_j(t)=A\right)(1) \\
&+Pr\left(s_i(t)=A, s_j(t)=B\right)\left\{(1-\alpha)\left[1-f\left(u_j(t)-u_i(t)\right)\right]+\alpha\left[1-f\left(q_{ij}(t)-\tfrac{1}{2}\right)\right]\right\} \\
&+Pr\left(s_i(t)=B, s_j(t)=A\right)\left\{(1-\alpha)\left[f\left(u_j(t)-u_i(t)\right)\right]+\alpha\left[f\left(q_{ij}(t)-\tfrac{1}{2}\right)\right]\right\}
\end{aligned}
$$

Figure 2: Probability of individual i having strategy A at time step $t+1$ after cultural transmission from model j

which is the discrete-time equivalent of the standard replicator dynamics (Taylor and Jonker, 1978; Hofbauer and Sigmund, 1998; Gintis, 2000). Substituting Eq. 2 and 3 in the last expression and doing little algebra:

$$\Delta p = p_t(1-p_t)\beta\left\{(a-b-c+d)p_t+b-d\right\}. \quad (5)$$

Equilibria of this equation can be found by looking at the values of p_t that make $\Delta p = 0$. The two *pure equilibria* are given by $p_t = 0$ and $p_t = 1$. In the following, these equilibria will be respectively called all-B and all-A. A third *internal equilibrium*, in which players with strategies A and B are present in the population, may exist. When this is the case, the proportion of individuals with strategy A in equilibrium is given by

$$p^* = \frac{d-b}{(a-c)+(d-b)}.$$

In general, the equilibrium p is stable[5] whenever

$$\left|\left.\frac{dp_{t+1}}{dp_t}\right|_{p_t=p}\right| < 1.$$

From this, it can be easily shown that

- all-B is stable when $b < d$,
- all-A is stable when $a > c$, and
- p^* is stable when both $a < c$ and $b > d$.

Depending on the ranking of the entries of the payoff matrix, four different possibilities[6] for the imitation dynamics can thus be distinguished (Nowak, 2006):

1. $a > c \wedge b > d$: only all-A is stable (A *dominates* B).
2. $a < c \wedge b < d$: only all-B is stable (B *dominates* A).
3. $a > c \wedge b < d$: both all-A and all-B are stable (A and B are *bistable*). In this case, the internal unstable equilibrium p^* determines the sizes of the basins of attraction of the two pure equilibria. The equilibrium with the largest basin of attraction is called *risk dominant*. In particular
   a) all-A is risk dominant if $d-b < a-c$, and
   b) all-B is risk dominant if $d-b > a-c$.
4. $a < c \wedge b > d$: pure equilibria are unstable and the internal equilibrium is stable (A and B *coexist*).

How this picture changes when cultural transmission has also a conformist component ($\alpha > 0$)? In order to answer to this question, an equilibrium analysis similar to the one done in the case $\alpha = 0$ can be performed here for $\alpha \neq 0$. A second possibility is to rewrite Eq. 4 as

$$
\begin{aligned}
\Delta p &= p_t(1-p_t)\{[(1-\alpha)\beta(a-b-c+d)+2\alpha]\,p_t \\
&\quad +(1-\alpha)\beta(b-d)-\alpha\},
\end{aligned}
$$

and perform the following variable substitutions

$$
\begin{aligned}
a' &= (1-\alpha)\beta a+\alpha, \\
b' &= (1-\alpha)\beta b, \\
c' &= (1-\alpha)\beta c, \\
d' &= (1-\alpha)\beta d+\alpha,
\end{aligned}
$$

to obtain:

$$\Delta p = p_t(1-p_t)\left\{(a'-b'-c'+d')p_t+b'-d'\right\}. \quad (6)$$

Notice (see Eq. 5) that this recursion is equivalent to the discrete replicator dynamics of a population game with the following payoff matrix M':

	A	B
A	a'	b'
B	c'	d'

Hence, in the framework of the replicator dynamics, *the addition of conformism to the cultural evolutionary process is equivalent to a transformation of the payoff matrix of the underlying game*. Observe that $\alpha = 0$ recovers the original game and $\alpha = 1$ completely transforms the original game into a pure coordination game with the following payoff matrix:

	A	B
A	1	0
B	0	1

The addition of conformism to imitation dynamics can have considerable effects in the nature of equilibria of the modelled cultural evolutionary process (Boyd and Richerson, 1985; Henrich and Boyd, 2001; Henrich, 2001; Skyrms,

[5] The condition is necessary and sufficient for hyperbolic equilibria only. All-B (resp. all-A) is non-hyperbolic when $b = d$ (resp. $a = c$).

[6] Actually, there is a fifth possibiity: A and B are neutral when $a = c$ and $b = d$. In this case there is no evolution since $\Delta p = 0$ $\forall p_t$.

2005). In particular, since the entries of M are non-negative and $0 \leq \alpha \leq 1$,

$$a < c \not\Rightarrow a' < c'$$
$$b > d \not\Rightarrow b' > d',$$

which means that *a*) originally unstable pure equilibria could become stable and *b*) an originally stable internal equilibrium could become unstable. Furthermore, if A and B coexist, the proportion of individuals with strategy A in equilibrium is now given by

$$p'^{*} = \frac{(1-\alpha)\beta(d-b)+\alpha}{(1-\alpha)\beta\{(a-c)+(d-b)\}+2\alpha}.$$

Not everything changes in the dynamics of the game when conformism is introduced. In particular,

$$a > c \Rightarrow a' > c',$$
$$b < d \Rightarrow b' < d',$$

which means that originally stable pure equilibria will continue to be stable in the transformed game. Moreover,

$$d-b < a-c \Rightarrow d'-b' < a'-c',$$
$$d-b > a-c \Rightarrow d'-b' > a'-c',$$

which means that, if A and B are bistable, the risk dominant equilibrium of the transformed game will be the same as the one of the original game.

The new conditions for stability are

1. All-B is stable if

$$\alpha > \frac{\beta(b-d)}{1+\beta(b-d)} \tag{7}$$

2. All-A is stable if

$$\alpha > \frac{\beta(c-a)}{1+\beta(c-a)} \tag{8}$$

3. The internal equilibrium, when it exists, is stable if neither Eq. 7 nor Eq. 8 holds.

Social dilemmas

Let us now focus on the effect of conformist biases in games reflecting social dilemmas, such as the PD, SD and the SH. In order to simplify the analysis for these games, it is customary to rescale their payoff matrices so that they depend on a single parameter. For the PD, we follow Nowak and May (1992) and make $T = b$, $R = 1$, $P = \epsilon \approx 0$ and $S = 0$, where $1 < b < 2$ characterises the advantage of defectors against cooperators. For the SD game, we follow Hauert and Doebeli (2004) and make $T = \gamma > 1$, $R = \gamma - 1/2$, $S = \gamma - 1$ and $P = 0$, such that the cost-to-benefit ratio of mutual cooperation is given by $r = 1/(2\gamma - 1)$, with $0 \leq r \leq 1$. For the SH we make $T = P = 1$, $R = g$ and $S = 0$, with $1 < g < 2$. With these settings, $\beta = 1/b$ for the PD, $\beta = 1/\gamma$ for SD and $\beta = 1/g$ for the SH in the case of payoff-based biased imitation (see Eq. 1).

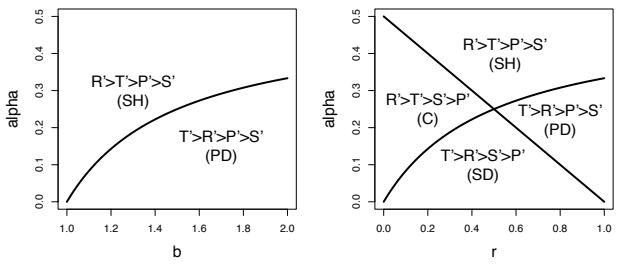


Figure 3: Effect of conformist bias in the PD (*left*) and the SD game (*right*).

As it has been previously analysed, the effect of conformist transmission may be interpreted as a transformation in the payoff matrix that can alter the original ordering of its entries. This in turn can drastically change the nature of the game played. In the PD with conformism, the all-C equilibrium (unstable in the original game) can become stable if $R' > T'$. This holds when

$$\alpha > \frac{b-1}{2b-1}.$$

The resulting ordering of the payoffs ($R' > T' > P' > S'$), and the fact that all-D is always the risk-dominant equilibrium, effectively converts the game into a SH (see Fig. 3).

In the case of the SD game, the ordering of the entries of the transformed payoff matrix M' can be different from that of the original matrix M if $R' > T'$ (all-C becomes stable), $P' > S'$ (all-D becomes stable) or both conditions hold. For the rescaled version of this game, $R' > T'$ whenever

$$\alpha > \frac{r}{1+2r},$$

and $P' > S'$ when

$$\alpha > \frac{1-r}{2}.$$

There are thus 4 different possibilities for the SD game with conformist transmission (see Fig. 3):

1. $T' > R' > S' > P'$ (the game is still a SD),
2. $R' > T' > S' > P'$ (C dominates D),
3. $T' > R' > P' > S'$ (the game becomes a PD), and
4. $R' > T' > P' > S'$ (the game becomes a SH). In this last case the game is a proper SH (C and D are bistable and all-D is the risk-dominant equilibrium) when

$r > 0.5$. When $r < 0.5$, all-C is both payoff and risk dominant.

Finally, in the case of the SH the ordering of the payoffs is not importantly affected, but the unstable equilibrium moves towards $p = 1/2$, thus reducing the basin of attraction of all-D, i.e. the riskiness of all-C.

Broadly speaking, conformist transmission can promote cooperation in the PD by turning it into a SH, and in the SH by diminishing the basin of attraction of all-D. In the SD game, results are dependent on the cost-to-benefit ratio of mutual cooperation. For $r < 0.5$, cooperation is generally favoured: all-C can become the only stable equilibrium (when $R' > T' > S' > P'$), or the risk dominant equilibrium (when $R' > T' > P' > S'$). For $r > 0.5$ the opposite happens, with all-D possibly becoming the only stable equilibrium (when $T' > R' > P' > S'$) or the risk-dominant equilibrium (when $R' > T' > P' > S'$).

Although conformist transmission opens the possibility of a cooperative equilibrium in the PD and diminishes the riskiness of engaging in cooperative actions in the SH, populations with an initial majority of defectors are always doomed to a non-cooperative equilibrium in these two games. In the SD case, defection prevails for $r > 0.5$, and this for any amount of conformism. In this sense, conformist transmission alone is unable to sustain cooperation in both PD and SH, and it promotes cooperation for the SD game only when $r < 0.5$. For cooperation to be sustained, other mechanisms are necessary to be present along with conformism. Punishment has been suggested as one such possible mechanism (Henrich and Boyd, 2001). In the next section, we explore another mechanism: graph reciprocity.

Simulation results for the case of medium-sized, spatially structured populations

Here, the evolutionary dynamics of the three social dilemmas discussed above are studied by means of computer simulations for the case of medium-sized populations (1024 individuals) organised into a 32×32 square lattice with periodic boundary conditions. For the three games, the rescaled versions presented in the last section were used[7].

Square lattices were implemented using both Moore and von Neumann neighbourhoods with ranges equal to 1. Simulations were conducted using each of the three imitation rules previously defined (IIB, RD1, RD2), varying values of the game parameters (b in the PD, r in SD and g in the SH) and different amounts of conformism ($\alpha \in \{0.0, 0.125, 0.25, 0.375, 0.5\}$). Agents were updated synchronously.

For each simulated condition, 50 runs were executed. Each simulation was initialised with 50% cooperators and terminated whenever the population converged to any of the two absorbing states (all-C, all-D) or after 3000 simulation steps. In this last case, the equilibrium proportions of cooperators were calculated by averaging over the last 1000 time steps of each run, well after transients have passed.

Fig. 4 shows the average level of cooperation in equilibrium for the Moore neighbourhood case. Results for the von Neumann neighbourhood case are qualitatively similar and are not reproduced here for reasons of space. In the figures corresponding to the SD game, the dashed lines represent the equilibrium fraction of cooperators predicted by Eq. 6 (the well-mixed case).

Fig. 4 shows how cultural transmission including a conformist component consistently promotes higher levels of cooperation than payoff-based biased transmission alone for both the PD and the SH. Moreover, the larger the amount of conformism, the larger the proportion of cooperators at equilibrium, as it can be seen from the nice ordering of the curves for different values of α. For the SD game, the addition of conformist bias results in higher frequencies of cooperators for small r but also in lower frequencies of cooperators for large r. Thus, the general observations made for the effects of conformist transmission on the well-mixed case continue to hold for the case of spatially structured populations, i.e. that conformism promotes cooperation in the PD and the SH for the whole range of their game parameters, and that it promotes cooperation in the SD game for $r < 0.5$ while inhibiting cooperation for $r > 0.5$.

Regarding the effects of embedding the population in a lattice, our results confirm those already classic in evolutionary game theory: spatial structure promotes cooperation in the PD (Nowak and May, 1992; Nowak et al., 1994) and the SH (Skyrms, 2003), but can inhibit cooperation in the SD game (Hauert and Doebeli, 2004). In general, for the SD game, cooperators in a lattice do better than their counterparts in a well-mixed population for *a*) $\alpha < 0.25$ and small r, and *b*) $\alpha > 0.25$ and large r.

Notice that these qualitative results do not depend on the specific imitation rule being used. However, quantitative results do depend on the specificities of these rules. For instance, the higher stochasticity of the RD2 with respect to the other two imitation rules seems to hinder the evolution of cooperation in the PD and SH games, where only moderate levels of cooperation can be sustained, and only for very small b or very large g.

Conclusions

We have augmented traditional evolutionary graph-theoretic models with conformist transmission (the tendency to imitate common behaviours) and studied the effects of this extension on the evolutionary dynamics of social dilemmas. From a replicator dynamics perspective, the addition of conformism is equivalent to a simple transformation of the payoff matrix favouring the stability of pure equilibria. In par-

[7] We effectively set $P = \epsilon = 0$ in the PD.

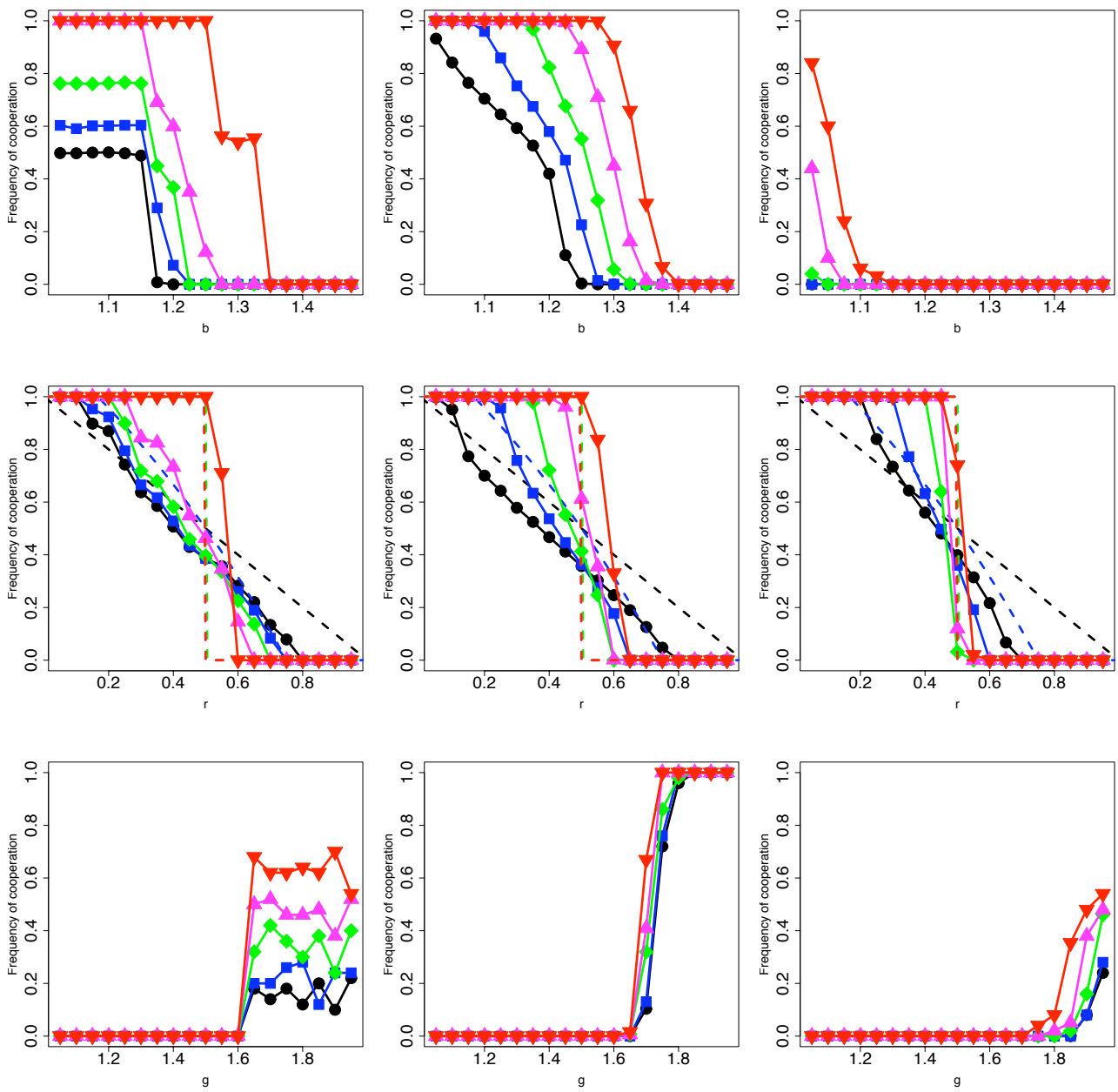


Figure 4: Average values of the equilibrium proportion of cooperators as a function of the game parameter for the PD (*first row*), the SD game (*second row*) and the SH (*third row*). Results are given for IIB (*first column*), RD1 (*second column*) and RD2 (*third column*) imitation rules and different amounts of conformism: $\alpha = 0.0$ (*black*), $\alpha = 0.125$ (*blue*), $\alpha = 0.25$ (*green*), $\alpha = 0.375$ (*magenta*) and $\alpha = 0.5$ (*red*). For the SD game, the corresponding proportions of cooperators in well-mixed populations for each value α are also reported (*dashed lines*).

ticular, a Prisoner's Dilemma can become a Stag Hunt, and a Snowdrift can become a Stag Hunt, a Prisoner's Dilemma or a game in which cooperation dominates defection. In the Stag Hunt case, where both pure equilibria are already stable, conformist transmission moves the unstable equilibrium towards $p = 1/2$, thus reducing the basin of attraction of the non-cooperative equilibrium. Although unable to sustain cooperation by its own when cooperators are not the majority at the beginning of the evolutionary process, conformist transmission enhances cooperation when other mechanisms, such as spatial locality, are also present in the model, at least for the PD and the SH cases. For the spatial SD, conformism can also be shown to promote higher levels of cooperative behaviour, but only for small cost-to-benefit ratios.

Acknowledgements. This work is funded by the Future and Emerging Technologies programme IST-STREP of the European Community, under grant number IST-034632 (PERPLEXUS). The author would like to thank Isis Fuchs, Olivier Jorand, Andrés Pérez-Uribe and Henri Volken for valuable discussions, and the three anonymous reviewers for their comments on this paper.

References

Asch, S. E. (1951). *Groups, leadership and men*, chapter Effects of group pressure upon the modification and distortion of judgment, pages 177–190. Pittsburgh, PA: Carnegie Press.

Boyd, R. and Richerson, P. J. (1985). *Culture and the evolutionary process*. Chicago: University of Chicago Press.

Boyd, R. and Richerson, P. J. (2002). Group beneficial norms can spread rapidly in a structured population. *Journal of Theoretical Biology*, 215(3):287–296.

Carpenter, J. P. (2004). When in rome: conformity and the provision of public goods. *Journal of Socio-Economics*, 33(4):395–408.

Coultas, J. C. (2004). When in rome... an evolutionary perspective on conformity. *Group Processes Intergroup Relations*, 7(4):317–331.

Gintis, H. (2000). *Game theory evolving*. Princeton: Princeton University Press.

Hauert, C. and Doebeli, M. (2004). Spatial structure often inhibits the evolution of cooperation in the snowdrift game. *Nature*, 428(6983):643–646.

Henrich, J. (2001). Cultural transmission and the diffusion of innovations: Adoption dynamics indicate that biased cultural transmission is the predominate force in behavioral change and much of sociocultural evolution. *American Anthropologist*, 103:992–1013.

Henrich, J. and Boyd, R. (1998). The evolution of conformist transmission and between-group differences. *Evolution and Human Behavior*, 19:215–242.

Henrich, J. and Boyd, R. (2001). Why people punish defectors: conformist transmission stabilizes costly enforcement of norms in cooperative dilemmas. *Journal of Theoretical Biology*, 208:79–89.

Hofbauer, J. and Sigmund, K. (1998). *Evolutionary Games and Population Dynamics*. Cambridge: N. Y.

Lieberman, E., Hauert, C., and Nowak, M. A. (2005). Evolutionary dynamics on graphs. *Nature*, 433(7023):312–316.

Maynard Smith, J. (1982). *Evolution and the Theory of Games*. Cambridge Univ. Press, Cambridge.

McElreath, R. and Boyd, R. (2007). *Mathematical Models of Social Evolution: A Guide for the Perplexed*. University Of Chicago Press.

Nowak, M. (2006). *Evolutionary Dynamics*. Cambridge, MA: Harvard University Press.

Nowak, M., Bonhoeffer, S., and May, R. (1994). Spatial Games and the Maintenance of Cooperation. *Proceedings of the National Academy of Sciences of the United States of America*, 91(11):4877–4881.

Nowak, M. A. and May, R. M. (1992). Evolutionary games and spatial chaos. *Nature*, 359(6398):826–829.

Santos, F. C. and Pacheco, J. M. (2005). Scale-free networks provide a unifying framework for the emergence of cooperation. *Phys. Rev. Lett.*, 95(9):098104–4.

Skyrms, B. (2003). *The Stag-Hunt Game and the Evolution of Social Structure*. Cambridge: Cambridge University Press.

Skyrms, B. (2005). Dynamics of conformist bias. *The Monist*, 88(2):259–269.

Szabo, G. and Fath, G. (2007). Evolutionary games on graphs. *Physics Reports*, 446(4-6):97–216.

Taylor, P. and Jonker, L. (1978). Evolutionary stable strategies and game dynamics. *Mathematical Biosciences*, 40:145–156.

Hawks and Doves in an Artificial Dynamically Structured Society

E. Pestelacci and M. Tomassini
Information Systems Institute, HEC, University of Lausanne, Switzerland
{enea.pestelacci,marco.tomassini@unil.ch}

Abstract

Using a dynamical network model of society, we show that cooperation is the norm in the Hawks-Doves game when individuals are allowed to break ties with undesirable neighbors and to make new acquaintances in their extended neighborhood. This is an interesting result, as standard theory for mixing populations prescribes that a certain fraction of defectors must always exist at equilibrium. We discuss the empirical network structure reasons that allow cooperators to thrive in the population.

Introduction and Previous Work

Hawks-Doves, also known as Chicken, is a two-person, symmetric game with the following payoff bi-matrix:

	C	D
C	(R,R)	(S,T)
D	(T,S)	(P,P)

In this matrix, D stands for strategy "hawk", and C stands for strategy "dove". Metaphorically, a hawkish behavior means a strategy of fighting, while a dove, when facing a confrontation, will always yield. R is the *reward* the two players receive if they both cooperate (C), P is the *punishment* for bilateral defection (D), and T is the *temptation*, i.e. the payoff that a player receives if it defects, while the other cooperates. In this case, the cooperator gets the *sucker's payoff* S. The game has a structure similar to that of the *Prisoner's Dilemma* (Axelrod, 1984). However, the ordering of payoffs for the Prisoner's Dilemma is $T > R > P > S$ rendering defection the best rational individual choice, while in the Hawks-Doves game the ordering is $T > R > S > P$ thus making mutual defection, i.e. result (D,D), the worst possible outcome. Note that in game theory, as long as the above orderings are respected, the actual numerical payoff values do not matter (Vega-Redondo, 2003).

In contrast to the Prisoner's Dilemma which has a unique Nash equilibrium that corresponds to both players defecting, the strategy pairs (C,D) and (D,C) are both Nash equilibria of the Hawks-Doves game in pure strategies, so the game is *antagonistic*, and there is a third equilibrium in mixed strategies in which strategy D is played with probability p, and strategy C with probability $1 - p$, where $0 \leq p \leq 1$ depends on the actual payoff values. We recall that a Nash equilibrium is a combination of strategies (pure or mixed) of the different players such that any unilateral deviation by any agent from this combination can only decrease her expected payoff (Vega-Redondo, 2003).

As is the case for the Prisoner's Dilemma (Axelrod, 1984; Lindgren and Nordahl, 1994), Hawks-Doves, for all its simplicity, appears to capture some important features of social interactions. In this sense, it applies in many situations in which "parading", "retreating", and "escalating" are common. One striking example of a situation that has been thought to lead to a Hawks-Doves dilemma is the Cuban missile crisis in 1962 (Poundstone, 1992). Other well known applications are found in the animal kingdom (Maynard Smith, 1982).

Considering now not just two players but rather a large, mixing population of identical players where randomly chosen pairs play a sequence of two-person games, *evolutionary game theory* (Hofbauer and Sigmund, 1998) prescribes that the only *Evolutionary Stable Strategy* (ESS) of the population is the mixed strategy, giving rise, at equilibrium, to a frequency of hawks in the population equal to p, the probability with which strategy hawk, i.e. D, would be played in a mixed strategy.

In the case of the Prisoner's Dilemma, one finds a unique ESS with all the individuals defecting. However, Nowak and May (1992) showed that cooperation in the population is sustainable in the Prisoner's Dilemma under certain conditions, provided that the network of the interactions between players has a lattice spatial structure. Killingback and Doebeli (1996) extended the spatial approach to the Hawks-Doves game and found that a planar lattice structure with only nearest-neighbor interactions may favor cooperation, i.e. the fraction of doves in the population is often higher than what is predicted by evolutionary game theory. In a more recent work however, Hauert and Doebeli (2004) were led to a different conclusion, namely that spatial struc-

ture does not seem to favor cooperation in the Hawks-Doves game. Further studies (Tomassini et al., 2006) extended the structured population approach to other graph structures representing small worlds. Small-world networks are produced by randomly rewiring a few links in an otherwise regular lattice such as a ring or a grid (Watts and Strogatz, 1998). These "shortcuts", as they are called, give rise to graphs that have short path lengths between any two nodes in the average as in random graphs, but in contrast to the latter, also have a great deal of local structure as conventionally measured by the *clustering coefficient*[1]. These structures are much more typical of the networks that have been analyzed in technology, society, and biology than regular lattices or random graphs (Newman, 2003). In Tomassini et al. (2006) it was found that cooperation may be either enhanced or inhibited in small-world networks depending on the gain-to-cost ratio $r = R/(R-P)$, and on the strategy update rule. But Watts–Strogatz small worlds and scale-free networks, although more realistic than lattices or random graphs, are not faithful representation of typical social networks. Santos and Pacheco (2005) and Santos et al. (2006b) extended the study of the Hawk-Doves game to scale-free networks, i.e. to networks having a power-law distribution of the connectivity degree (Newman, 2003). They found that cooperation is remarkably enhanced in them with respect to previously described population structures through the existence of highly connected cooperator hubs. However, pure static scale-free networks are not found among the typical socioeconomic networks that have been studied (Amaral et al., 2000; Newman, 2001, 2003). Using real and model static social networks, Luthi et al. (2008) also found that cooperation is enhanced, although to a lesser degree, thanks to the existence of tight clusters of cooperators that reinforce each other. Static networks having a resemblance with actual social networks are a good starting point; however, the static approach ignores fluctuations and non-equilibrium phenomena. Instead, real social networks are dynamical, i.e. nodes may join the network forming new links, and old nodes may leave it as social actors come and go. Furthermore, new links between agents already in the network may also form or be dismissed. Thus, the motivation of the present work is to study the co-evolution of strategy and network structure and to investigate under which conditions cooperative behavior may emerge and be stable in the Hawks-Doves game. A related goal is to study the topological structures of the emergent networks and their relationships with the strategic choices of the agents. Some previous work has been done on evolutionary games on dynamic networks (Zimmermann and Eguíluz, 2005; Luthi et al., 2006; Santos et al., 2006a). The only one citing the Hawks-Doves game is (Santos et al., 2006a) but our model differs in several important respects and we obtain new results on the structure of the cooperating clusters.

The paper is organized as follows. In the next section we present our dynamical models. This is followed by an exhaustive numerical study of the game's parameter space. After that we describe and discuss the statistical structure of the emerging networks and finally we give our conclusions.

Model and Dynamics

Our model is strictly local. No player uses information other than the strength of the links with its neighbors and the knowledge of her own payoff and, indirectly, the payoffs of her immediate neighbors. Moreover, as the model is an evolutionary one, no rationality, in the sense of game theory, is needed (Vega-Redondo, 2003). Players just adapt their behavior such that they imitate more successful strategies in their environment with higher probability. Furthermore, they are able to locally assess the worth of an interaction and possibly dismiss a relationship that does not pay off enough. The model and its dynamics are described in detail in the following sections.

Network and Interaction Structure. The network of agents is represented by an undirected graph $G(V, E)$, where the set of vertices V represents the agents, while the set of edges (or links) E represents their symmetric interactions. The population size N is the cardinality of V. A neighbor of an agent i is any other agent j such that there is an edge $\{ij\} \in E$. The set of neighbors of i is called V_i and its cardinality is the degree k_i of vertex $i \in V$. The average degree of the network will be called $\bar{k}$. Although there is formally a single undirected link between a player i and another player $j \in V_i$, we shall maintain two links: one going from i to j and another one in the reverse direction. Each link has a weight or "force" f_{ij} (respectively f_{ji}). This weight, say f_{ij}, represents in an indirect way the "trust" player i attributes to player j. This weight may take any value in $[0, 1]$ and its variation is dictated by the payoff earned by i in each encounter with j, as explained below.

The idea behind the introduction of the forces f_{ij} is loosely inspired by the potentiation/depotentiation of connections between neurons in neural networks, an effect known as the *Hebb rule* (Hebb, 1949). In our context, it can be seen as a kind of "memory" of previous encounters. However, it must be distinguished from the memory used in iterated games, in which players "remember" a certain number of previous moves and can thus conform their future strategy on the analysis of those past encounters (Vega-Redondo, 2003). Our interactions are strictly one-shot, i.e. players "forget" the results of previous rounds and cannot recognize

[1] The clustering coefficient $\mathcal{C}_i$ of a node i is defined as $\mathcal{C}_i = 2E_i/k_i(k_i - 1)$, where E_i is the number of edges in the neighborhood of i. Thus $\mathcal{C}_i$ measures the amount of "cliquishness" of the neighborhood of node i and it characterizes the extent to which nodes adjacent to node i are connected to each other. The clustering coefficient of the graph is simply the average over all nodes: $\mathcal{C} = \frac{1}{N}\sum_{i=1}^{N} \mathcal{C}_i$ (Newman, 2003).

previous partners and their possible playing patterns. However, a certain amount of past history is implicitly contained in the numbers f_{ij} and this information may be used by an agent when it will come to decide whether or not an interaction should be dismissed (see below).

We also define a quantity s_i called *satisfaction* of an agent i which is the sum of all the weights of the links between i and its neighbors V_i divided by the total number of links of that node k_i:

$$s_i = \frac{\sum_{j \in V_i} f_{ij}}{k_i}.$$

We clearly have $0 \leq s_i \leq 1$.

Initialization. The constant size of the networks during the simulations is $N = 1000$. The initial graph is generated randomly with a mean degree $\bar{k} = 10$ which is of the order of those actually found in many social networks; see, for instance, (Newman, 2003). Players are distributed uniformly at random over the graph vertices with 50% cooperators. Forces between any pair of neighboring players are initialized at 0.5.

We use a parameter q which is a real number in $[0, 1]$ and it represents the frequency with which an agent wishes to dismiss a link with one of its neighbors. The higher q, the faster the link reorganization in the network. This parameter has a role analogous to the "time scale" parameter of (Santos et al., 2006a) and it controls the speed at which topological changes occur in the network. All the agents have the same value of q. It is an important consideration, as social networks may structurally evolve at widely different speeds, depending on the kind of interaction between agents. For example, e-mail networks change their structure at a faster pace than, say, scientific collaboration networks.

Update Timing. Usually, agents systems such as the present one, are updated synchronously (Nowak and May, 1992; Santos and Pacheco, 2005; Zimmermann and Eguíluz, 2005). However, strictly speaking, simultaneous update is physically unfeasible as it would require a global clock, while real extended systems in biology and society in general have to take into account finite signal propagation speed. Simultaneity may cause some artificial effects in the dynamics which are not observed in real systems (Huberman and Glance, 1993; Luthi et al., 2006). On the other hand, updating a randomly chosen agent at a time also seems a rather arbitrary extreme case that is not likely to represent reality very accurately. We have thus chosen to update our population in a partially synchronous manner. In practice, we define a fraction $f = n/N$ (with $N = an, a \in \mathbb{N}$) and, at each simulated discrete time step, we update only $n \leq N$ agents randomly chosen with replacement. This is called a *microstep*. After N/n microsteps a whole population update, i.e. a *macrostep* will have taken place. With $n = N$ we recover the fully synchronous update, while $n = 1$ gives the extreme case of the fully asynchronous update. In this work we use $f = 0.01$.

Strategy and Link Dynamics

Here we describe in detail how individual strategies, links, and link weights are updated. Once a given node i is chosen to be activated, i.e. it belongs to the fraction f of nodes that are to be updated in a given microstep, i goes through the following steps:

- if the degree of agent i, $k_i = 0$ then player i is an isolated node. In this case a link with strength 0.5 is created from i to a player j chosen uniformly at random among the other $N - 1$ players in the network.
- otherwise,
  - either agent i updates its strategy according to a local *replicator dynamics* rule with probability $1 - q$ or, with probability q, agent i may delete a link with a given neighbor j and creates a new 0.5 force link with another node k ;
  - the forces between i and its neighbors V_i are updated

Let us now describe each step in more detail.

Strategy Evolution. We use a local version of replicator dynamics (RD) as described in (Luthi et al., 2008). The local dynamics of a player i only depends on its own strategy and on the strategies of the k_i players in its neighborhood V_i. Let us call π_{ij} the payoff player i receives when interacting with neighbor j. This payoff is defined as

$$\pi_{ij} = \sigma_i(t)\ M\ \sigma_j^T(t),$$

where M is the payoff matrix of the game and $\sigma_i(t)$ and $\sigma_j(t)$ are the strategies played by i and j at time t. The quantity

$$\widehat{\Pi}_i(t) = \sum_{j \in V_i} \pi_{ij}(t)$$

is the rescaled accumulated payoff (Luthi et al., 2008) collected by player i at time step t. The rule according to which agents update their strategies is the conventional RD in which strategies that do better than the average increase their share in the population, while those that fare worse than average decrease. To update the strategy of player i, another player j is drawn at random from the neighborhood V_i. It is assumed that the probability of switching strategy is a function ϕ of the payoff difference; ϕ is required to be monotonic increasing; here it has been taken linear (Hofbauer and Sigmund, 1998). Strategy σ_i is replaced by σ_j with probability

$$p_i = \phi(\widehat{\Pi}_j - \widehat{\Pi}_i). \quad (1)$$

The major differences with standard RD is that two-person encounters between players are only possible among neighbors, instead of being drawn from the whole population, and the latter is finite in our case. Other commonly used strategy update rules include imitating the best in the neighborhood (Nowak and May, 1992; Zimmermann and Eguíluz, 2005), or replicating in proportion to the payoff (Hauert and Doebeli, 2004; Tomassini et al., 2006).

Link Evolution. The active agent i, which has $k_i \neq 0$ neighbors will, with probability q, attempt to dismiss an interaction with one of its neighbors in the following way. Player i will look at its satisfaction s_i. The higher s_i, the more satisfied the player, since a high satisfaction is a consequence of successful strategic interactions with the neighbors. Thus, the natural tendency is to try to dismiss a link when s_i is low. This is simulated by drawing a uniform pseudo-random number $r \in [0, 1]$ and breaking a link when $r \geq s_i$. Assuming that the decision is taken to cut a link, which one, among the possible k_i, should be chosen? Our solution only relies on the strength of the relevant links. First a neighbor j is chosen with probability proportional to $1-f_{ij}$, i.e. the stronger the link, the less likely it is that it will be selected. This intuitively corresponds to i's observation that it is preferable to dismiss an interaction with a neighbor j that has contributed little to i's payoff over several rounds of play. However, in our system dismissing a link is not free: j may "object" to the decision. The intuitive idea is that, in real social situations, it is seldom possible to take unilateral decisions: often there is a cost associated, and we represent this hidden cost by a probability $1-(f_{ij}+f_{ji})/2$ with which j may refuse to be cut away. In other words, the link is less likely to be deleted if j appreciates i, i.e. when f_{ji} is high. If the link is not cut there is no further attempt during the current microstep update.

Assuming that the $\{ij\}$ link is finally cut, how is a new link to be formed? The solution adopted here is inspired by the observation that, in social settings, links are usually created more easily between people who have a mutual acquaintance than those who do not. First, a neighbor k is chosen in $V_i \setminus \{j\}$ with probability proportional to f_{ik}, thus favoring neighbors i trusts. Next, k in turn chooses player l in his neighborhood V_k using the same principle, i.e. with probability proportional to f_{kl}. If i and l are not connected, a link $\{il\}$ is created, otherwise the process is repeated in V_l. Again, if the selected node, say m, is not connected to i, a new link $\{im\}$ is established. If this also fails, a new link between i and a randomly chosen node is created. In all cases the new link is initialized with a strength of 0.5 in both directions. This rewiring process is schematically depicted in Fig. 1 for the case in which a link can be successfully established between players i and l thanks to their mutual acquaintance k.

At this point, we would like to stress several important differences with previous work in which links can be dismissed in evolutionary games on networks. In (Zimmermann and Eguíluz, 2005), only links between defectors are allowed to be cut unilaterally and the study is restricted to the Prisoner's Dilemma. Instead, in our case, any link has a finite probability to be abandoned, even a profitable link between cooperators if it is recent, although links that are more stable, i.e. have high strengths, are less likely to be rewired. This smoother situation is made possible thanks to our bilateral view of a link which is completely different from the undirected choice made in (Zimmermann and Eguíluz, 2005). It also allows for a moderate amount of "noise" in the system, which could reflect to a certain extent the uncertainties present in the system.

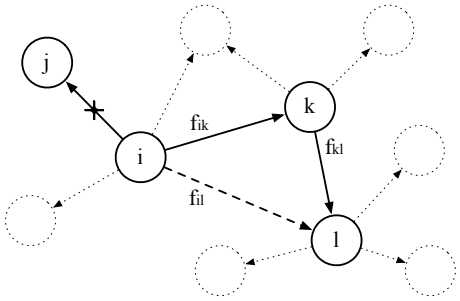


Figure 1: Illustration of the rewiring of link $\{ij\}$ to $\{il\}$. Agent k is chosen to introduce player l to i (see text).

In (Santos et al., 2006a), links can be cut by an unsatisfied player, where the concept of satisfaction is different from ours, and simply means that a cooperator or a defector will wish to break a link with a defector but there is no analogous of our "negotiation" process as the concept of link strength is absent. In (Luthi et al., 2006) links are cut according to a threshold decision rule and are rewired randomly anywhere in the network.

Updating the Link Strengths. Once the chosen agents have gone through their strategy or link update steps, the strengths of the links are updated accordingly in the following way:

$$f_{ij}(t+1) = f_{ij}(t) + \frac{\pi_{ij} - \bar{\pi}_{ij}}{k_i(\pi_{max} - \pi_{min})},$$

where π_{ij} is the payoff of i when interacting with j, $\bar{\pi}_{ij}$ is the payoff earned by i playing with j, if j were to play his other strategy, and π_{max} (π_{min}) is the maximal (minimal) possible payoff obtainable in a single interaction. This update is performed in both directions, i.e. both f_{ij} and f_{ji} are updated $\forall j \in V_i$.

Numerical Simulations

Simulation Parameters. We simulated the Hawks-Doves game with the dynamics described above exploring the entire game space by limiting our study to the variation of only

two game parameters. We set $R = 1$ and $P = 0$ and the two parameters are $1 \leq T \leq 2$ and $0 \leq S \leq 1$. Setting $R = 1$ and $P = 0$ determines the range of S (since $T > R > S > P$) and gives an upper bound of 2 for T, due to the $2R > T + S$ constraint, which ensures that mutual cooperation is preferred over an equal probability of unilateral cooperation and defection. Note however, that the only valid value pairs of (T, S) are those that satisfy the latter constraint.

We simulated networks of size $N = 1000$, randomly generated with an average degree $\bar{k} = 10$ and randomly initialized with 50% cooperators and 50% defectors. In all cases, the parameters are varied between their two bounds in steps of 0.1. For each set of values, we carry out 50 runs of at most 10000 macrosteps each, using a fresh graph realization in each run. After an initial transient period, the system is considered to have reached a pseudo-equilibrium strategy state when the strategy of the agents (C or D) does not change over 150 further macrosteps, which means 15×10^4 individual updates. We speak of pseudo-equilibria or steady states and not of true evolutionary equilibria because there is no analog of equilibrium conditions in the dynamical systems sense.

Cooperation and Stability. Cooperation results in contour plot form are shown in Fig. 2. We remark that, as observed in other structured populations, cooperation is achieved in almost the whole configuration space. Thus, the added degree of freedom represented by the possibility of refusing a partner and choosing a new one does indeed help to find player's arrangements that help cooperation. When considering the dependence on the fluidity parameter q, one sees in Fig. 2 that the higher q, the higher the cooperation level, although the differences are small, since full cooperation prevails already at $q = 0.2$. This was a somewhat expected result, since being able to break ties more often clearly gives cooperators more possibilities for finding and keeping fellow cooperators to interact with. The same effect has been previously observed in (Santos et al., 2006a) with the use of a different model both for strategy evolution and tie breaking. Thus the finding is robust and relatively independent of the other details of the models.

Compared with the level of cooperation observed in simulations in static networks, we can say that results are consistently better for co-evolving networks. For all values of q (Fig. 2) there is significantly more cooperation than what was found in model and real social networks (Luthi et al., 2008) where the same local replicator dynamics was used but with the constraints imposed by the invariant network structure. A comparable high cooperation level has only been found in static scale-free networks (Santos et al., 2006b), which is theoretically interesting, but those topologies are unlikely models for social networks, which often show fat-tailed degree distribution functions but not pure power-laws (see, for instance, (Amaral et al., 2000; Newman, 2001)). As a further indication of the latter, we shall see later that, indeed, emerging networks do not have a power-law degree distribution.

The above considerations are all the more interesting when one observes that the standard RD result is that the only asymptotically stable state for the game is a polymorphic population in which there is a fraction α of doves and a fraction $1 - \alpha$ of hawks, with α depending on the actual numerical payoff matrix values. To see the positive influence of making and breaking ties we can compare our results with what is prescribed by the standard RD solution. Referring to the payoff table of the Introduction section, let's assume that the column player plays C with probability α and D with probability $1 - \alpha$. In this case, the expected payoffs of the row player are:

$$E_r[C] = \alpha R + (1 - \alpha)S$$

and

$$E_r[D] = \alpha T + (1 - \alpha)P$$

The row player is indifferent to the choice of α when $E_r[C] = E_r[D]$. Solving for α gives:

$$\alpha = \frac{P - S}{R - S - T + P}. \quad (2)$$

Since the game is symmetric, the result for the column player is the same and $(\alpha C, (1 - \alpha)D)$ is a NE in mixed strategies. We have numerically solved the equation for all the sampled points in the game's parameter space. Let us now use the following payoff values in order to bring them within the explored game space (remember that NEs are invariant w.r.t. such an affine transformation):

	C	D
C	$(1, 1)$	$(2/3, 4/3)$
D	$(4/3, 2/3)$	$(0, 0)$

Substituting in 2 gives $\alpha = 2/3$, i.e. the dynamically stable polymorphic population should be composed by about $2/3$ cooperators and $1/3$ defectors. Now, if one looks at Fig. 2 at the points where $S = 2/3$ and $T = 4/3$, one can see that the point, and the region around it, is one of full cooperation instead. Even within the limits of the approximations caused by the finite population size and the local dynamics, the non-homogeneous graph structure and an increased level of tie rewiring has allowed the cooperation to be greatly enhanced with respect to the theoretical predictions of standard RD.

Structure of the Emerging Networks

In this section we present a statistical analysis of the global and local properties of the networks that emerge when the pseudo-equilibrium states of the dynamics are attained. First, the mean degree $\bar{k}$ increases only slightly and tends

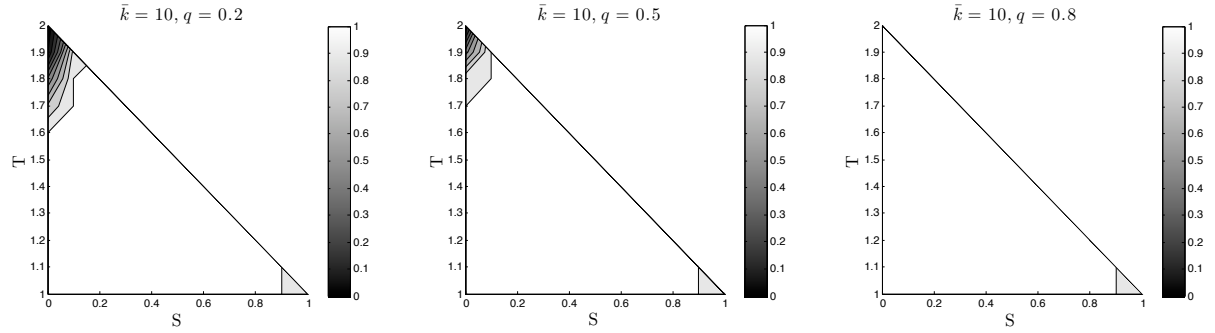


Figure 2: Average cooperation values for the Hawks-Doves game when the steady-state has been reached. Results are the average of 50 runs.

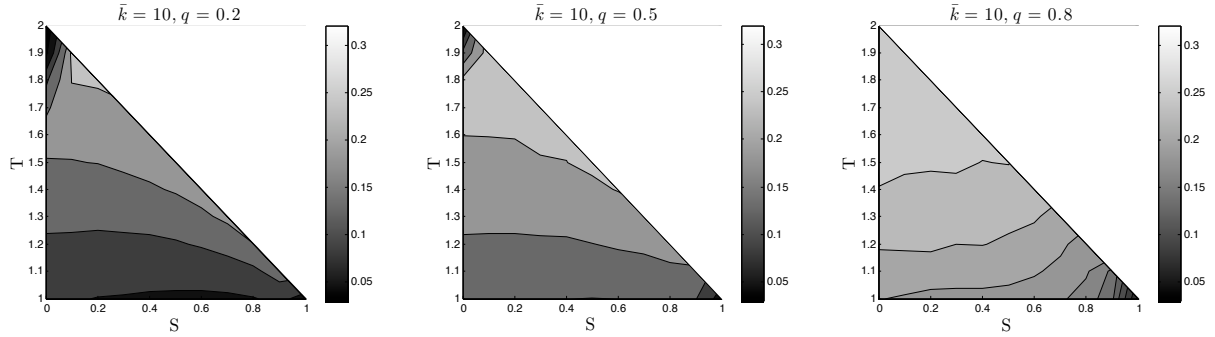


Figure 3: Average values of the clustering coefficient over 50 runs.

to stabilize around $\bar{k} = 11$. Next, let us consider first the clustering coefficient $\mathcal{C}$, which was previously defined. Random graphs are locally homogeneous in the average and for them $\mathcal{C}$ is simply equal to the probability of having an edge between any pair of nodes independently. In contrast, real networks have local structures and thus higher values of $\mathcal{C}$. Fig. 3 gives the average clustering coefficient $\bar{\mathcal{C}} = \frac{1}{50}\sum_{i=1}^{50}\mathcal{C}$ for each sampled point in the Hawks-Doves configuration space, where 50 is the number of network realizations used for each simulation. The networks self-organize through dismissal of partners and choice of new ones and they acquire local structure, since the clustering coefficients are higher than that of the random graph with the same number of edges and nodes, which is $\bar{k}/N = 10/1000 = 0.01$. This effect was expected, since the model favors relinking with closer neighbors rather than arbitrary individuals. The clustering tends to increase with q (i.e. from left to right in Fig. 3).

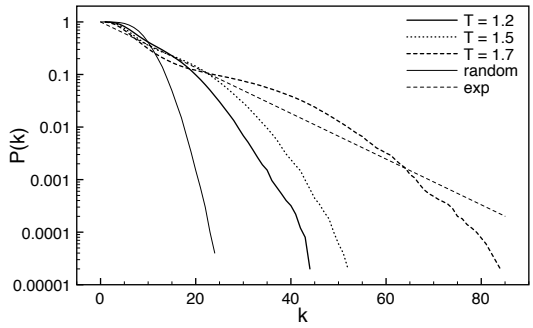


Figure 4: Empirical cumulative degree distribution functions for three different values of the temptation T. A Poissonian and an exponential distribution are also plotted for comparison. Distributions are discrete, the continuous lines are only a guide for the eye. Lin-log scales.

The *degree distribution function* (DDF) $p(k)$ of a graph represents the probability that a randomly chosen node has degree k. Random graphs are characterized by DDF of Poissonian form $p(k) = \bar{k}^k e^{-\bar{k}}/k!$, while social and technological real networks often show long tails to the right, i.e. there are nodes that have an unusually large number of neighbors (Newman, 2003). In some extreme cases the DDF has a power-law form $p(k) \propto k^{-\gamma}$; the tail is particularly extended and there is no characteristic degree. The *cumulative degree distribution function* (CDDF) is just the probability that the degree is greater than or equal to k and has the advantage of being less noisy for high degrees. Fig. 4 shows the CDDFs for the Hawks-Doves for three values of T, and $q = 0.5$. A Poisson and an exponential distribution are also shown for comparison. The Poisson curve actually represents the initial degree distribution of the (random) popula-

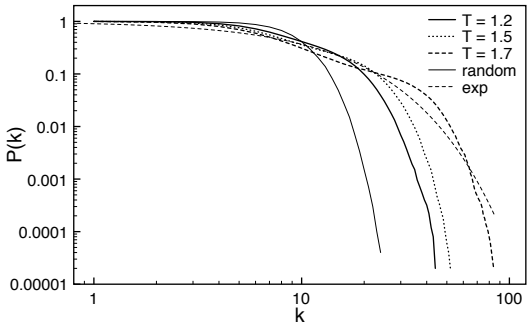


Figure 5: Empirical cumulative degree distribution functions for three different values of the parameter T. Log-log scales.

tion graph. The distributions are far from the Poissonian that would apply if the networks would remain essentially random. However, they are also far from the power-law type, which would appear as a straight line in the log-log plot of Fig 5. Although a reasonable fit with a single law appears to be difficult, these empirical distributions are closer to exponentials, in particular the curve for $T = 1.7$. It can be observed that the distribution is broader the higher T is. In fact, although cooperation is attained nearly everywhere in the game's configuration space, higher values of the temptation T mean that agents have to rewire their links more extensively, which results in a higher number of neighbors for some players, and thus it leads to a longer tail in the CDDF.

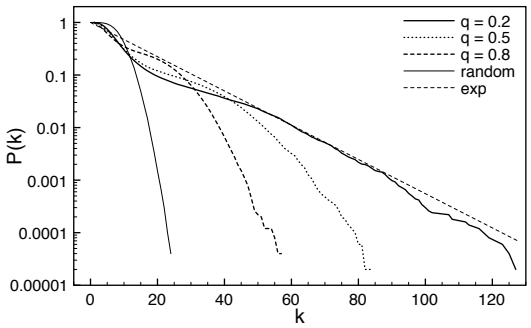


Figure 6: Empirical cumulative degree distribution functions for three different values of the temptation q. Lin-log scales.

The influence of the q parameter on the shape of the degree distribution functions is shown in Fig. 6 where average curves for three values of q, $T = 1.7$, and $S = 0.2$, are reported. For high q, the cooperating steady-state is reached faster, which gives the network less time to rearrange its links. For lower values of q the distributions become broader, despite the fact that rewiring occurs less often, because cooperation in this region is harder to attain and more simulation time is needed.

Cooperator Clusters

From the results of the previous section, it appears that a larger amount of cooperation than what is predicted by the standard theory for mixing populations can be reached when ties can be broken and rewired. We have seen that this dynamics causes the graph to acquire local structure, and thus to loose its initial randomness in terms of links. In other words, the network self-organizes in order to allow players to cooperate as much as possible. At the microscopic, i.e. agent level, this happens through the formation of clusters of players using the same strategy. Fig. 7 shows one typical cooperator cluster .

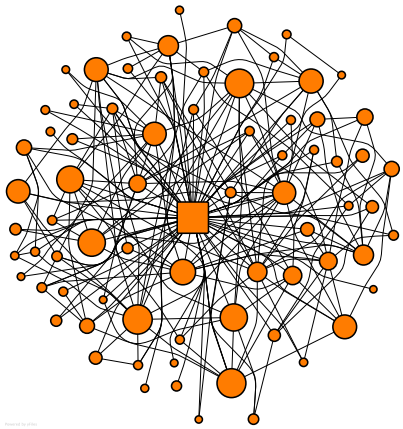


Figure 7: A typical cooperator cluster. Links to the rest of the network have been suppressed for clarity. The size of a node is proportional to its connectivity in the whole graph. The most connected central cooperator is shown as a square.

In the figure one can clearly see that the central cooperator is a highly connected node and there are many links also between the other neighbors. Such a tightly packed structure has emerged to protect cooperators from defectors that, at earlier times, were trying to link to cooperators to exploit them. These observations help understand why the degree distributions are long-tailed (see previous section), and also the higher values of the clustering coefficient in this case.

Conclusions

In this paper we have introduced a new dynamical population structure for agents playing a series of two-person Hawks and Doves game. The most novel feature of the model is the introduction of a variable strength of the bidirectional social ties between pairs of players. These strengths change dynamically and independently as a function of the relative satisfaction of the two end points when playing with their immediate neighbors in the network. A player may wish to break a tie to a neighbor and the probability of cutting the link is higher the weaker the directed link strength is. The ensemble of weighted links implicitly represent a kind of memory of past encounters although, tech-

nically speaking, the game in not iterated. The model takes into account recent knowledge coming from the analysis of the structure and of the evolution of social networks and, as such, should be a better approximation of real social conflicting situations than static graphs such as regular grids. In particular, new links are not created at random but rather taking into account the "trust" a player may have on her relationally close social environment as reflected by the current strengths of its links. This, of course, is at the origin of the de-randomization and self-organization of the network, with the formation of stable clusters of cooperators. The main result concerning the nature of the pseudo-equilibrium states of the dynamics is that cooperation is greatly enhanced in such a dynamical artificial society. This is encouraging, as the Hawks-Doves game is a paradigm for a number of social and political situations in which aggressivity play an important role. The standard result is that bold behavior does not disappear at evolutionary equilibrium. However, we have seen here that a certain amount of plasticity of the networked society allows for cooperation to be attained. Although the model is an extremely abstract one, it shows that there is place for peaceful resolution of conflict. Ongoing and future work for which there is no space here will deal with the stability of the system against massive and targeted defector invasions in a society of cooperators. Other strategy evolution models based on more refined forms of learning than simple imitation should also be investigated.

Acknowledgements. This work is funded by the Swiss National Science Foundation under grant number 200021-111816/1. We gratefully acknowledge this financial support.

References

Amaral, L. A. N., Scala, A., Barthélemy, M., and Stanley, H. E. (2000). Classes of small-world networks. *Proceedings of the National Academy of Sciences USA*, 97(21):11149–11152.

Axelrod, R. (1984). *The Evolution of Cooperation*. Basic Books, Inc., New-York.

Hauert, C. and Doebeli, M. (2004). Spatial structure often inhibits the evolution of cooperation in the snowdrift game. *Nature*, 428:643–646.

Hebb, D. O. (1949). *The Organization of Behavior*. Wiley, New York.

Hofbauer, J. and Sigmund, K. (1998). *Evolutionary Games and Population Dynamics*. Cambridge University Press, Cambridge, UK.

Huberman, B. A. and Glance, N. S. (1993). Evolutionary games and computer simulations. *Proceedings of the National Academy of Sciences USA*, 90:7716–7718.

Killingback, T. and Doebeli, M. (1996). Spatial evolutionary game theory: hawks and doves revisited. *Proceedings of the Royal Society of London B*, 263:1135–1144.

Lindgren, K. and Nordahl, M. G. (1994). Evolutionary dynamics of spatial games. *Physica D*, 75:292–309.

Luthi, L., Giacobini, M., and Tomassini, M. (2006). A minimal information prisoner's dilemma on evolving networks. In Rocha, L. M., editor, *Artificial Life X*, pages 438–444, Cambridge, Massachusetts. The MIT Press.

Luthi, L., Pestelacci, E., and Tomassini, M. (2008). Cooperation and community structure in social networks. *Physica A*, 387:955–966.

Maynard Smith, J. (1982). *Evolution and the Theory of Games*. Cambridge University Press.

Newman, M. E. J. (2001). Scientific collaboration networks. I. network construction and fundamental results. *Phys. Rev E*, 64:016131.

Newman, M. E. J. (2003). The structure and function of complex networks. *SIAM Review*, 45:167–256.

Nowak, M. A. and May, R. M. (1992). Evolutionary games and spatial chaos. *Nature*, 359:826–829.

Poundstone, W. (1992). *The Prisoner's Dilemma*. Doubleday, New York.

Santos, F. C. and Pacheco, J. M. (2005). Scale-free networks provide a unifying framework for the emergence of cooperation. *Phys. Rev. Lett.*, 95:098104.

Santos, F. C., Pacheco, J. M., and Lenaerts, T. (2006a). Cooperation prevails when individuals adjust their social ties. *PLOS Comp. Biol.*, 2:1284–1291.

Santos, F. C., Pacheco, J. M., and Lenaerts, T. (2006b). Evolutionary dynamics of social dilemmas in structured heterogeneous populations. *Proc. Natl. Acad. Sci. USA*, 103:3490–3494.

Tomassini, M., Luthi, L., and Giacobini, M. (2006). Hawks and doves on small-world networks. *Phys. Rev. E*, 73:016132.

Vega-Redondo, F. (2003). *Economics and the Theory of Games*. Cambridge University Press, Cambridge, UK.

Watts, D. J. and Strogatz, S. H. (1998). Collective dynamics of 'small-world' networks. *Nature*, 393:440–442.

Zimmermann, M. G. and Eguíluz, V. M. (2005). Cooperation, social networks, and the emergence of leadership in a prisoner's dilemma with adaptive local interactions. *Phys. Rev. E*, 72:056118.

Evolving Morphological and Behavioral Diversity Without Predefined Behavior Primitives

Peter-Paul Pichler and Lola Cañamero

Adaptive Systems Research Group, University of Hertfordshire
AL10 9AB Hatfield, Herts, UK
p.pichler@herts.ac.uk

Abstract

Virtual ecosystems, where natural selection is used to evolve complex agent behavior, are often preferred to traditional genetic algorithms because the absence of an explicitly defined fitness allows for a less constrained evolutionary process. However, these model ecosystems typically pre-specify a discrete set of possible action primitives the agents can perform. We think that this also constrains the evolutionary process with the modellers preconceptions of what possible solutions could be. Therefore, we propose an ecosystem model to evolve complete agents where all higher-level behavior results strictly from the interplay between extremely simple components and where no 'behavior primitives' are defined. On the basis of four distinct survival strategies we show that such primitives are not necessary to evolve behavioral diversity even in a simple and homogeneous environment.

Introduction

The evolution of 'novel' behavior by autonomous agents in any simulated system is determined by the predefined components and dynamics of that system. Consequently, the evolutionary possibilities of such a system are necessarily restricted and biased by the preconceptions of the designer. Artificial ecosystems like *Echo* (Holland, 1990), *PolyWorld* (Yaeger, 1994; Yaeger and Sporns, 2006), *LEE* (Menczer and Belew, 1996b,a), or *Geb* (Channon and Damper, 1998) use *natural selection* to overcome one of these biases imposed by the need for an explicit fitness function (artificial selection) in traditional genetic algorithms. All these models vary in the employed level of abstraction and in the details regarding constituents of the agents under evolutionary control (e.g. sensory system, controller, actuation, and morphological properties). However, all of these models resemble each other in that the agents adapt to choose from a predetermined and discrete set of *behavior primitives* which are assumed to be relevant for survival (e.g. *eating, mating, fighting, moving, turning*). This forces the designer of the system to explicitly decide what actions are available and possibly restricts the nature in which they are implemented by the agents.

In our model (Pichler and Cañamero, 2007) actuation is solely based on movement and reproductive investment. More complex behaviors (e.g. obstacle avoidance, fighting, foraging) are phenomena arising from the interplay between agents and environment. We are interested in *what* strategies arise and *how* they are implemented by low-level interactions of the agent components and its environment in the absence of pre-specified behavior primitives. We think that such an approach further reduces the designer bias and might be more conducive to evolving diverse and adaptive survival strategies.

The results of our simulation show that in such a setting behavioral diversity emerges even in a simple and homogeneous environment. We discuss four different and viable survival strategies and their properties on the level of the individual agent as well as of the whole population.

Virtual Ecosystem

The simulated environment is a space-continuous, time-discrete wrap-around world containing different kinds of objects. All objects in the environment are circular and share certain properties; they have an *energy signature* $e(t)$, a *solidness* ρ and a *radius* r. The energy signature indicates the amount of potentially consumable energy at time t. The solidness determines whether an agent can pass through an object ($\rho = 0$) or whether it collides with it ($\rho > 0$). For agents, radius and solidness are heritable parameters which affect their energy budget in critical ways. Their energy signature is the amount of energy remaining in the world after an agent's 'death' (see next section). Beside agents the environment contains two other types of object: *energy sources* and *obstacles*.

An energy source has a given maximum energy capacity $c > 0$ which defines its initial energy content. If an agent is in contact with an energy source, a certain amount of energy is transferred from the source to the agent and thereby consumed. The energy content of a source cannot fall below zero and 'grows' back to its capacity at a constant rate. Energy sources have an energy signature equal to their current energy content, a solidness of zero and a radius equal to their energy signature. Throughout the simulation

they are relocated to random positions with a certain probability. This mechanism was introduced to 'encourage' active foraging.

An obstacle is an object with zero energy capacity ($e(t) = 0$) but non-zero solidness. The radius of an obstacle equals its solidness. If an agent collides with an obstacle it is stopped and loses an amount energy depending on its speed and the properties of both objects.

Agent Components

The morphology of an agent is defined by its radius, its solidness and the number and configuration of actuators and sensors along the circumference of its body. Radius and solidness define the mass $m = \rho \cdot r^2\pi$ and the maximum energy capacity $c = \sqrt{m}$ of an agent. The capacity determines the amount and the rate at which the agent can absorb energy from an energy source. It also determines the cost for reproduction (e.g. $0.6 \cdot c$) and influences energy loss (damage) in a collision.

Sensing and Acting

We distinguish two types of sensors; *internal sensors* provide information about the internal variables of the agent and *external sensors* respond to properties of objects in the environment. All sensors function as input nodes to the neural controller network. We define two fixed internal sensors (life energy $l(t)$, reproductive depot $d(t)$) which cannot be removed by evolution. However, they are not necessarily connected to the rest of the network, so it is not predetermined whether or not they are used (see Fig. 2).

External sensors are defined by their position on the body and the type of stimulus they respond to. Each external sensor corresponds to an object property ($e(t)$,ρ, r). The information provided by the environment might roughly be thought of as a chemical gradient. The activation a of a sensor s is given by

$$a_s = \sum_{o \in O} \frac{v_o}{d_o^2 + 1} \tag{1}$$

where O is the set of all objects o within a maximum range, v is the value of the respective object property (e.g. solidness) and d is the distance between the sensor and the object.

Every agent has an actuator which regulates reproductive investment. At every time step an energy amount proportional to the activation of that actuator is transferred from the agent's life energy to its reproductive depot. If this depot reaches a certain threshold, the agent reproduces and an imperfect copy is placed close to it. If an agent 'dies' it is replaced by a corpse object with an initial energy content $f(0) = d(t) + 0.1 \cdot c$. Corpses are like energy sources, only their energy decreases (decay) over time. In addition to the reproductive node, an agent can have any number of locomotive actuators. Individually, these work like little jets or flagella, giving an impulse in a specific direction, but combined they can be used to generate more complex movement. A locomotive actuator is defined by its position on the agent's body and the angle it makes with it (see Fig. 1). This allows us to calculate a rotational and a translational component proportional to the activation of the actuator. The integration over all actuators yields the overall movement of the agent. An actuator is a node in the output layer of the neural controller network.

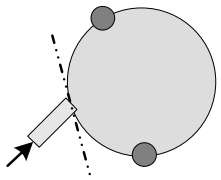


Figure 1: Exemplary body of a first generation agent with two sensors (round) and one locomotive actuator (arrow indicates impulse direction).

Neural Controller

Initial agents have few fixed components and no specific functionality. As described above, every agent's controller network has two internal sensors in the input layer and the reproductive actuator as a node in the output layer. Additionally, initial networks have a small random number of external sensors and locomotive actuators. The two layers are connected by a small random number of links (see Fig. 2). We use nodes with piecewise linear transfer functions and real valued (unbounded) connection weights. The output $N_o(t)$ of a node is given by:

$$N_o(t) = \begin{cases} 0 & \ldots \quad N_a(t) < \theta \\ 1 & \ldots \quad N_a(t) \geq \theta + I \\ \frac{N_a(t) - \theta}{I} & \ldots \quad otherwise \end{cases} \tag{2}$$

where $N_a(t)$ is the accumulated activation of the node, θ is the threshold and I defines a responsive range (slope of the function). The two parameters that define the operating range of a node (θ and I) and the connection weights are randomly initialized and evolved individually for each node and connection respectively. All nodes are arranged in layers and signals travel one layer per time step.

During evolution, both the structure and the parameters of the neural controller networks are freely evolved. Note that in many neuroevolution scenarios (e.g. (Kodjabachian and Meyer, 1998; Stanley and Miikkulainen, 2004)) neural network topologies are evolved to fit specific input and output structure (sensors and actuators). In this model the function and structure of the sensory and actuation systems are completely under evolutionary control. Variability operators

during reproduction may modify all parameters of existing structure and can also add or remove components (sensors, actuators, hidden layers, nodes, and connections) to form arbitrary recurrent networks.

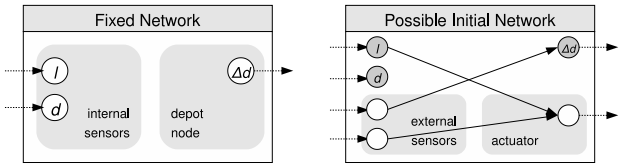


Figure 2: Controller networks have two fixed internal sensor nodes (life energy level $l(t)$, depot level $d(t)$) and the depot node $\Delta d(t)$ in the output layer (left). Additionally, every first generation agent has a (small) random number of sensors, actuators, and connections (right); All parameters are randomly initialized.

Metabolism

The energy budget of an agent is influenced by the properties of its body and its behavior. The base metabolic cost for an agent increases linearly in its mass and in the number of network components. Additional costs are variable and consist of locomotion costs (actuator activation in proportion to mass) and information processing costs (accumulated node activation). These relationships between the agent and the environment defined by the metabolic model shape the dynamics of the system. They create the selection pressures in this artificial ecosystem. All survival-relevant capabilities (sensing, acting, information processing, energy storage) come at an energetic cost. The balance of these aspects should create various trade-offs where agents can follow different strategies to successfully acquire and manage resources and generate a sustained population.

The energy balance of the agents and the resource renewal

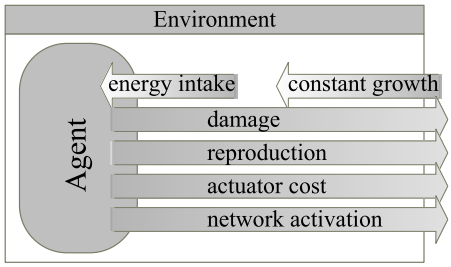


Figure 3: Total energy balance of agents and environment.

(energy sources) and decay (corpses) determine the total energy budget of the ecosystem (illustrated in Fig. 3) which is updated every time-step. The ecosystem is not a closed system with respect to energy as energy is added to it and dissipates via the metabolic consumption of the agents described by the following equations:

$$l_{t+1} = l_t + \Delta e_t - \Delta d_t - C_s - C_{ot} - C_{at} \qquad (3)$$

with:

$$C_s = c \cdot \mu_{cc} + \mu_{cc} \qquad (4)$$

$$C_{ot} = \sum_{n \in N} a_{nt} \cdot \mu_{co} \qquad (5)$$

$$C_{et} = \sum_{e \in E} a_{nt} \cdot \mu_{ce} \qquad (6)$$

where l is the life energy level of the agent at time t, Δe is the energy consumed, Δd the energy lost to collision damage, C_s are constant costs (with c the capacity of the agent), C_o are the costs for node activation a over all nodes n in N and C_e are costs for actuator activation a over all actuators e in E (including the investment in reproduction). The μ's are proportionality constants which were set by trial and error with the goal of balancing the influence of each aspect in a way that each would have a significant and similar impact while still allowing evolution to occur.

The energy content E of an energy source s at time t is:

$$E_s(t+1) = E_s(t) - \sum_{a \in A'} \Delta e^s_a(t) + \mu_g \qquad (7)$$

where a is an agent in the set A' of all agents which have consumed energy from source s at time t. The energy content of a source cannot be negative. This equation also holds for corpses if the constant growth rate $\mu_g > 0$ is replaced by a decay rate $\mu_d < 0$.

Reproduction

There are many possible ways to define a reproduction criterion in a foraging scenario like the one presented here. Two straightforward ideas are either a life time dependent criterion or using the life energy of an agent (see e.g. (Bedau et al., 1992)). Here, agents would periodically reproduce after a certain number of time steps or whenever their energy level reaches a specified threshold. However, solely 'optimizing' individual longevity disables survival strategies with short individual life times and thereby excludes potentially interesting dynamics like persistence vs. progeny trade-offs (Polani et al., 2006). The same is true when using the energy threshold as the single criterion; this strips the agent of much of its autonomy on how to manage the acquired resources. Using the reproductive actuator we have a reproduction criterion which gives the agents full control over *when* and to *what extent* they invest in reproduction. Whenever this node is activated an amount of energy proportional to the activation is transferred from the agent's life energy to its reproductive depot. Once this depot reaches a certain threshold, an imperfect copy is

placed close to the agent. Reproduction in our model is strictly asexual. Mutation operators exist to modify all body properties and the topology as well as all parameters of the controller network. While there is no final consensus about what is the best way to encode neural networks for artificial evolution it has been shown repeatedly (see e.g. (Stanley and Miikkulainen, 2002) or (Seys and Beer, 2006)) that the encoding has a crucial impact on the evolvability of the system. Keeping this in mind we presently use no 'genetic' encoding and all mutation operators are performed directly on the agent's object structure (this is equivalent to a direct encoding scheme).

Adaptation and development in this experiment occurs solely on an evolutionary scale through reproduction. Agents do not change or adapt during their lifetime. However, change on an evolutionary scale can only happen if a *turnover of generations* exists. In a classical genetic algorithm this turnover is an inherent property which is explicitly enforced by the design of the algorithm itself. In our model (and other models based on natural selection) this turnover of generations is to some extent an emergent property of the dynamics of the system. Because reproduction is 'optional' it is in some sense an adaptation itself. Agents have to actively invest their *life energy* into creating offspring and doing so jeopardizes their own survival because the invested energy is no longer available to them and reproducing creates a direct competitor in the vicinity. A first intuition might suggest that this would eventually lead to zero investment in reproduction. In this case evolution would cease to happen or, in fact, never happen at all. On second thought, however, it is clear that in a dynamic based on natural selection the notion of selecting *for* zero reproduction is contradictory as reproduction is the very vehicle of selection. Additionally, in an environment where individual survival is to some degree dependent on chance and thus effective immortality is unachievable, an infertile population is unsustainable and inevitably doomed. Randomly created agents are more often than not unable to survive for any length of time, let alone spare enough energy to reproduce if they even do so at all. To guarantee a certain number of agents in the environment we use a mechanism similar to (Yaeger, 1994). The *minimum enforced agents mechanism* (MEAM) creates new random agents whenever the total population size falls below a given threshold. Therefore, it guarantees that there are always agents present in the environment but becomes inactive once agents reproduce and successfully establish a sustained population of a certain size. Population size is therefore not fixed or constant, but depends on the environment and the properties of the evolved agents (see next section). To track the existence of a generational turnover we assign a *phylogenetic generation* (PG) to each agent. Agents created by the MEAM have a PG of zero, their offspring a PG of one, and so on. Evolution only occurs if this number increases.

Experiment and Results

To obtain the results discussed in this paper the simulation was run in relatively small 100x100 unit arenas (minimum agent size is 0.1 units) with 35 energy sources and 35 obstacles. Energy sources had an energy capacity of 1.0 and obstacles a solidness of 1.0. Objects were randomly placed in the environment following a uniform distribution. We repeated the simulation 85 times using different random seeds for the random number generator which determines object placement, initial agent configuration and all mutation operators. The minimum enforced number of agents was 15 in all 85 runs. Since in this setup there is no obvious 'convergence point', simulations were run until the average PG of a population was above 500 or a set maximum time was reached (80 hours). From each of the 76 'successful' runs (where a sustained population was established) a sample of the first 100 agents of PG ≥ 500 was taken.

Behavior and Morphology

In 76 out of 85 runs the MEAM eventually established a sustained population and evolution could occur. Actual computation time to reach this point depended greatly on a number of factors: the moment a sustained population was established, the average population size, the complexity of the average controller network, and the average lifetime of the individual agents. While in some runs a sustained population was established almost immediately, in 9 runs it did not happen at all before the maximum time was reached. These runs were discarded. A general observation was that all populations were quite homogeneous within a single run. On reason for that is that all agents within the population of one run were ultimately descendants of one respective *founder* agent which spawned the initial population. Other possible reasons are that the environments were rather small (an agent could travel 'around the world' frequently during its lifetime) and both obstacles and energy sources were uniformly distributed. In the following sections we will describe some of the evolved agents, their behavior and their morphologies (for illustrative examples that convey the nature of the evolved strategies much better than words we kindly refer the reader to the videos on the first author's website[1]). All agents in this experiment exhibited base movement (movement in the absence of stimulus). For the first part of the analysis of the results we distinguish three basic evolved behavior patterns solely by observable behavior:

- *Energy response:* agents show *some* response (e.g. slowing down) in the presence of or on contact with an energy source.

[1] http://homepages.feis.herts.ac.uk/~pp6bs/

	ER	EA	OA
Drifter	yes	no	no
Forager	yes	yes	no
Avoider	yes	no	yes
Allrounder	yes	yes	yes

Table 1: Classification of agents by the three observable behavior patterns: ER (energy response), EA (energy approach), and OA (obstacle avoid).

- *Energy approach:* agents change direction and actively try to approach an energy source.
- *Obstacle avoidance:* agents change their behavior in the presence of an object of non-zero solidness.

The definition of these behavioral patterns is intentionally careful. If an agent changes its behavior in response to an obstacle, it might do it in a way that will generally increase the probability of avoiding a collision. However, those mechanisms are not perfect and in some situations the behavioral change of the agent might actually cause it to hit the obstacle even harder than without any change. Because behavior is the result of the interaction between the agent and the environment (Beer, 1995), no observer would speak of the resulting behavior as obstacle avoidance if the agent actually makes the impact worse.

We have classified the agent strategies in four basic kinds, based on the three behavioral patterns identified above (see Tab. 1). Overall, agents of the same class share essential behavioral tendencies, even though they vary in the details of their implementation. Figure 4 shows the morphological properties of the four agent classes, categorized by behavior patterns, and Fig. 5 shows differences between agent categories on the population level. It is interesting (though not surprising) to note that even though the categorization was done solely on the basis of behavioral observations it is nearly perfectly reflected in the body properties of the agents. As could be expected it turns out that if both body and controllers are evolved as a functional unit one cannot discuss one without the other. The evolutionary dynamics shape the complete agent and adapt it to a certain survival strategy.

Drifters exhibit relatively fast base movement using their (usually) single functional locomotive actuator. With a single actuator an agent cannot change its direction, it can only modulate its speed. Consequently, drifters can neither avoid obstacles nor can they actively approach an energy source. Instead, they modify their speed in the presence of an energy source. This is achieved either by 'monitoring' their life energy supply and stop moving if it exceeds a certain threshold, or by using energy sensors to measure the energy concentration of the environment. Whenever the energy concentration is high they slow down or come to a complete stop. Drifters are typically very small and light-weight (see Fig. 4). Their life span is comparatively short but their population size is larger than that of all other types (see Fig. 5). Drifters usually only have one sensor, one (functional) actuator and minimal networks to control their extremely simple behavior. In many simulation runs, the first sustained population consists of drifter-like agents. Sometimes they evolve into other types, but often a relatively stable drifter population establishes itself where only the morphological properties are further refined to suit this strategy. It is worth noting that even this simple strategy requires a fair amount of adaptation to first acquire and then 'calibrate' the required sensory and actuation system. No viable strategy emerged where output was constant (e.g. comparable to 'always go forward and kill' reported in (Channon and Damper, 1998)).

Foragers have base movement, change their behavior in the presence of an energy source, but do not avoid obstacles. In their simplest form, a single energy sensor and one actuator placed roughly opposite the one responsible for base movement are sufficient to perform successful approach behavior. The translational component of the base actuator is counteracted by a usually slightly tilted second actuator. This results in an inward spiraling movement dependent on the strength of the sensory stimulus. However, the experiments show that usually two energy sensors and a larger number of actuators are used to implement this behavior. Also the actual behavior resulting from the agents' actions and its robustness vary from population to population and over evolutionary time. Some agents will always manage to approach an energy source within their sensor range while others may only succeed if they are approaching from a particular side. Another difference is how well an agent is able to keep contact once it has approached the energy source. While some agents spend most of the time ineffectively circling around an energy source, others can perfectly center themselves over them and remain there until the source is either fully consumed or disappears.

While drifters usually minimize their body size to the lower bound of 0.1, foragers almost consistently have a size of about 0.3 (see Fig. 4). Some foragers also increase their solidness instead of their size. Both adaptations lead to higher capacity but also higher movement costs. Foragers have to find energy more often than drifters but also consume energy sources more efficiently.

Avoiders follow a somewhat surprising strategy. They are the only agents that completely abandon energy perception through external sensors. Avoiders exhibit base movement and obstacle avoidance. The different populations responded to contact with energy sources in different ways. In all cases the resulting behavior can be explained by the internal sensor for the agent's life energy level. In the first case

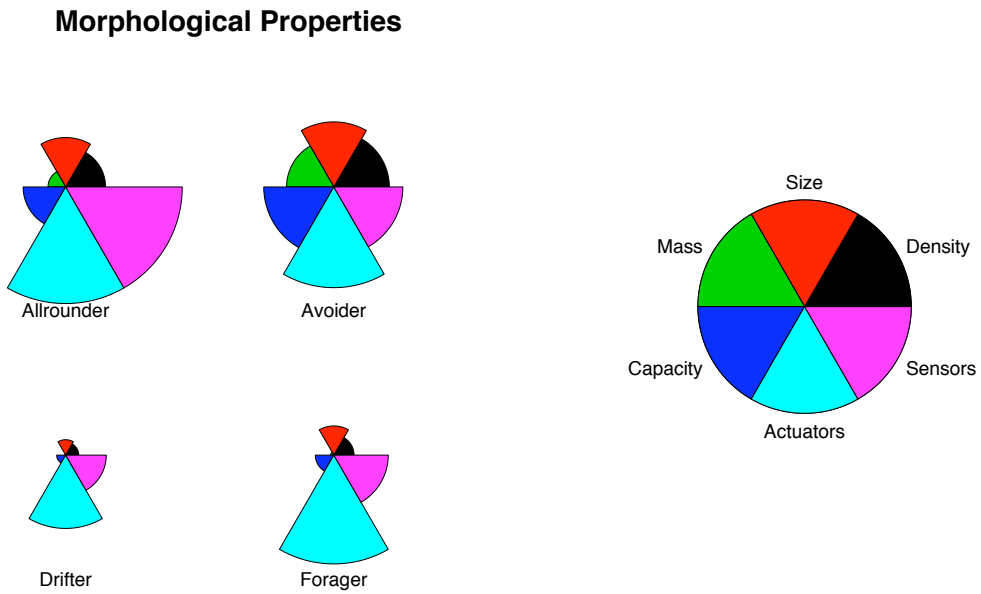


Figure 4: Morphological properties (all normalized to 1) of evolved agents (PG 500) categorized by behavior patterns.

the base actuator of the agent is inhibited once its life energy level exceeds a certain threshold and the agent stops on top of the energy source. In the second case the same trigger activates the actuator used for obstacle avoidance causing the agent to start moving on a perfectly circular trajectory. In both cases the agent (at least partly) consumes the energy source without directly sensing its presence. The respective behavior patterns persist even if the energy source disappears until the life energy level drops below the triggering threshold. Avoiders have slower base movement than other agents. This seems to be an adaptation to their increased weight and their consumption strategy as there is a considerable delay between first contact with the energy source and the life energy reaching the needed threshold to trigger the agent's response. The observed avoiders are bigger than foragers and have a higher solidness. The increased solidness gives them a much larger capacity at a medium risk because of their obstacle avoidance capabilities.

Allrounders are agents which exhibit all three behavior patterns. Basically they are the same as foragers with the added ability to avoid obstacles. Their foraging behavior is the same and they can sometimes evolve from forager agents. However, they tend to have a higher capacity than basic foragers. Most of the evolved allrounders achieve this by increasing the solidness value. As with avoiders the risk of increasing the solidness is lowered by the ability to avoid obstacles. Allrounders (as can be expected) have the most complex networks and the most sensors and actuators. They also have the smallest population sizes.

General Properties To show that behavioral diversity emerges even in simple and uniform environments we have only presented four survival strategies. However, it is worth noting that changing only the concentration of obstacles and energy sources can lead to completely different behavioral strategies. We will mention one observed type of agent because of their radically different approach. This strategy appeared in environments with high concentration of both obstacles and food sources. A high concentration of obstacles 'penalizes' movement early in evolution when agents are not yet well adapted (by either being light-weight or by avoiding the obstacles). There, agents can be nearly or completely sessile. These agents exhibit no base movement at all. They remain stationary until an energy source appears within their sensor range. Once in range, they quickly approach the energy source, center themselves over it and remain there. These agents have much larger bodies and simpler controller networks than mobile agents. Larger size consumes a lot of energy when moving but it also increases the maximum energy capacity of the agent. A larger agent which does not move can survive longer without consuming energy.

More generally, however, selection seems to favour small and light-weight agents that exhibit some base movement early in the evolution. This is further optimized if agents follow the drifter strategy. Agents with an active foraging strategy (foragers and allrounders) are slightly larger and agents

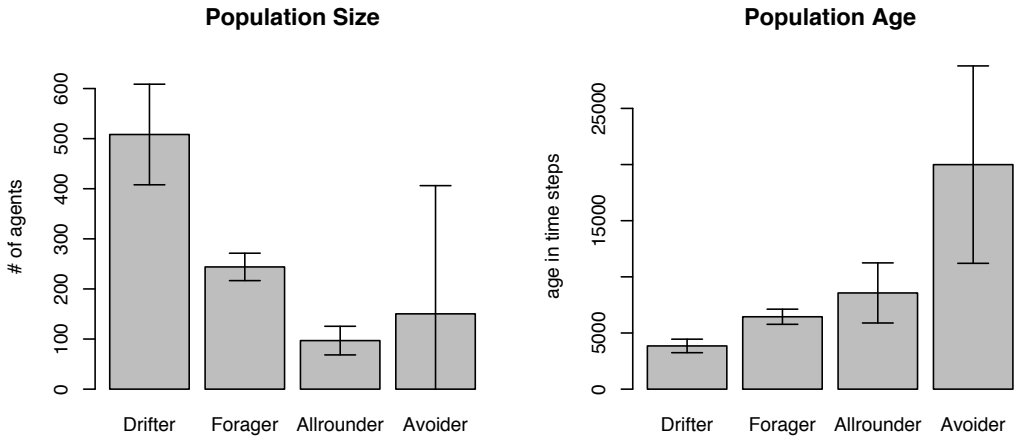


Figure 5: Average population size (left) and average population age (right) of evolved agent populations categorized by behavioral strategies. Error bars show the 95% confidence interval. The large bars for avoider populations are due to the small sample size (3).

with slow base movement are even larger still to increase their energy capacity. All agents without collision avoidance minimize solidness. Agents with collision avoidance often increase solidness and size to increase their energy capacity. Sensors are effectively restricted to the required minimum while actuators seem to accumulate even if they are not used efficiently or not at all (see Fig. 4).

Reproductive strategies are very hard to analyze in detail as they can only be understood by analyzing the dynamics of each agent's network. Supporting the rationale behind our reproductive criterion which gives the agents control over *when* they invest in offspring, constant reproductive activity (irrespective of internal and external circumstances) did not emerge as a viable strategy in a single sustained population. However, most agents follow simple reproductive strategies or combinations thereof; these can roughly be summarised as follows: Invest in reproduction if energy is present, otherwise don't. There are different ways to achieve this. The most commonly used is a positive correlation between the energy sensors and the actuator for the reproductive depot. Alternatively, the activation of the reproductive depot is positively correlated to either the internal energy level or the activation of a locomotive actuator used for foraging. Many agents use a combination of these strategies. Additionally, often a negative correlation between a solidness sensor (or an actuator used for collision avoidance) and the reproductive activity exists.

Discussion & Conclusion

We have shown data of four distinct behavioral strategies evolved in a virtual ecosystem. The different types of agents evolved 'high level' behaviors (foraging, obstacle avoidance) without a discrete set of predefined behavior primitives and without other pre-defined functionality or structure. All behavior is the result of the agents interacting with the environment via a very simple but versatile locomotion model. The evolution was done in an artificial ecosystem by natural selection and both neurocontrollers as well as morphology (size, solidness, sensory and actuation structure) of the agents were freely evolved. Based on the results of these more general experiments we are satisfied that this approach is very capable of evolving diverse behavior while further reducing the need preconceive necessary action possibilities the agents might need to survive under different environmental conditions. To keep evolved strategies comparable we have only used a small and homogeneous environment in this experiment. A possible extension of the presented experiment is to investigate the impact of more variable environments on the evolution of survival strategies.

While we think that replacing discrete behavior primitives by our simpler actuation model in combination with the proposed reproduction criterion is more conducive to the evolution of diverse behavior, it is also clear that such a reduction of the set of predefined biases is not possible or even desirable *ad infinitum*. Apart from obvious computational complexity considerations the actual goal of the simulation has to be considered. We tried to create an evolutionary setting which is flexible enough to allow the evolution of distinctly diverse and non-trivial agent strategies. In other situations a different set of biases might be appropriate. One main message of this paper is that, also when using natural selection in an ecosystem scenario, one has to be aware what biases

are built into the system and how they affect the simulation. One avenue of future research will consist of a comparative study about how different reproductive criteria influence the evolved diversity of agent strategies.

Another future aim of this project is to investigate the potential emergence of phenomena comparable to basic affect in natural organisms. Basic affect in this context includes individualistic affect like approach-avoidance, arousal and agonistic affect, as well as prosocial affect like cooperation. These phenomena are thought to be the physiological bases for higher level affect (as e.g. described in (Buck, 1999)).

We are currently extending our ecosystem model to include the possibility to evolve simple neuromodulatory mechanisms which are used in animals to support affect. Similar to neuromodulation these mechanisms would allow the neurocontrollers to regulate whole groups of neurons as opposed to the direct synaptic transmission in standard neural networks. Therefore, in the next set of experiments we will investigate if providing this possibility will lead to the evolution of agents that exhibit properties normally ascribed to such basic affect. Targeted results of these experiments include changes in foraging behaviour depending on the life energy level (arousal) or flexible weighting in approach-avoidance conflicts (e.g. approaching energy source close to an obstacle only in certain situations). If such mechanisms are successfully evolved we expect agents to develop more flexible behavior strategies which are also more robust to changes in the environment. We also hope to be able to draw some conclusions about the necessary conditions and origins of functionally similar processes in real organisms.

Acknowledgment

Peter-Paul Pichler is funded by a research scholarship of the University of Hertfordshire.

References

Bedau, M. A., Norman, A., and Packard, H. (1992). Measurement of evolutionary activity, teleology, and life. In Langton, C., Taylor, C., Farmer, D., and Rasmussen, S., editors, *Artificial Life II*, pages 431–461, Redwood City, CA. Addison-Wesley.

Beer, R. D. (1995). A dynamical systems perspective on agent-environment interaction. *Artif. Intell.*, 72(1-2):173–215.

Buck, R. (1999). The biological affects: A typology. *Psychological Review*, 106(2):301–336.

Channon, A. D. and Damper, R. I. (1998). Evolving novel behaviors via natural selection. In Adami, C., Belew, R. K., Kitano, H., and Taylor, C., editors, *Proceedings of the 6th International Conference on Artificial Life*, pages 27–29, Cambridge, MA, USA. MIT Press.

Holland, J. H. (1990). Echo: Explorations of evolution in a minature world. In Langton, C. G., Farmer, J. D., and Doyne, J., editors, *Second Conference on Artificial Life*. Addison-Wesley.

Kodjabachian, J. and Meyer, J. (1998). Evolution and development of neural networks controlling locomotion, gradient-following, and obstacleavoidance in artificial insects. *IEEE Transactions on Neural Networks*, 9 (5):796–812.

Menczer, F. and Belew, R. K. (1996a). From complex environments to complex behaviors. In *Adaptive Behavior 4*, pages 317–363.

Menczer, F. and Belew, R. K. (1996b). Latent energy environments. In Belew, R. K. and Mitchell, M., editors, *Adaptive Individuals in Evolving Populations: Models and Algorithms*, pages 191–208, Reading, MA. Addison Wesley.

Pichler, P.-P. and Cañamero, L. (2007). An evolving ecosystems approach to generating complex agent behaviour. In *Proceedings of the First IEEE Symposium on Artificial Life*.

Polani, D., Nehaniv, C. L., Martinetz, T., and Kim, J. T. (2006). Relevant information in optimized persistence vs. progeny strategies. In M.Rocha, L., Bedau, M., Floreano, D., Goldstone, R., Vespignani, A., and Yaeger, L., editors, *Proc. Artificial Life X*.

Seys, C. W. and Beer, R. D. (2006). Effect of encoding on the evolvability of an embodied neural network. In *GECCO 2006 Workshop Proceedings, Workshop on Complexity through Development and Self-Organizing Representations (CODESOAR)*.

Stanley, K. O. and Miikkulainen, R. (2002). Efficient evolution of neural network topologies. In *Proceedings of the 2002 Congress on Evolutionary Computation (CEC '02)*.

Stanley, K. O. and Miikkulainen, R. (2004). Competitive coevolution through evolutionary complexification. *J. Artif. Intell. Res. (JAIR)*, 21:63–100.

Yaeger, L. (1994). Computational genetics, physiology, metabolism, neural systems, learning, vision and behaviour or polyworld: Life in a new context. In Langton, C. G., editor, *Artificial Life III*, volume XVII, pages 263–298. Addison-Wesley.

Yaeger, L. S. and Sporns, O. (2006). Evolution of neural structure and complexity in a computational ecology. In Rocha, L. e. a., editor, *Artificial Life X*, Cambridge, MA. MIT Press.

Complex Systems Models: Engineering Simulations

Fiona A. C. Polack[1], Tim Hoverd[1], Adam T. Sampson[2], Susan Stepney[1], and Jon Timmis[1,3]

[1]Department of Computer Science, University of York, UK, YO10 5DD
[2] Computing Laboratory, University of Kent, Canterbury, UK, CT2 7NF
[3]Department of Electronics, University of York, UK, YO10 5DD
Fiona.Polack@cs.york.ac.uk

Abstract

As part of research towards the CoSMoS unified infrastructure for modelling and simulating complex systems, we review uses of definitional and descriptive models in natural science and computing, and existing integrated platforms. From these, we identify requirements for engineering models of complex systems, and consider how some of the requirements could be met, using state-of-the-art model management and a mobile, process-oriented computing paradigm.

Introduction

In computing contexts, and particularly the context of artificial life, complex systems are studied through computer simulation. Reynolds' boids (Reynolds, 1987) is a classic example, where the complex flocking or swarming behaviours are shown by visualisation of a large number of simple boid processes obeying simple rules.

Simulations are used to model complex systems – biological phenomena, economies, human societies, and much more. Typically, a simulation is built to explore a specific problem in a specific context; there is little attempt to develop generic solutions, or to record any design or engineering. Often a valid simulation is judged to be a model that produces the expected results by a process that looks a bit like reality; there is little concern for the quality of the underlying simulation (Epstein, 1999). General support for complex systems and agent modelling tends to be at the implementation level (see, for instance, the ACE resources, www.econ.iastate.edu/tesfatsi/ace.htm). An immediate result of this focus is a long-running intellectual debate about whether it is possible to do science through simulation (see Miller (1995); Paolo et al. (2000); Wheeler et al. (2002); Bryden and Noble (2006)). Similar issues with the validity of simulation evidence arise in safety engineering and other dependability, assurance (Alexander, 2007). For ALife, it has already been noted (e.g. by S. Bullock, in Wheeler et al. (2002)) that to assess the role and value of complex systems simulation, we need to address deep questions of comparability: we need a record of experience, of how good solutions are designed, of how to chose parameters and calibrate agents, and, above all, how to validate a complex system simulation.

The sorts of systems in which we are interested are complex in the sense of having elaborate behaviour at a high level that is the consequence of many simple behaviours at a lower level. The high-level behaviour cannot be deduced as a simple combination of low-level behaviours – in the same way that the velocity of a flock of birds is not derivable by any simple analysis of the behaviours of the individual birds. Space, time and the environmental context are critical features of these systems. Engineering of such complex systems requires support for the software engineering of computer simulations, for use in the investigation of complex systems in nature, *in vitro* and *in silico*.

Ultimately, our goal is to engineer simulations of systems exhibiting several layers of emergence – the lowest level gives rise to emergent behaviours at an intermediate level, and the ensemble of these behaviours gives rise to further behaviours at still higher levels (see Turner et al. (2007); Stepney et al. (2006); Polack et al. (2005)). We see this as an essential feature of initiatives such as molecular nanotechnology that aim to engineer interventions in natural and complicated systems through management of emergent properties; our work is also relevant to macro-scale complex systems – often referred to as systems of systems – such as human organisational systems, traffic management.

This paper reports an initial investigation into the state of the art in complex system modelling and software engineering, that leads to a consideration of how existing approaches and techniques can be adapted and used for engineering simulations of complex systems. We consider some interdisciplinary approaches to modelling and simulating complex systems that adopt software engineering models and tools to describe complex natural systems. We identify advantages of these models, but also their failure to adequately express and manage emergent properties.

The state of the art in software engineering of simulations for systems biology comprises a number of interdisciplinary projects that integrate modelling tools and visualisation facilities, to construct specific, flexible platforms for experi-

mentation. The review shows that these projects have many of the expressive features needed for a platform, but that they do not necessarily generalise in the ways that we wish. As a first step towards a general simulation platform, we consider how advances in software engineering might help.

Models of Complex Emergent Simulation

A *model* is an abstraction that is made to aid understanding or description of something. We can distinguish two orthogonal modelling goals: description and definition. For a complex emergent system, a descriptive model might capture aspects of the observed high-level behaviour; in modelling natural systems, scientists use models to capture what they observe. A definitional model is more typical of conventional engineering – it expresses required characteristics of a system at an appropriate level of abstraction. A definitional model can be refined, translated and analysed, to improve understanding of system characteristics, and, in engineering, to support construction of an artificial system.

Here, we consider some existing approaches, divided into mathematical models and diagrams. Both include models that are descriptive and models that are definitional. We then consider existing tool support for these approaches. Finally, we look at two state-of-the-art approaches that combine existing modelling and tool support. Our aim is to postulate requirements for engineering simulations of complex systems, through identification of good practice in explanatory and exploratory simulation of complex systems.

Mathematical Models

In science, mathematical models are essentially descriptive, attempting to replicate observed aspects of natural structure or behaviour. Physics and biology often use differential equations to approximate the observed behaviour of a high-level system, based on continuous variables at a lower level. For example, the Lotka-Volterra differential equations are important for modelling predator-prey systems. Stochastic models (for instance, Monte Carlo simulations) also aim to capture the high-level behaviour of complex systems.

The scientific use of mathematical models is instructive; the models allow scientists to explore variables that might contribute to observed behaviour. Once candidate variables are selected, the hypothetical result of changing the values or relative importance of variables can be studied. The best mathematical models provide convincing evidence that the modelled variables do indeed influence the real behaviour. These models also provide benchmark results: a simulation that produces realistic observable behaviour should also produce data. Mathematical models could form a basis for evaluating simulation-derived data against real-world data.

However, in the context of complex systems engineering, there are several limitations to the scientific use of mathematical models. The models rely on already having identified the key system components; furthermore, there must be an objective, typically discretised, representation of those components. In the real world, emergent behaviour does not arise through solving differential equations; these models are analogues, but do not provide significant insight into the continuous internal process of a complex system. Furthermore, the scientific models are not definitional – they do not directly admit engineering refinement or analysis.

In software engineering, definitional mathematical models use discrete mathematical concepts, from set theory, predicate logic, etc. Models are formalisations of programming concepts such as the Hoare logics (Hoare, 1969) and Dijkstra's predicate transformers (Dijkstra, 1975). Referred to as formal specifications, the models capture the structure, behaviour and/or communication protocols of systems, and provide the basis for various analyses of correctness.

Diagrams

Historically, biological illustration uses bespoke, informal sketches to express observed relationships or interactions, without any systematic notational definition. More recently, modelling techniques have been adopted from other disciplines – systems biologists, and their interdisciplinary collaborators, are turning to existing diagrammatic notations with defined syntax (and sometimes defined semantics). The use of diagrams is still largely descriptive, even though the notations originate in the definitional context of software engineering. Three classes of diagram can be distinguished.

Connectivity diagrams express the known connectivity of natural systems, using analogies to electrical circuits or software components. Examples include circuit diagrams, interaction diagrams, and various message sequence charts. Connectivity diagrams map well to mathematical languages – process algebras have been used to model many aspects of cells (and other biological systems) and to formally express and analyse communication protocols.

Structural, or class, diagrams describe static components and their relationships. The current fashion is to use class diagrams, where a class is an intensional definition of some local data (variables, constants) and the behaviours needed to maintain that data. The extension of a class is an object, that holds specific instances of data. The associations of a class determine how objects of various classes can interact; associations can be thought of as providing the potential for connectivity, whilst class behaviours include those needed to establish and maintain connectivity among objects.

An important, and biologically attractive, aspect of class diagrams is that the classes and associations represent families of conformant instances (objects and links, respectively). Thus, a class *cell* represents arbitrarily many similar instances of the cell. Scientists sometimes prefer to capture the structure of specific scenarios, using object diagrams – an object is an instance of a class. In this context, snapshot diagrams can also be used, to express the structural effects of the execution of methods or operations on objects.

A problem with structural diagrams for complex system modelling is that there is no sense of the system as an entity – the system view is a collection of type descriptions. We can constrain the number of objects that are linked to each object of another class, but we cannot easily define how many objects exist (relatively or absolutely; the number of objects may be highly dynamic). Furthermore, we can define methods to create and destroy instances, but these can only be constrained by static predicates, not by system-wide observation or dynamic preconditions (how many are needed, or how many can be supported by the current environment).

State machines are essentially variants on (finite) state automata. They express the possible evolutions, either of an object or of a system as a whole. Object-level diagrams have the advantage of simplicity – interaction is indicated by shared events or generation of events to other state diagrams. There are many notations and variants, including Petri nets, Harel state charts, UML state diagrams.

A state machine defines, firstly, the different states of existence of an object (or system). In current realisations of state machines, a state is distinguished by the applicable range of values of its variables (if a state machine relates to objects of an object-oriented class, states are defined over attribute values). Next, the state machine defines the ways in which an object can change state, via transitions. A transition is a response to an event, and an event is, typically, an input received by the object (or system). Transitions are protected by guards – a set of conditions, concerning the wider system state (and perhaps the environmental context of the system) that must be true if the state is to change. Semantically, a state machine may require the state to change whenever an event is received and the guards are true, or, less-commonly, it may simply permit the state change.

An advantage of state machines for biological systems is that they can express known stimuli and responses. Most state machine notations admit concurrent states, which, with the ability to capture incoming and outgoing events, make it possible to construct sophisticated models of, for instance, cell interaction. The diagrams express potentially-dynamic structures, and can provide drivers for simulation of collections of objects. However, the same limitation arises as on class diagrams: the number of objects that are operational at any time cannot be defined in the models.

Tools for Models

In computer systems engineering, and in scientific description, modelling is increasingly tool-driven: use of models generally means use of modelling tools. In computer science, tools support formal specification (definitional mathematical modelling), providing type-checking and proof assistance. Proof can be applied to conjectures about a model and about the relationships between models (refinements, retrenchments, reifications). In natural sciences, descriptive mathematical models are equations that simulate behaviour; tools include statistical techniques to assist in identification of variables (used in deriving equations) and in analysis of results. Tools to solve equations (heuristic or absolute) are also common. In both contexts, tools usually support a single language, and generally require some expertise.

Tools for diagrammatic modelling tend to be commercially-driven. Usability, in high-productivity commercial contexts, takes precedence over strict conformance to standards and accuracy. Like mathematical tools, diagramming tools usually support one notation, which is often a proprietary variant of a public, *de facto* or industry standard, with at best limited documentation of less-standard features. (Note that the widely-used UML is one standardised notation that supports many views of a system (http://www.omg.org/spec/UML/2.1.2/).) Traditional tools for diagrammatic modelling support concrete syntax, and may impose some well-formedness conditions. The ability to check well-formedness has improved significantly in recent tools aligned to management of models; the ability to refine and analyse diagrammatic models is also improving. We return to this aspect of tool support later.

A Brief Review of the State of the Art

Rather than attempting a review of complex systems modelling in general, we consider two state-of-the-art approaches, noting their strengths and limitations.

Perhaps the most advanced computer contribution to the simulation of real biological systems is currently found in Reactive Animation (RA) (Efroni et al., 2007; Sadot et al., 2007), an approach that combines off-the-shelf tools into a sophisticated and flexible simulation environment. The key modelling components are Rhapsody statecharts (state machines) and Live Sequence Charts (connectivity diagrams). The authors describe their work as reverse-engineering biological systems into protocols and object-evolution models. Experimentally-derived (real) biological data is used to populate the initial state of a simulation. Among the facilities for interacting with and manipulating the simulation are adjustable biological-scale time, and zoom-in and tracking facilities. It is also possible to adjust the underlying models and see the effects directly on the simulation.

A key aspect of RA is its modularity: the modelling tools are separate, integrated through the *InterPlay* application, and manipulated through a *PlayEngine*. Similarly, the systems that are modelled can be composed in a modular way. Clever integration means that modification to simulations can either be initiated through the interface and reflected in models, or initiated in models and reflected in the interface.

RA comes from an interdisciplinary team, with leading researchers from several communities bringing their complementary skills and problems. Although the integration is modular and thus flexible, the current work is closely tied to proprietary modelling tools. RA is an existence proof that integrated, flexible simulation and modelling is possi-

ble, rather than a general solution to modelling and simulation of complex systems. Also, the motivation for the work is to model a complete organism; our more general motivation is to support the engineering of complex systems. Knowing how to replicate the behaviour of a complex system, and being able to extend our knowledge, are critically important, but are only part of this wider motivation.

The second example of state-of-the-art modelling of complex systems comes from the process algebra community. PEPA (Calder et al., 2006, 2008) uses stochastic process algebra to construct complementary models of a biological network – a reagent view (perhaps akin to the state machine models) and a network view (akin to the connectivity models). The reagent view can express concentrations and triggers to biochemical product formation, whilst the network view captures time-ordered sequences of events across the system. Whilst diagrammatic views are supported, the PEPA modelling is strictly mathematical; the views use the same mathematical language, and have been proved isomorphic. The formalism supports proof of properties – proof of deadlock-freedom, for instance, improves the confidence of the modellers in their networks, since nature does not normally exhibit the forms of deadlock that we observe in communicating (computational) systems.

Like RA, PEPA was developed in a well-integrated interdisciplinary context, to help researchers understand the biological networks that they could observe and measure in the laboratory. The PEPA workbench (http://www.dcs.ed.ac.uk/pepa/tools/), which supports property expression and proof, also supports an algorithmic approach to generating conventional ordinary differential equations from the PEPA models, which allows a clear comparison of observed behaviour of the system represented by the models with results of laboratory analysis.

PEPA (and other process algebra approaches such as bioAmbients (Regev et al., 2004)) demonstrates the benefit of deep integration. The models are different representations in the same notation, with a common semantics. The approaches work well in closely-coupled interdisciplinary contexts, where experts in process algebra work alongside laboratory scientists. However, experts in process algebra are not particularly common, even in Computer Science. Like the proprietary-tool buy-in of RA, PEPA is an existence proof for simulation environments, rather than a general solution to modelling complex systems that would be amenable to use by research groups and system engineers without specific expertise.

Requirements for Complex Emergent Systems Design

Whilst the component models of RA and PEPA are definitional, the goal of these simulation initiatives, like much complex systems modelling, is descriptive, motivated by a need to express observations of real systems, in order to explain or explore natural processes. In seeking models for engineering complex systems simulations, we need definitional models that are amenable to use by interdisciplinary researchers. We need to be able to relate definitional models (functional requirements and their realisations) to descriptive models that identify what the high-level system should achieve (the emergent behaviours that we want). We start by considering what desirable aspects of complex systems are expressible in the reviewed forms of model. We then consider other requirements and how they might be met.

Existing modelling approaches can express (and define) features such as:

- known structures within and among components – using mathematical relations, or structure diagrams;
- protocols for communication among components – using process algebras or diagrammatic models of interaction;
- potential state changes – using state machines and other variants of state automata.

Each form of model presents a limited view, or a single aspect, of the system. Most of the models are static – they either capture the state of a system or they prescribe possible histories of a system. None is really explanatory, in the sense of providing understanding of the layered processes that determine a particular complex system. For engineering complex systems, we need models that,

- express the characteristics of multiple instances of low-level systems, as well as the required emergent characteristics of high level systems;
- represent the context (in terms of space, time and relevant environmental features) of systems;
- capture the cumulative make-up of systems (quantities of objects etc.).

Where models of natural systems are used as a basis for simulation, it is sometimes the case that, rather than model knowledge about a natural system directly, the diagrams express a software engineering design or aspects of the implementation – natural concepts are modelled with computing-related attributes (`name: string`) and operations (e.g. `print()`). Both natural and design models are necessary, but there is a need to be explicit about the purpose of a model, and there is a need to understand and express correspondence between the two sorts of model.

In addition to general engineering needs, we can divide other requirements into: features of complex systems that are not met by existing approaches to modelling; and desirable features of models.

System Features Not Covered

The key omissions, for accurately capturing the range of views of a complex system, can be summarised as dimensionality and scale.

Dimensionality must be considered, since a complex emergent system is, by definition, concerned with time – the emergent properties emerge when the system runs for a period of time. In most cases, a complex emergent system is also concerned with space, since the separation of components fundamentally affects their ability to interact.

Scale can mean two things. Firstly, the relative or absolute quantities of components in a system can affect whether emergence occurs (what the system actually achieves). As noted above, diagrams of the system state do not have an obvious way to define the quantities of objects created, or to define dynamic constraints on behaviour (other than through transition guards).

The second meaning of scale concerns the scale of observation. This is what determines the subject of models: the emergent system or the system components. Conventional engineering models operate at one observation scale, so the diagrams (even in combination) cannot be used to explore inter-level effects such as emergence or self-organisation.

Scale of interaction is critical. Complex behaviour arises when many (hundreds, thousands, or even billions) instances interact. The models typically used in systems biology, and in their conventional electronics and computer science origins, express constraints on interaction, but cannot express the cumulative interaction that is the root of emergence.

Furthermore, the emergent characteristics of a complex system are typically the result of interaction across scales of observation – low-level components induce local effects on their environment; higher-level components monitor their environment and thus react to changes, once the cumulative local effects are detectable at the higher level.

Desirable Features of Models

We can identify a number of desirable features for engineering models of complex systems; these features are often apparent in the modelling forms that already exist.

Modifiability is essential if models are going to be effective tools in engineering or scientific research. It is highly desirable that modifications in one view or instance of a model are reflected (automatically) in other views. In all areas, modification is used to adjust models to meet some external criteria (e.g. realism, customer requirements, etc.) In engineering, modification also means *translation* from an abstract model to an implementation level, or between notations that are in some sense equivalent. The problems of consistency under modification have challenged designers for many years, and are exacerbated by inconsistent or unintegrated modelling tools and ill-defined notations.

Understandability has many aspects, but in complex systems it tends to rely heavily on visualisation – photographs, sketches, diagrams, mathematical formulae, simulations. However, the understandability of visualisations depends on the ability of the reader to interpret the visual forms in the ways intended by the author of the model. We need ways to express and encourage shared interpretations.

There have been many (very many) works defining the meaning of model elements, and semantics is widely studied. For static models, it is useful to distinguish the specific language of the model (the visualisation – the shapes of components, location of labels etc), from the abstract concepts (the underlying model) – hence concrete and abstract syntax. Furthermore, it is useful to provide a definition of the meaning of the concept (semantics), for instance by reference to a well-understood or better-defined concept (so, mathematical sets can represent the semantics of the data aspect of a class of objects). However, an area that is less well studied is behavioural semantics. We do not have well-defined ways to develop models, or well-defined ways to interpret what static models tell us about the temporal and spatial behaviour of the systems that are modelled.

Towards Meeting the Requirements

Reviews of existing modelling approaches and their possible contributions to the engineering of complex system simulations demonstrate some ways in which the requirements can start to be addressed. Indeed, state-of-the-art modelling of natural systems has already provided bespoke solutions in restricted contexts. We first consider ways in which the required system features might be addressed in engineering methods. Then, to address requirements for features of models, we outline ways in which model integration and model management might be used to support the integrated tool platforms needed to engineer complex system simulations.

Requirements for Coverage of System Features: Exploring Simulation Environments

As demonstrated in the RA approach, above, time and space are inherent to simulated models. The RA simulations are derived by a specific form of execution of the static object and state machine models on multiple (diverse) instances simultaneously. The visual simulation shows the emergent effects across time and space. In modelling terms, the static models represent potential point-in-time observations of single objects in a collection. The simulation is then a simultaneous running of many possible paths through the static models.

A simulation environment for engineering complex systems needs to be able to relate simulations and static models in ways that are not commonly attempted. Simulation environments need to be able to constrain the simulation to follow the static models, but to free the simulation from biases – accidental constraints imposed by over-eager modelling or by the simulation environment itself. A common form of over-constraint is the use of absolute spatial co-ordinates –

in natural complex systems, components have only local reference, to their nearest neighbours; there is no component-level view of the whole system. If a simulation locates components by absolute co-ordinates, locality becomes a derived attribute, not an inherent concept. In general, such simulations are inadequate because emergence due to local interaction is masked by unnatural global effects. Furthermore, simulations using absolute spatial co-ordinates pose engineering problems: they are hard to distribute and hard to extend dynamically, because the spatial algorithms are hard-coded.

When simulating complex systems, it is difficult to avoid bias in the execution. This can be illustrated at various levels. Simulated systems of differential equations often display some realistic-looking behaviour. However, the behaviour is significantly biased by the way that the equation is constructed (the formula selected) and the variables and constants chosen. Next, spatial emergence can be shown in simulation, but the form of spatial emergence is typically biased by the form of underlying representation. Consider systems such as "game of life" cellular automata, where the representation is usually shading of cells in a regular, two-dimensional grid: changing the grid or the shading can make a significant difference in the perceived emergent behaviours. Finally, there are biases related to metrics – the number of instances, the time step, the spatial granularity, even the number of time-steps over which a characteristic is measured.

A key question in the use of simulation for engineering complex systems (or for understanding naturally-occurring complex systems) is – how far down must we go to avoid bias? In most cases, we do not have to go as far as a simulation of quantum mechanics for a meaningful biological simulation, but it is a sobering thought that even macro-scale complex systems (such as a galaxy, or at a more tangible scale, traffic flows or distributed command and control systems) are subject to the fundamental laws of physics.

From this discussion, two things arise. The more obvious is the need to integrate static modelling with simulation. Perhaps less obvious is the need for high-performance, flexible simulation environments, which support local reference (to avoid building in biases such as unrealistic long-range communication structures), and can handle appropriate numbers of interacting instances.

It is arguable that an interaction model, to statically express the ways in which instances interact, is missing from the existing approaches. Formal approaches (based on communicating process calculi) and associated informal visualisations (based on models of the connectivity diagram type) can be extended to express this aspect. We need to understand how to integrate on this scale.

Requirements for Features of Models: Managing Models

Recent advances in the theory of and tool support for model management are predicated on the use of modelling tools. The issues that arise in management of mathematical and diagrammatic models relate both to the conceptual basis of the models, and to the ways in which models are used.

In computing contexts, model integration and model comparison are becoming important – often driven by commercial imperatives when organisations (tool developers or client users) merge, but also motivated by academic interest in patterns and commonalities, and by the simple availability of the computational resources to manage collections of models. The solutions being explored are not one-off integrations (in the sense exemplified by RA and PEPA), but generic bases for integration and model management.

Where groups of models are used to express features of a system, the interaction of models is often overlooked, even by professional designers of modelling languages. The different views of a system overlap, and the overlap needs to be consistent. A classical case is where a class diagram and state machine are used together – the states in the state machine should be expressed in terms of attributes shown in the class diagram, and the transitions should be effected by invoking operations of the objects of a class, in accordance with the association structures defined in the class diagram. Similarly, where connectivity diagrams are used, it is important that the internal methods and links are consistent with those in the class diagram, and the sequences of communication are consistent with those permitted in the corresponding state machines. Another classical angle from Computer Science is that, where formal models exist alongside diagrammatic models, there is an obligation to demonstrate the continuing equivalence of concepts – there is a need to define correspondences, or traceability links, between different model concepts.

Unless we understand, and can characterise, the semantics of the engineering models that we adopt, we cannot adequately address adaptation of models to the needs of complex systems engineering – in simple terms, we need to understand the classical concepts of state, transition and class, in order to find sensible ways of accommodating space, time and environment.

Recently, software engineering has considerably advanced the management of models; two movements are establishing fundamentally-comparable definitions of modelling languages. The *unified theory of programming* movement (see Hoare and He (1998); Woodcock (2003); Cavalcanti and Woodcock (2006)) is seeking common mathematical underpinnings for modelling and programming languages. Separately, commercially-driven research into model-driven development (Swithinbank et al., 2005) led by organisations such as the Object Management Group (http://www.omg.org/), and large inter-

national projects such as the EU Modelplex initiative (https://www.modelplex.org/), focus on defining common abstract concepts for modelling languages, and domain-specific capabilities, through the use of *metamodels*. The associated model management concepts have the ability to support formal as well as diagrammatic languages.

Both initiatives open new ways to compare, validate, extend, and transform models. Here, we focus on model-driven development (MDD), because it is more accessible to developers.

Model-driven development seeks to manage the consequences of distributed development: developers in large projects typically produce different variants of many types of models that must be shown to express equivalent concepts; furthermore, having produced integrated design models, implementation can be made faster and less error-prone if repetitive programming is automated from design models.

The fundamental concept of MDD is the model stack: a particular realisation (M_0 level) is an instance of a model (M_1 level), whilst that model is an instance, or realisation, of a metamodel (M_2 level). The current top of the stack defines languages for metamodelling (M_3 level). The four layers are not always clear-cut, but they provide a reasonable separation of concerns and a sufficient basis for model management. MDD tool suites such as Epsilon (http://www.eclipse.org/gmt/epsilon/, Kolovos et al. (2006)) support expression of models and metamodels, checking of models against metamodels (for syntactically-correct use of notations), and checking of consistency across models (that all models in a family use concepts in the same way, at both the notational and application levels). These tools support model transformation, with mappings between concepts in different metamodels being applied to transform models, either between notations, or from design models to implementations.

Predominantly, MDD has been used for diagrammatic models, annotated with quasi-formal constraints. However, recent work adds consideration of text models (programs, formal texts, meta-language texts). This raises the intriguing possibility of creating general integrations, to exploit whatever models or tools are suitable for a particular system, or to a particular research group's expertise.

Discussion

In common with Cohen and Harel (2007), we take the view that complex emergent systems cannot be constructed (or simulated) solely as hierarchies of sequential transformations. We must capture the concurrent, reactive nature of these systems. More explicitly, however, we must recognise the importance of scale dependencies and interactions across scales – the many concurrent inputs that Cohen and Harel (2007) observe are themselves the results of many concurrent outputs at other scales. In the reviewed approaches (RA, PEPA), the teams have, by chance or with great skill, selected areas of study where one or two adjacent levels, or scales, can be modelled to give realistic results that match the level of observation of the research scientists concerned.

In researching a broad platform for complex systems simulation, we would exploit work in mobility and process algebras. The inclusion of mobility allows the modelling of processes that dynamically change their relative location by changing their channels of communication with other processes, affording an effective way to model the structural plasticity within systems. One example is the Graphical Stochastic π-calculus (GSπ), developed by Phillips and Cardelli (2007) and proven equivalent to Milner (1999)'s π-calculus. This provides an accessible front-end environment whilst retaining the power of the underlying π-calculus. On a larger scale, the CoSMoS project (http://www.cosmos-research.org/) is building capacity in generic modelling tools and simulation techniques for complex systems; it is predicated on a well-founded process-oriented modelling platform, using the occam-π language. occam-π (http://occam-pi.org/) is a small language that implements the communication strengths of CSP (Hoare, 1985) and the mobile aspects of π-calculus; the well-grounded semantics of these specification calculi provide a formal basis for the programming environment, and support an engineering approach to the underlying mathematics (Barnes and Welch, 2004; Welch and Barnes, 2005). As in PEPA, process algebra can be used to prove properties (such as deadlock-freedom), but, here, the proven properties are then built in to the language (in the GSπ calculus; in the occam-π Kroc compiler) and used implicitly in models produced in the more-accessible front ends (the GSπ environment; occam-π code). Expertise in process algebra is not needed to use these languages, and occam-π can efficiently support millions of concurrent processes, distributed over multiple processors.

A general environment for simulation and modelling of complex systems is about more than concurrent mobile programming. The CoSMoS platform should eventually support many levels and scales – extending upwards (to observe more global effects) and downwards (in search of key causalities and origins). Seth Lloyd (2005) eloquently presented the ultimate simulation: the quantum computer that efficiently simulates the Universe (it is big!). A strict approach to modelling complex systems might expect to start at the very bottom – after all, classical physics emerges from quantum mechanics, and chemistry from classical physics. However, a rational view is that, at each level of interest, the effects of lower levels are of varying importance, and can sometimes be aggregated or omitted without a significant effect on the emergent behaviour. We hope that we will not need to engineer a Universe computer, but to successfully research and engineer complex systems we need tools that helps us to determine the relative importance of lower levels and of views of lower levels.

Finally, we are completely in agreement with Cohen and

Harel (2007) when they state, ... *a computer methodology [sic] that would allow us to zoom back and forth between lower-scale data and higher-scale behaviour while experimenting* in silico *is an ideal way — possibly the only way — to study emergence computationally.* (Cohen and Harel, 2007).

Acknowledgements

This work is part of the CoSMoS project, funded by EPSRC grant EP/E053505/1, and a Microsoft Research Europe PhD studentship.

References

Alexander, R. (2007). *Using Simulation for Systems of Systems Hazard Analysis*. PhD thesis, Department of Computer Science, University of York.

Barnes, F. and Welch, P. (2004). Communicating Mobile Processes. In East, I., Martin, J., Welch, P., Duce, D., and Green, M., editors, *Communicating Process Architectures 2004*, volume 62 of *Concurrent Systems Engineering Series*, pages 201–218. IOS Press.

Bryden, J. and Noble, J. (2006). Computational modelling, explicit mathematical treatments, and scientific explanation. In *Aretificial Life X*, pages 520–526. MIT Press.

Calder, M., Gilmore, S., and Hillston, J. (2006). Modelling the influence of RKIP on the ERK signalling pathway using the stochastic process algebra PEPA. *Transactions on Computational Systems Biology VII*, 4230:1–23.

Calder, M., Gilmore, S., Hillston, J., and Vyshemirsky, V. (2008). Formal methods for biochemical signalling pathways. In *Formal Methods: State of the Art and New Directions*. Springer.

Cavalcanti, A. and Woodcock, J. (2006). A tutorial introduction to CSP in Unifying Theories of Programming. In *Refinement Techniques in Software Engineering*, volume 3167 of *LNCS*, pages 220–268. Springer.

Cohen, I. R. and Harel, D. (2007). Explaining a complex living system: dynamics, multi-scaling and emergence. *Journal of the Royal Society Interface*, 4:175–182.

Dijkstra, E. W. (1975). Guarded commands, nondeterminacy and formal derivation of programs. *Communications of the ACM*, 18(8):453–457.

Efroni, S., Harel, D., and Cohen, I. R. (2007). Emergent dynamics of thymocyte development and lineage determination. *PLoS Computational Biology*, 3(1):0127–0135.

Epstein, J. M. (1999). Agent-based computational models and generative social science. *Complexity*, 4(5):41–60.

Hoare, C. (1985). *Communicating Sequential Processes*. Prentice-Hall.

Hoare, C. and He, J. (1998). *Unifying Theories of Programming*. Prentice Hall.

Hoare, C. A. R. (1969). An axiomatic basis for computer programming. *Communications of the ACM*, 12(10):576–580.

Kolovos, D., Paige, R., and Polack, F. (2006). The Eepsilon Oabject Llanguage (EOF). In *Second European Conference on Model-Driven Architecture*, volume 4066 of *LNCS*. Springer.

Lloyd, S. (2005). *Programming the Universe: A Quantum Computer Scientist Takes On the Cosmos*. Vintage.

Miller, G. F. (1995). Artificial life as theoretical biology: How to do real science with computer simulation. Technical Report Cognitive Science Research Paper 378, School of Cognitive and Computing Sciences, University of Sussex.

Milner, R. (1999). *The Pi Calculus*. Cambridge University Press.

Paolo, E. D., Noble, J., and Bullock, S. (2000). Simulation models as opaque thought experiments. In *Articial Life VII*, pages 497–506. MIT Press.

Phillips, A. and Cardelli, L. (2007). Efficient, correct simulation of biological processes in the stochastic π-calculus. In *Computational Methods in Systems Biology*, volume 4695 of *LNBI*, pages 184–199. Springer.

Polack, F., Stepney, S., Turner, H., Welch, P., and Barnes, F. (2005). An architecture for modelling emergence in CA-like systems. In *ECAL*, volume 3630 of *LNAI*, pages 433–442. Springer.

Regev, A., Panina, E. M., Silverman, W., Cardelli, L., and Shapiro, E. (2004). BioAmbients: an abstraction for biological compartments. *Theoretical Computer Science*, 325(1):141–167.

Reynolds, C. W. (1987). Flocks, herds, and schools: A distributed behavioral model (SIGGRAPH '87). *Computer Graphics*, 21(4):25–34.

Sadot, A., Fisher, J., Barak, D., Admanit, Y., Stern, M. J., Hubbard, E. J. A., and Harel, D. (2007). Towards verified biological models. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*.

Stepney, S., Polack, F., and Turner, H. (2006). Engineering emergence. In *ICECCS'06*, pages 89–97. IEEE Computer Society.

Swithinbank, P., Chessell, M., Gardner, T., Griffin, C., Man, J., Wylie, H., and Yusuf, L. (2005). *Patterns: Model-Driven Development Using IBM Rational Software Architect*. IBM Redbook. http://www.redbooks.ibm.com/abstracts/sg247105.html?Open.

Turner, H., Stepney, S., and Polack, F. (2007). Rule migration: Exploring a design framework for emergence. *Int. J. Unconventional Computing*, 3(1):49–66.

Welch, P. and Barnes, F. (2005). Communicating mobile processes: introducing occam-pi. In Abdallah, A., Jones, C., and Sanders, J., editors, *25 Years of CSP*, volume 3525 of *LNCS*, pages 175–210. Springer.

Wheeler, M., Bullock, S., Paolo, E. D., Noble, J., Bedau, M., Husbands, P., Kirby, S., and Seth, A. (2002). The view from elsewhere: Perspectives on alife modelling. *Artificial Life*, 8(1):87–100.

Woodcock, J. (2003). Unifying theories of parallel programming. In *Logic and Algebra for Engineering Software*. IOS Press.

Modelling Stigmergic Gene Transfer

Daniel Polani[1], Mikhail Prokopenko[2] and Matthew Chadwick[2]

[1]Department of Computer Science, University of Hertfordshire
Hatfield AL10 9AB, United Kingdom
[2]CSIRO Information and Communication Technology Centre
Locked bag 17, North Ryde, NSW 1670, Australia
Corresponding author: mikhail.prokopenko@csiro.au

Abstract

We consider an information-theoretic model studying the conditions when a separation between the dynamics of a 'proto-cell' and its proto-symbolic representation becomes beneficial in terms of preserving the proto-cell's information in a noisy environment. In particular, we are interested in understanding the behaviour at the "error threshold" level which, in our case, turns out to be a whole "error interval". We separate the phenomena into a "waste" and a "loss" component; the "waste" measures "packaging" information which envelops the proto-cell's information, but itself does not contain any information of interest, the "loss" measures how much of the proto-symbolically encoded information is actually lost. We observe that transitions in the waste/loss functions correspond to the boundaries of the "error interval". Secondly, we study whether and how different proto-cells can share such information via a joint code, even if they have slightly different individual dynamics. Implications for the emergence of biological genetic code are discussed.

Introduction

It can be argued that "the capacity to represent nucleic acid sequences symbolically in terms of a (colinear) amino acid sequence" (Woese, 2004) did not exist at the very early evolutionary stages, and developed only in response to certain environmental conditions. The phase of nucleic acid life that did not use genetic coding is separated from the later evolutionary stages where such coding became beneficial, by the "coding threshold". In this paper, we consider a model for evolutionary dynamics in the vicinity of the "coding threshold". The model is an extension of the model introduced by Piraveenan et al. (2007) who identified conditions under which a separation between a proto-cell and its symbolic encoding becomes beneficial in terms of preserving the information within a noisy environment.

It is important to realize two features of the early phase in cellular evolution that existed before the "coding threshold". First of all, the "players are cell-like entities still in early stages of their evolution", and that "the evolutionary dynamics...involves communal descent" (Vetsigian et al., 2006). That is, the cells are not yet well-formed entities that replicate completely, with an error-correcting mechanism . Rather, the proto-cells can be thought of as conglomerates of substrates, that exchange components with their neighbours freely — horizontally. The notion of vertical descent from one generation to the next is not yet well-defined. This means that the descent with variation from one generation to the next is not genealogically traceable but is a descent of a cellular community as a whole.

Secondly, genetic code that appears at the coding threshold is "not only a protocol for encoding amino acid sequences in the genome but also an innovation-sharing protocol" (Vetsigian et al., 2006), as it used not only as a part of the mechanism for cell replication, but also as a way to encode relevant information about the environment. Different proto-cells may come up with different innovations that make them more fit to the environment, and the "horizontal" exchange of such information may be assisted by an innovation-sharing protocol - a proto-code. With time, the proto-code develops into a universal genetic code.

Such innovation-sharing is perceived to have a price: it implies ambiguous translation where the assignment of codons to amino acids is not unique but spread over related codons and amino acids. (Vetsigian et al., 2006). In other words, accepting innovations from neighbours requires that the receiving proto-cell is sufficiently flexible in translating the incoming fragments of the proto-code. Such a flexible translation mechanism, of course, would produce imprecise copies. However, a descent of the whole innovation-sharing community may be traceable: i.e., in a statistical sense, the next "generation" should be correlated with the previous one. As noted by Woese (2004), "a sufficiently imprecise translation mechanism could produce "statistical proteins", proteins whose sequences are only approximate translations of their respective genes (Woese, 1965). While any individual protein of this kind is only a highly imprecise translation of the underlying gene, a consensus sequence for the various imprecise translations of that gene would closely approximate an exact translation of it". That is, the consensus sequence would capture the main information content of the innovation-sharing community.

Moreover, it can be argued that the universality of the

code is a generic consequence of early communal evolution mediated by horizontal gene transfer (HGT), and that thus HGT enhances optimality of the code (Vetsigian et al., 2006):

> HGT of protein coding regions and HGT of translational components ensures the emergence of clusters of similar codes and compatible translational machineries. Different clusters compete for niches, and because of the benefits of the communal evolution, the only stable solution of the cluster dynamics is universality.

In this paper, we adopt an information-theoretic view that allows us to concentrate on generic processes common to a collection of primitive cells rather than on specific biochemical interactions within an environmental locality. Moreover, it allows us to handle particular HGT scenarios where certain fragments necessary for cellular evolution begin to play the role of the proto-code. One scenario may assume that the proto-code is initially located within its proto-cell, and is functionally "separated" from the rest of the cell when such a split becomes beneficial. Another scenario suggests that the proto-code is present in an environmental locality, and subsequently entrapped by the proto-cells that benefit from such interactions. We believe that the first scenario ("internal split") is less likely to produce either universal code or universal translational machinery than the second scenario ("entrapment"). In general, it is quite possible that internal split and entrapment played complementary roles. Importantly, however, there was an indirect exchange of information among the cells via their local environment, which is indicative of stigmergy. Henceforth, we would like to refer to such gene transfer as *stigmergic gene transfer* (SGT): proto-cells find matching fragments, use them for coding, modify and evolve their translation machinery, and exchange certain fragments with each other via the local environment. SGT can be thought of as a sub-class of HGT, differing from the latter in that the fragments exchanged between two proto-cells may be modified during the transfer process by other cells in the locality.

It is conjectured that maximization of information transfer through selected channels is one of the main evolutionary pressures (Prokopenko et al., 2006; Klyubin et al., 2007; Piraveenan et al., 2007; Laughlin et al., 2000; Bialek et al., 2007): although the evolutionary process involves a larger number of drives and constraints, information preservation is a consistent motif throughout biology. Adami, for instance, argues that the evolutionary process extracts valuable information and stores it in the genes (Adami, 1998). Since this process is relatively slow (Bennett, 1990; Lloyd, 1990), it is a selective advantage to preserve this information, once captured.

In this paper, we follow the model of Piraveenan et al. (2007), and focus on the information preservation property of evolution within a coupled dynamical system. Piraveenan et al. (2007) verified that the ability to symbolically encode nucleic acid sequences does not develop when environmental noise φ is too large or too small. In other words, it is precisely a limited reduction in the information channel's capacity, brought about by the environmental noise, that creates the appropriate selection pressure for the separation between a proto-cell and its encoding.

Here we extend the model of Piraveenan et al. (2007) by identifying both encoding and translation that maximize the ability to recover as much original information as possible in the face of environmental noise and in presence of an imperfect internal processing. In doing so, we enhance the analysis by considering both the loss and the waste of the information. Finally, we study effects of co-evolution of multiple encodings entrapped by multiple ensembles using SGT.

Modelling evolutionary dynamics

Our generic model for evolutionary dynamics involves a dynamical coupled system, where a proto-cell is coupled with its potential encoding, evolving in a fitness landscape shaped by a selection pressure. The selection pressure rewards preservation of information in presence of both environmental noise and inaccuracy of internal coupling. When the proto-cell is represented as a dynamical system, the information about it may be captured generically via the structure of the phase-space (e.g., states and attractors) of the dynamical system.

For example, the states of the system may loosely correspond to dominant substrates (e.g., prototypical amino acids), used by the cell. The chosen representation does not have to deal with the precise dynamics of biochemical interactions within the cell, but rather focuses on structural questions of the cell's behavior: does it have more than one attractor, are the attractors stable (periodic) or chaotic, how many states do the attractors cycle through, etc. Representing the dynamics in this way avoids the need to simulate the unknown cellular machinery, but allows us to analyze under which environmental conditions the SGT may have become beneficial. In particular, if the potential encoding develops to have a compact structure that matches the structure of the cell's phase-space, then the encoding would be useful in recovering such structure, should it be affected by environmental noise. Information is understood in Shannon sense (reduction of uncertainty), and a loss of such information corresponds to a loss of structure in the phase-space. At the same time, informational recovery would correspond to recovery of some isomorphic structure in the phase-space.

The generic dynamical coupled system is described by the equations

$$X_{t,m} = \begin{cases} f_m\left(X_{t-1,m}\right) + \varphi_t & t \neq t^* \\ \alpha\left[f_m\left(X_{t-1,m}\right) + \varphi_t\right] + & \\ (1-\alpha) h_m\left(Y_{t-1,m} + \psi_{t,m}\right) & t = t^* \end{cases} \quad (1)$$

$$Y_{t,m} = \begin{cases} g_m\left(X_{t,m} + \psi_{t,m}\right) & t = t_0 \\ Y_{t-1,m} & t > t_0 \end{cases} \quad (2)$$

where $X_{t,m}$ are the variables that describe multiple proto-cells, $1 \leq m \leq M$, and and $Y_{t,m}$ their potential encodings at time t, respectively. Function f_m defines the dynamical system representing the dynamic for proto-cell m. Parameter $\alpha \in [0, 1]$ sets the relative importance of the translation h from symbols (e.g., proto-codons) into the proto-cell state (e.g., proto amino acids).

In the simplest case, $m = 1$ (one cell), and $\alpha = 1/2$, the system reduces to

$$X_t = \begin{cases} f\left(X_{t-1}\right) + \varphi_t & t \neq t^* \\ \frac{1}{2}\left[f\left(X_{t-1}\right) + \varphi_t\right] + \frac{1}{2}h\left(Y_{t-1} + \psi_t\right) & t = t^* \end{cases} \quad (3)$$

$$Y_t = \begin{cases} g\left(X_t + \psi_t\right) & t = t_0 \\ Y_{t-1} & t > t_0 \end{cases} \quad (4)$$

The function φ_t describes the external (environment) noise that affects the proto-cells: it is the same for all cells, i.e, φ_t is independent of m. It is implemented as a random variable $\varphi_t \in [-l, u]$, where $u > 0$ and $l > 0$, which is uniformly distributed, with probability $1/2$, between 0 and l, and with probability $1/2$ between 0 and u (sampled at each time step). The function $\psi_{t,m}$ represents both the matching noise associated with accessing information from $X_{t_0,m}$ by $Y_{t_0,m}$ at time t_0, and the noise of ambiguous back-translation (applied only at t^*). In other words, it represents the inaccuracy within the internal encoding/translation channel. This noise is modelled as uniform random noise $\psi_{t,m} \in [-b_m, b_m]$, where $0 < b_m \ll 1.0$, and is used only at t_0 and t^*.

The entrapment mechanism that matches information from the proto-cell with its encoding (i.e. which encodes its information) at time t_0 is given by g_m. At time $t = t_0$, noise is introduced into the environment affecting dynamics of the proto-cell. At the time $t = t_0$, information from the proto-cell $X_{t_0,m}$ is accessed by the system $Y_{t_0,m}$ (encoding) via the matching function g_m. This process is affected by the noise ψ. The feedback from Y to X (henceforth we drop subscripts when the meaning is clear) occurs at time t^*, i.e. the function h_m translates the input $Y_{t^*-1,m}$ from the encoding back into the proto-cell. This internal translation is subjected to internal noise as well.

Piraveenan et al. (2007) considered the case $m = 1$, equations (3)–(4), and function h being the identity (a single system). Here we consider a system with multiple proto-cells: $m \geq 1$, and contrast universality of the translation machinery: all functions h_m are identical, while $g_i \neq g_j$ for $i \neq j$, with universality of the proto-code: all proto-codes g_m are identical, while $h_i \neq h_j$ for $i \neq j$. We would like to point out that the system (1)–(2) is coupled not only due to the common environment noise φ, but also due to the shared translation machinery h or shared proto-code g. This coupling supports a simple information-theoretic model of HGT and specifically, SGT. As we are dealing only with the information content, the consideration of identical h_m's and/or identical g_m's allows us to study gene transfers without details of molecular (state-to-state) interactions.

Coupled logistic maps

The dynamical system employed is a logistic map $X_{t+1} = rX_t\left(1 - X_t\right)$, where r is a parameter, i.e. the function f_m is given by $f\left(x\right) = r_m x\left(1 - x\right)$. The logistic map f is initialized with a value between 0.0 and 1.0, and stays within this range if the value of r is within the range $[0, 4.0]$. We used $r = 3.5$ (for the single system), resulting in four states of the attractor of the logistic map (approximately $0.38, 0.50, 0.83, 0.87$). For multiple proto-cells, we used proto-cells with $r = 3.5$ as well as with $r = 3.46$ and $r = 3.48$. Each of these possesses four states of the respective attractor. The time $t = t_0$ is set after the logistic map settles into its attractors, having passed through a transient. The functions g and h are mappings from $[0, 1]$ to $[0, 1]$.

Coupled logistic maps have been extensively used in modelling of biological processes. One prominent study is the investigation of spatial heterogeneity in population dynamics (Lloyd, 1995) who examined the dynamic behaviour of the model using numerical methods and observed a wide range of behaviours. For instance, the coupling was shown to stabilize individually chaotic populations as well as cause individually stable periodic populations to undergo more complex behaviour. Importantly, a single logistic map can only have one attracting periodic orbit, but multiple attractors were shown by Lloyd (1995) for coupled logistic maps.

Logistic maps were chosen to model the system (1)–(2) mostly due to their simplicity, well-understood behaviour in the vicinity of chaotic regimes (e.g., bifurcations and symmetry breaking), the possibility of multiple attractors in coupled maps, as well as their ability to capture both reproduction and starvation effects (that are important for studying the structure in the phase-space).

Information preservation

In evolving the potential encoding system Y coupled with X via a suitable function g, we minimize Crutchfield's information distance (Crutchfield, 1990) between the initial X_{t_0} and recovered X_{t^*} states of the system:

$$d(X_{t_0}, X_{t^*}) = H(X_{t_0}|X_{t^*}) + H(X_{t^*}|X_{t_0}) \quad (5)$$

The entropies are defined as

$$H(A) = -\sum_{a \in A} P(a)\ \log P(a)\ , \quad (6)$$

$$H(A, B) = -\sum_{a \in A}\sum_{b \in B} P(a, b)\ \log P(a, b)\ , \quad (7)$$

$$H(A|B) = H(A,B) - H(B) \quad (8)$$

where $P(a)$ is the probability that A is in the state a, and $P(a,b)$ is the joint probability.

The distance $d(X_{t_0}, X_{t^*})$ measures the dissimilarity of two information sources X_{t_0} and X_{t^*}; it is a true metric in the sense that it fulfils the axioms of metrics, including the triangle inequality. In addition, as opposed to the mutual information used in Piraveenan et al. (2007), the information metric d is sensitive also to the case when one information source is contained within another. While the results do not radically depend on the choice of distance d over the mutual information, the former leads to a more crisp recovery of structure in the phase-space.

The use of d also indicates the presence of two components of dissimilarity. The first is the loss of information, $H(X_{t_0}|X_{t^*})$, which measures how much uncertainty the final state has about the original state of the system. The second is the waste, $H(X_{t^*}|X_{t_0})$; as the system will aim to preserve as much information as possible about the state X_{t_0} (and only this information), any additional variability in X_{t^*} will be considered as "waste".

Minimization of the information distance (more precisely, maximization of $-d(X_{t_0}, X_{t^*})$) is achieved by employing a simple genetic algorithm (GA) (described in the Appendix).

In order to estimate the probability distribution of a random variable (X or Y) at a given time, we generate an initial random sample $(X_0) = (X_0^1, X_0^2, \ldots, X_0^K)$ of size K. Each X_0^i, where $1 \leq i \leq K$, is chosen from a uniform random distribution within $[0.0, 1.0]$. The mapping $X_{t+1}^i = f(X_t^i)$ produces an ensemble of K corresponding time series, $1 \leq i \leq K$, denoted as $[X] = [X_t^1, X_t^2, \ldots, X_t^K]$, where $0 \leq t \leq T$, and T is a time horizon. Within the ensemble, each time series X_t^i may have a different initial value X_0^i. At any given time t', we can obtain a sample $(X_{t'}) = (X_{t'}^1, X_{t'}^2, \ldots, X_{t'}^K)$.

Given the sample (X_{t_0}) at the time $t = t_0$, and the mapping $Y_{t_0} = g(X_{t_0} + \psi)$, we can generate the sample $(Y_{t_0}) = (Y_{t_0}^1, Y_{t_0}^2, \ldots, Y_{t_0}^K)$ for the variable Y. In the corresponding ensemble $[Y] = [Y_t^1, Y_t^2, \ldots, Y_t^K]$ each sample is identical to the the sample (Y_{t_0}).

Recapitulation of the Results for a Single System

We begin by revisiting the simple case $m = 1$ that was considered by Piraveenan et al. (2007): the function h is identity $h(y) = y$. The structure evolving in Y can be associated with "proto-symbols" ("codes") that help to retrieve at time t^* some (or most of the) information stored at t_0.

Figure 1 shows the ensemble $[X]$ at the time $t^* - 1$, i.e. right before the moment when the feedback from Y to X occurs. The environment noise φ ($u = 0.025$ and $l = 0.025$) disrupts the logistic map dynamics, and some information about the attractor of X and its four states is lost in the course of time: the observed sample (X_{t^*-1}) does not contain four clear clusters.

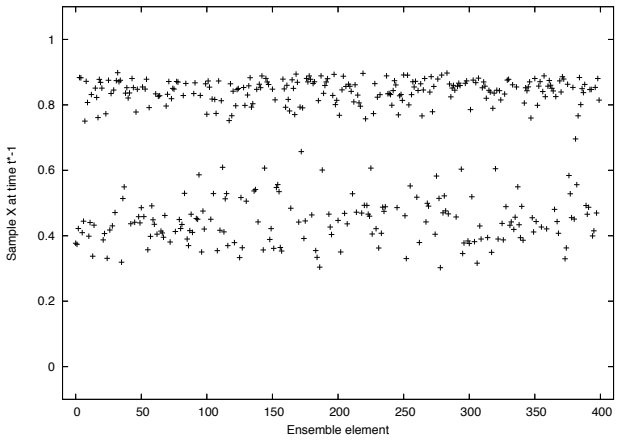


Figure 1: Two remaining "clusters" in the sample (X_{t^*-1}).

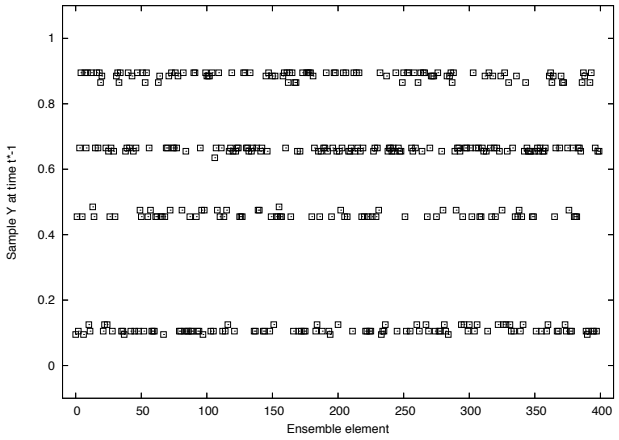


Figure 2: Evolved g (noise $\varphi = \pm 0.025$; $\psi = \pm 0.015$) containing four clusters in the encoding (Y_{t^*-1}). Function h is identity.

Figure 2 shows the evolved encoding ensemble $[Y]$ at the time $t^* - 1$, while Figure 3 shows the recovered ensemble $[X]$ of the evolved coupled system at the time t^*. The sample (Y_{t^*-1}) settles into four clusters that can be easily represented by four "codes" corresponding to the four states of the attractor of X. The evolved encoding allows to recover the information within X, as evidenced by four clear clusters within the sample (X_{t^*}).

The clustering corresponds to the emergence of discrete "proto-symbols" in the encoding Y. The information reconstructed at time t^* is not precise, and rather than having four crisp states, X can be described as an individual with an imprecise translation of the underlying gene within a "consensus sequence" (Woese, 2004), analogous to a "statistical protein". So far the recapitulation of the past results.

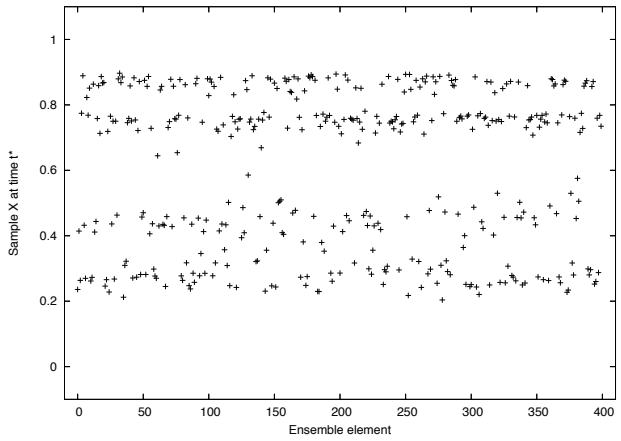


Figure 3: Four recovered clusters in sample (X_{t^*}). $d(X_{t_0}, X_{t^*}) \approx 1.5$ bits. Contrast with Figure 1.

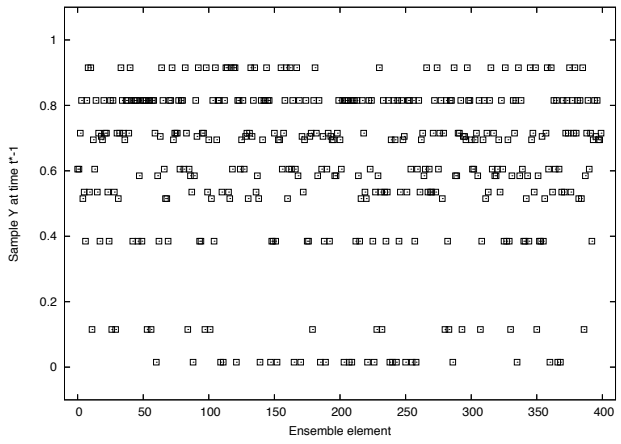


Figure 4: Evolved g (noise $\varphi = \pm 0.025$; $\psi = \pm 0.015$), with variable function h (see Figure 5).

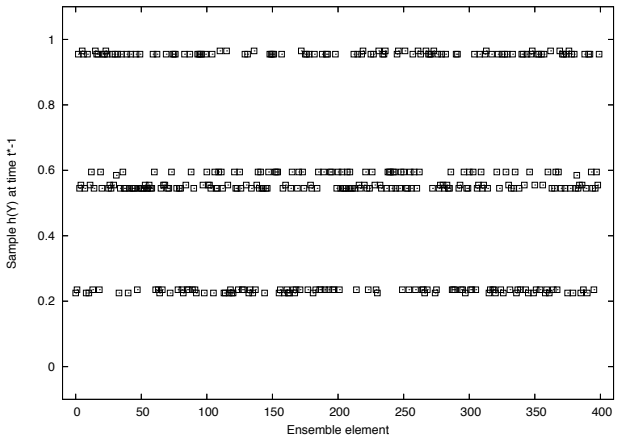


Figure 5: Evolved h (noise $\varphi = \pm 0.025$; $\psi = \pm 0.015$), complementing the encoding g (see Figure 4).

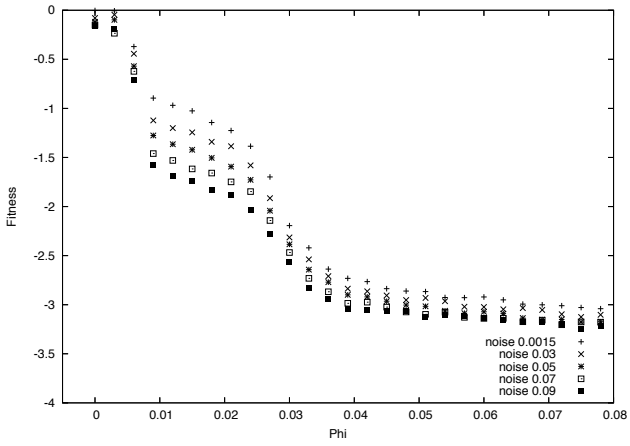


Figure 6: Fitness, i.e. $-d(X_{t_0}, X_{t^*})$, over noise φ, for different noise levels ψ.

Optimizing the Recovery Function h

Now we consider the extended case where the translation function h is subject to optimization as well. This time, the evolved encoding ensemble $[Y]$ at the time $t^* - 1$ (Figure 4), does not have four clear clusters. However, this lack of adequate encoding is complemented by a more refined translation that evolved in parallel, as evidenced by Figure 5. The end result (not shown) is analogous to the one presented by Figure 3.

Figure 6 traces fitness, $-d(X_{t_0}, X_{t^*})$ (for the best individual), over the external noise φ, for different internal noise levels ψ. We can observe a steady decrease in fitness punctuated by two sharper transitions, that form three plateaus. As conjectured by Piraveenan et al. (2007), the encoding is not beneficial when the environmental noise φ is outside a certain range. The middle plateau is precisely the region specifying this range, i.e. the "error interval". It is also evident that within this plateau, sensitivity to internal noise ψ is the highest.

Be reminded that the information distance $d(X_{t_0}, X_{t^*})$ consists of two components: the loss $H(X_{t_0}|X_{t^*})$, and the waste $H(X_{t^*}|X_{t_0})$. The waste measures packaging information which envelops the proto-cell's information, but itself does not contain any information of interest, while the loss measures how much of the proto-symbolically encoded information is actually lost. Figure 7 plots fitness over noise φ, for specific ψ, and shows loss and waste for the best individual. At the first plateau (very small noise), $d(X_{t_0}, X_{t^*}) = 0$, and both loss and waste are zero. At the medium plateau, the recovered system cannot get any closer to X, because the waste cannot be avoided, while the loss is still zero or minimal. At the last plateau ($\varphi > 0.025$), the loss begins to increase for the first time. So not only is there a waste, but the recovered system loses some information. The loss reaches 0.5, waste reaches 2.5, and $d(X_{t_0}, X_{t^*})$ reaches 3.0 (twice as large as the distance at the medium plateau). So the cascade of plateaus is explained by: (i) everything is recoverable (the first plateau); (ii) waste appears (the medium plateau); (iii) loss appears (the last plateau).

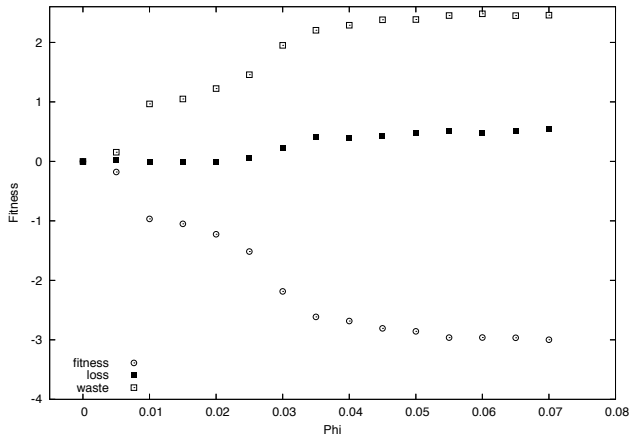


Figure 7: Fitness $-d(X_{t_0}, X_{t^*})$, loss $H(X_{t_0}|X_{t^*})$, and waste $H(X_{t^*}|X_{t_0})$, over noise φ, for specific $\psi = 0.015$.

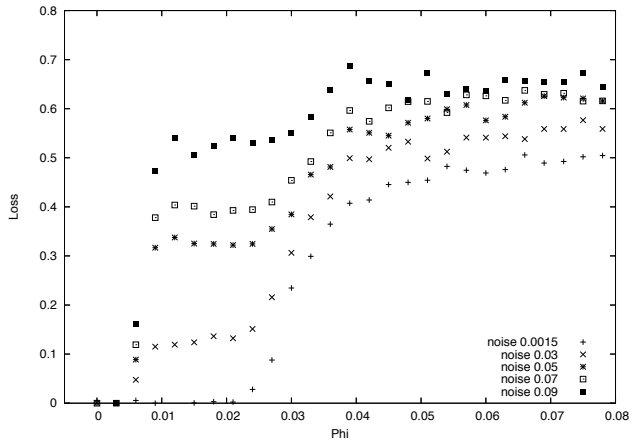


Figure 8: Loss $H(X_{t_0}|X_{t^*})$ over noise φ, for different noise levels ψ.

Figures 8 and 9 "zoom" into the dynamics of loss and waste for different levels of internal noise ψ, and show that the loss also appears if the internal noise $\psi > 0.015$. It is also evident that the loss is more sensitive to internal noise than waste. The waste, on the other hand, simply follows the cascade of plateaus. The difference between loss and waste is highlighted in Figure 10 that traces the ratio $\frac{H(X_{t_0}|X_{t^*})}{H(X_{t^*}|X_{t_0})}$. This ratio is most turbulent at the medium plateau, supporting the hypothesis of its special role. We note that the transitions in the waste/loss functions correspond to the boundaries of the medium plateau, marking the "error interval".

Results for multiple systems

In this section, we now focus on a system with multiple proto-cells which share the coding channel. Concretely, we consider $m = 3$ ($r = 3.5$, $r = 3.46$, and $r = 3.48$), and contrast the universality of the translation machinery with the universality of the proto-code.

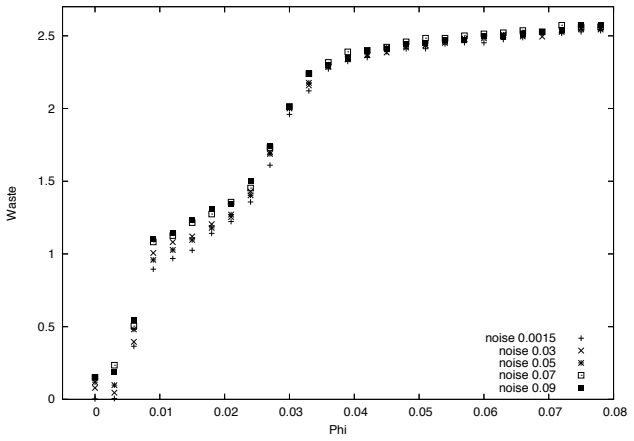


Figure 9: Waste $H(X_{t^*}|X_{t_0})$ over noise φ, for different noise levels ψ.

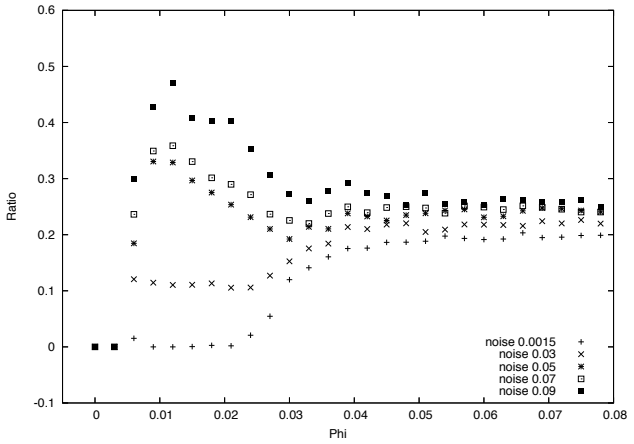


Figure 10: Loss/waste ratio over noise φ, for different noise levels ψ.

Single g and multiple h

Let us assume that all available proto-codes g_m are identical (universal code), but $h_i \neq h_j$ for $i \neq j$. In this case, the system achieves recovery comparable with the single system for each of the logistic maps, $d(X_{t_0}, X_{t^*}) \approx 1.5$, but the structure of the code is slightly different. As shown in Figure 11, for the logistic map $r = 3.5$ fewer clusters evolve than for the singular system (shown in Figure 4). However, the translation machinery depicted in Figure 12 is as structured as that of the singular system (shown in Figure 5). This supports a conjecture that multiple systems exert some pressure for proto-code's universality.

Multiple g and single h

Here we consider the opposite case: abundance of available proto-codes: $g_i \neq g_j$ for $i \neq j$, but translation machinery is universal: all functions h_m are identical. Again, the system achieves the recovery of the singular system for each of the logistic maps, $d(X_{t_0}, X_{t^*}) \approx 1.5$, but the structure of both the code and translation machinery is more compact, as

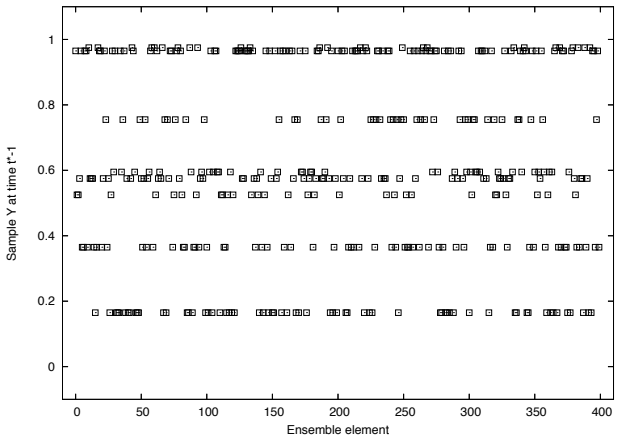


Figure 11: Single g. Evolved g (noise $\varphi = \pm 0.025$; $\psi = \pm 0.015$), for three ensembles with variable function h (see Figure 12). Shown for ensemble with $r = 3.5$.

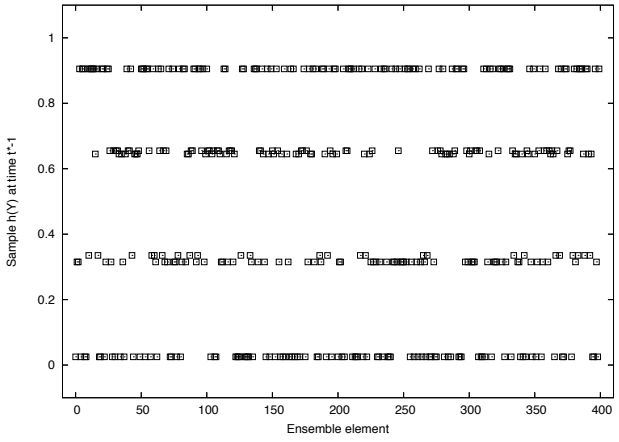


Figure 12: Single g. Evolved h (noise $\varphi = \pm 0.025$; $\psi = \pm 0.015$), for ensemble with $r = 3.5$, co-evolved with three ensembles; complementing the encoding g (see Figure 11).

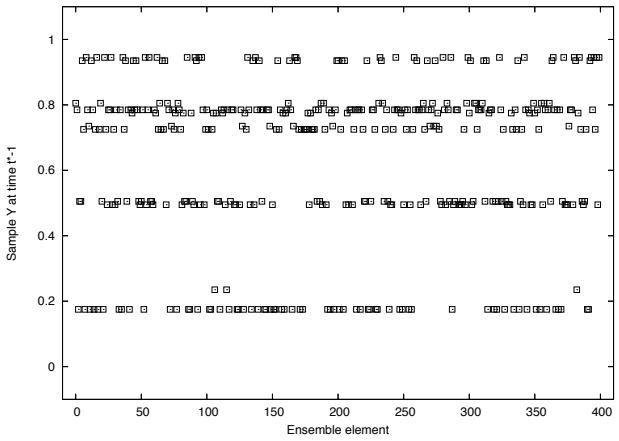


Figure 13: Single h. Evolved g (noise $\varphi = \pm 0.025$; $\psi = \pm 0.015$), for ensemble with $r = 3.5$, co-evolved with three ensembles; complementing the translation h (see Figure 14).

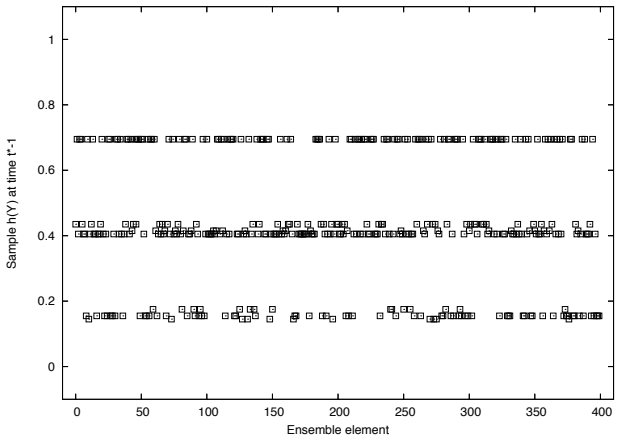


Figure 14: Single h. Evolved h (noise $\varphi = \pm 0.025$; $\psi = \pm 0.015$), for three ensembles with variable function g (see Figure 13).

shown in Figures 13 and 14. This supports a conjecture that co-evolution of multiple systems may yield not only universality of proto-code, but also uniform translation machinery.

Conclusion and Future Work

We considered an information-theoretical model based on dynamical systems for the emergence of protected informational channels able to preserve information in a system over time when the main channel is suffering from perturbations. Doing so, we extended previous work, by not only introducing the optimization of a backtranslation mechanism, but also the consideration of the information metric and the more refined analysis able to resolve loss as well as waste in the resulting encoding. Furthermore we studied the effects on a small population of systems sharing an encoding.

It is striking that the pressure to develop a distinctive "symbolic" encoding does only develop if the noise in the original system is in a particular range, not too small and not too large.

Scanning through different noise levels, we observe several plateaus of the fitness corresponding to qualitative jumps in the way not only the initial state is encoded but how the system dynamics is affected by the noise. The middle plateau which is most relevant for the emergence of distinct symbols turns out to be the most sensitive for the precise level of noise.

The waste/loss analysis shows that with increasing noise, first the waste grows away from 0, at first without any loss. Only at higher noise levels the loss begins its growth. These transitions correspond closely to the plateau transitions.

The multiple system scenario shows that joint translation "machineries" can be successfully used by several systems which differ slightly. However, at this point, we did not yet model the competition between different translation and information exchange models. This will be addressed in future work.

Appendix

We generate an ensemble of X_t time series, each series governed by equation (1). The ensemble $[X]$ provides a fixed constraint on the optimization. For *each* function g, an ensemble $[Y]$ is then generated, using equation (2) — i.e., the values of the series Y_t depend on the choice of function g (and function h). The ensemble $[X]$ is kept unchanged while we evolve the population of functions g (and h), being an optimization constraint, but the ensemble $[Y]$ differs for each individual within the population. The fitness of each function g (and h) is determined by the negative distance between X_{t_0} and X_{t^*}, denoted $d(X_{t_0}; X_{t^*})$, defined by equation (5), and estimated via the respective conditional entropies between samples (X_{t_0}) and (X_{t^*}).

Since the information from Y_{t^*-1} (different for each individual) is fed back into X_{t^*}, equation (1), the sample (X_{t^*}) is specific for each individual within the population. Therefore, it may be contrasted with the sample (X_{t_0}) which is identical across the population, producing distinct fitness values $I_g(X_{t_0}; X_{t^*})$ for each individual g. The experiments were repeated for different ensembles X_t.

We generate a population of g (and h) functions (the size of the population is fixed at 400). In order to implement the mapping g, the domain of g is divided into n consecutive bins x_i such that $x_i = [(i-1)/n, i/n)$ for $1 \leq i < n$, where [a,b) denotes an interval open on the right, and $x_n = [(n-1)/n, 1]$. The range of g is divided into m consecutive bins y_j such that $y_j = [(j-1)/m, j/m)$ for $1 \leq j < m$, and $y_m = [(m-1)/m, 1]$. Then each bin x_i in the domain is mapped to a bin y_j in the range: $G : x_i \rightarrow y_j$, where G represents the discretized mapping. Formally, any $x \in x_i$ is mapped to $g(x) \equiv \overline{G(x_i)}$, where $\overline{G(x_i)}$ is the median value of the bin $G(x_i)$. For example, if $n = 100$, $m = 10$, and $y_7 = G(x_{30})$, that is, the bin $x_{30} = [0.29, 0.30)$ is mapped to the bin $y_7 = [0.6, 0.7)$, then for any $x \in x_{30}$ (e.g., $x = 0.292$), the function $g(x)$ would return $0.65 = \overline{y_7}$.

Therefore, in the GA, each function g can be encoded as an array of n integers, ranging from 1 to m, so that the i-th element of the array (the i-th digit) represents the mapping $y_j = G(x_i)$, where $1 \leq j \leq m$. Function h is coded analogously.

We have chosen a *generation gap* replacement strategy. In our experiments, we set the generation gap parameter 0.3. In other words, the entire old population is sorted according to fitness, and we choose the best 30% for direct replication in the next generation, employing an elitist selection mechanism. The rest of selection functionality is moved into the (uniform) crossover. Mutation is implemented as additive creeping or random mutation, depending on the number of "digits" in the genome. If the number of digits is greater than 10, then additive creeping is used: a digit can be mutated within $[-5\%, +5\%]$ of its current value. If the number of digits is less than 10, the random mutation is used with the mutation rate of 0.01.

Acknowledgments

The authors are grateful to Joseph Lizier for open and motivating discussions; and to Mahendra Piraveenan for his exceptionally valuable prior contribution to this effort.

References

Adami, C. (1998). *Introduction to Artificial Life*. Springer, New York.

Bennett, C. H. (1990). How to define complexity in physics, and why. In Zurek (1990), pages 137–148.

Bialek, W., de Ruyter van Steveninck, R. R., and Tishby, N. (2007). Efficient representation as a design principle for neural coding and computation. In preparation.

Crutchfield, J. P. (1990). Information and its Metric. In Lam, L. and Morris, H. C., editors, *Nonlinear Structures in Physical Systems – Pattern Formation, Chaos and Waves*, pages 119–130. Springer Verlag.

Klyubin, A., Polani, D., and Nehaniv, C. (2007). Representations of space and time in the maximization of information flow in the perception-action loop. *Neural Computation*, 19(9):2387–2432.

Laughlin, S. B., Anderson, J. C., Carroll, D. C., and de Ruyter van Steveninck, R. R. (2000). Coding efficiency and the metabolic cost of sensory and neural information. In Baddeley, R., Hancock, P., and Földiák, P., editors, *Information Theory and the Brain*, pages 41–61. Cambridge University Press.

Lloyd, A. (1995). The coupled logistic map: A simple model for the effects of spatial heterogeneity on population dynamics. *J. Theor. Biol.*, 173:217–230.

Lloyd, S. (1990). Valuable information. In Zurek (1990), pages 193–197.

Piraveenan, M., Polani, D., and Prokopenko, M. (2007). Emergence of genetic coding: an information-theoretic model. In Almeida e Costa, F., Rocha, L., Costa, E., Harvey, I., and Coutinho, A., editors, *Advances in Artificial Life: 9th European Conference on Artificial Life (ECAL-2007), Lisbon, Portugal, September 10-14*, volume 4648 of *Lecture Notes in Artificial Intelligence*, pages 42–52. Springer.

Prokopenko, M., Gerasimov, V., and Tanev, I. (2006). Evolving spatiotemporal coordination in a modular robotic system. In Nolfi, S., Baldassarre, G., Calabretta, R., Hallam, J., Marocco, D., Meyer, J.-A., and Parisi, D., editors, *From Animals to Animats 9: 9th International Conference on the Simulation of Adaptive Behavior (SAB 2006)*, volume 4095 of *Lecture Notes in Computer Science*, pages 558–569. Springer.

Vetsigian, K., Woese, C., and Goldenfeld, N. (2006). Collective evolution and the genetic code. *PNAS*, 103(28):10696–10701.

Woese, C. R. (1965). On the evolution of the genetic code. *Proc. Natl. Acad. Sci. USA*, 54:1546–1552.

Woese, C. R. (2004). A new biology for a new century. *Microbiology and Molecular Biology Reviews*, 68(2):173–186.

Zurek, W. H., editor (1990). *Complexity, Entropy and the Physics of Information*, Santa Fe Studies in the Sciences of Complexity, Reading, Mass. Addison-Wesley.

The Efficacy of Group Selection is Increased by Coexistence Dynamics within Groups

Simon T. Powers, Alexandra S. Penn and Richard A. Watson

School of Electronics & Computer Science, University of Southampton, Highfield, Southampton SO17 1BJ, UK
stp05r@ecs.soton.ac.uk

Abstract

Selection on the level of loosely associated groups has been suggested as a route towards the evolution of cooperation between individuals and the subsequent formation of higher-level biological entities. Such group selection explanations remain problematic, however, due to the narrow range of parameters under which they can overturn within-group selection that favours selfish behaviour. In principle, individual selection could act on such parameters so as to strengthen the force of between-group selection and hence increase cooperation and individual fitness, as illustrated in our previous work. However, such a process cannot operate in parameter regions where group selection effects are totally absent, since there would be no selective gradient to follow. One key parameter, which when increased often rapidly causes group selection effects to tend to zero, is initial group size, for when groups are formed randomly then even moderately sized groups lack significant variance in their composition. However, the consequent restriction of any group selection effect to small sized groups is derived from models that assume selfish types will competitively exclude their more cooperative counterparts at within-group equilibrium. In such cases, diversity in the migrant pool can tend to zero and accordingly variance in group composition cannot be generated. In contrast, we show that if within-group dynamics lead to a stable coexistence of selfish and cooperative types, then the range of group sizes showing some effect of group selection is much larger.

Introduction

The evolution of cooperation between biological individuals, both generally and as a vital part of the formation of new higher-level composite individuals, is an important and much discussed open question in both evolutionary biology (Maynard Smith and Szathmary, 1995; Keller, 1999; Michod, 1999; Hammerstein, 2003; Okasha, 2006) and artificial life (Bedau et al., 2000). The fundamental problem is that any group of cooperative types, whose members donate some component of individual fitness in order to benefit their group, is vulnerable to invasion by selfish cheats that reap the group benefits of cooperation without paying the individual cost. Various mechanisms by which cooperation can nevertheless evolve have been suggested, including so-called 'group selection' models in which population structure exists such that individuals spend part of their time in groups (rather than freely mixed in the whole population) before mixing in a migrant pool from which new groups are formed (Wilson, 1980). Although selection within any given group will always favour selfish individuals, groups with a higher proportion of cooperators are more productive and hence contribute more individuals to the migrant pool and the next generation of groups.

Such models have been found to allow cooperation to evolve or be preserved in a population, but only within certain narrow parameter ranges (discussed below). This clearly presents a problem when appealing to such mechanisms as a route towards the evolution of higher-level individuals. As we have argued in previous work (Powers et al., 2007), the conditions that allow group selection to be effective and control its strength need not be externally imposed. Instead, they may be products of individual characters and hence subject to individual selection. Specifically, if key parameters such as group size are subject to individual adaptation (for example, via production of extracellular matrix in a bacterial biofilm), then a process akin to niche construction (Odling-Smee et al., 2003) supporting the evolution of cooperation may occur whereby cooperative traits and those affecting the strength of group selection evolve concurrently.

However, this process would require the existence of an adaptive gradient, such that a small parameter change could increase the strength of group selection and consequently cooperation and individual fitness. In this paper we consider one important parameter, group size. Where groups are formed randomly, existing models have shown that increasing group size rapidly causes the measurable effect of group selection to reach zero (Wade, 1978). This then means that for a large region of parameter space, a small decrease in group size, e.g. by individual mutation, would have no effect on fitness, i.e. there would be no selective gradient towards smaller groups and increased individual fitness. However, the model developed and presented in this paper suggests that the rapid tendency of group selection effects to zero is a consequence of an assumption of directional within-group selection. Specifically, classical group selection mod-

els attempt to explain the global promotion of a cooperative allele that is driven extinct at within-group equilibrium through directional selection for a rival selfish allele. While this assumption of directional selection and hence competitive exclusion of types is commonplace in population genetics models, a competitive *coexistence* of two types is instead often permitted in ecological models. For example, the classical Lotka-Volterra competition equations allow for coexistence as well as exclusion (May, 1976). Such dynamics are of relevance to potential multi-species group selection scenarios; for example, during egalitarian major evolutionary transitions in which different unrelated individuals eventually form a new level of selection, and within extant multi-species consortia such as bacterial biofilms (Burmolle et al., 2006) in which group selection effects may be pertinent. [1] Results presented in this paper show that where a within-group stable coexistence of types exists, measurable group selection effects are sustained over a much larger range of group sizes, potentially providing an individual adaptive gradient towards smaller groups that enhance group selection over a much larger range of parameter space.

The Limits of Group Selection

Group selection theory is typically understood as the idea that the differential productivity of groups can lead to changes in allele frequency in a gene pool, in a manner analogous to individual selection acting through the differential productivity of individuals (Wade, 1978; Wilson, 1980). In this paper we are concerned with type 1 group selection, which defines group fitness as the average fitness of the group members [2] (Okasha, 2006, chap. 2). Models of this process therefore seek to investigate the effect of group structure on the evolution of individual social traits such as cooperation. In particular, a trait that is individually disadvantageous may nevertheless evolve if it has a positive effect on the group as a whole, an idea that dates back to Darwin (1871). Using the language of multilevel selection theory, such cooperative traits are selected against within-group, since they confer a fitness disadvantage relative to other group members, but are favoured under between-group selection, since they increase group productivity, i.e. average absolute group member fitness. Whether or not the trait spreads in the global gene pool then depends on the balance of these two selective forces.

Three factors are pertinent in determining the outcome of such a scenario. The first is the individual cost to group benefit ratio of the cooperative act; a large individual cost strengthens within-group selection against cooperation, while a large group benefit strengthens between-group selection. Secondly, a group mixing mechanism must exist so that the increased productivity of more cooperative groups may affect the global allele frequencies. We consider here a multi-generational variant of D.S. Wilson's (1980) trait-group model, where a global mixing stage occurs every t generations in which the progeny of all groups disperse, join a global migrant pool, and then reform new groups of random composition. Since cooperative groups will contain more individuals prior to dispersal, they will constitute a larger fraction of the global migrant pool, thereby biasing the global allele frequencies. How often global mixing occurs is therefore a crucial parameter, since groups must be mixed before the within-group equilibrium has been reached, otherwise there will be no difference in group productivity for selection to act on (assuming selection within all groups leads to the same group equilibrium (Wilson, 1992)).

The third factor, and the one which is most often commented on in the group selection literature, is the requirement for there to be variation in the groups' allelic composition. The larger this variance, the greater the difference in group productivity (since within-group dynamics are assumed to be deterministic) and likewise the effect of group selection. Many theoretical models have shown that if group composition is random then very small initial group sizes are needed in order to produce the between-group variance necessary for a measurable effect (see (Wade, 1978) for a classical review). It is therefore usually concluded that some kind of non-random group formation is required in order for group selection to have any significant effect. The most common way that such assortative grouping is believed to occur in nature is through kin grouping, where the group members are related by descent from a common ancestor and are hence more similar to each other than to members of other groups. Consequently, kin selection (Hamilton, 1964a,b) is commonly seen as the only pertinent force in social evolution. However, even with only simple random sampling, the range of group sizes which produce a group selection effect may be strongly affected if the assumption of within-group competitive exclusion is relaxed. This is because possible between-group variance from sampling error is controlled by the frequency of the least frequent type in the migrant pool. In the competitive exclusion case the possible variance rapidly tends to zero as the selfish type approaches fixation. If, however, coexistence of the two types is possible at equilibrium then we would expect the possible variance to be maintained above zero even at within-group equilibrium frequencies.

Group Competition: Aggregation and Dispersal Through a Migrant Pool

In this paper, we consider a model of a population structure where individuals reproduce in groups for a number of generations. After this period of reproduction within groups,

[1] Coexistence dynamics may also apply to single-species scenarios under certain regimes such as balancing selection (see below) and are hence relevant to group selection discussions more generally.

[2] Type 2 defines group fitness as number of offspring *groups*.

all groups disperse and their progeny mix freely together in a migrant pool. Thereafter, new groups are created from the individuals in the migrant pool, and the process repeats. This process is a multi-generational variant of the trait-group model of Wilson (1980, 1987), and corresponds closely to the "Haystack" model of Maynard Smith (1964). Examples of natural populations that fit this model particularly well include soil-dwelling populations of micro-organisms that are occasionally mixed together during rainstorms (Wilson, 1980, p. 22), and a type of desert leaf cutting ant which lives in colonies that periodically disband and which are founded by unrelated females from a mating swarm (Rissing et al., 1989). However, the issues explored by the general form of the model are broadly applicable to a wide range of population structures where individuals have the majority of their interactions with a subset of the population. An algorithmic description of such an aggregation and dispersal process is as follows:

1. **Initialisation**: Initialise the migrant pool with N individuals;
2. **Group formation (aggregation)**: Assign individuals in the migrant pool randomly to groups of size S;
3. **Reproduction**: Perform reproduction within groups for t time-steps, as described in the next section;
4. **Migrant pool formation (dispersal)**: Return the progeny of each group to the migrant pool;
5. **Iteration**: Repeat from step 2 onwards for a number of generations, T.

Group competition occurs in a population structured in this fashion during the dispersal stage, since groups that have grown to a larger size will contribute more individuals to the migrant pool. Crucially, this means that cooperative traits that are individually disadvantageous but that benefit the group can potentially increase in frequency (see Figure 1), depending on the balance of within- and between-group selection.

A key factor in determining the balance between the levels of selection in any model of group selection is the variance in group composition (Wilson, 1980), for if there is no variance then there is nothing for selection to act on (Darwin, 1859). In particular, there must be variance in initial group composition which causes a variance in group size prior to dispersal. In the above model, this variance is generated through random sampling of individuals from the migrant pool, where the sample size corresponds to the initial group size. Since an increased sample size causes a decrease in between-sample variance, increasing the initial group size decreases between-group variance and hence the efficacy of group selection (Wade, 1978). It is therefore often assumed that the upper limits of group size which produce a non-zero group selection effect are very small.

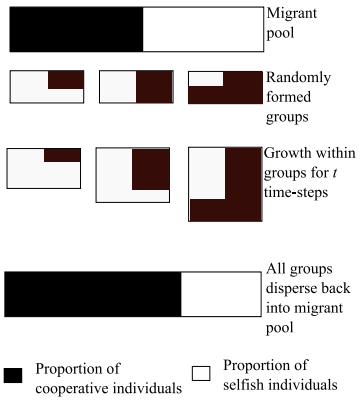


Figure 1: Cooperation can increase in frequency in the migrant pool due to differential group contributions, even though it decreases in frequency within each group.

However, in this paper we present results which suggest that this follows from an assumption of within-group dynamics that lead to the competitive exclusion of a cooperative type by its selfish counterpart at within-group equilibrium. Specifically, we are able to show that where within-group dynamics instead lead to a stable coexistence of types, then the range of initial group sizes over which an effect of group selection can be seen is much larger. This is due to the fact that since the cooperative type cannot be driven extinct, variance in group composition when sampling from the migrant pool is always possible. In the next section, we describe how both competitive exclusion and coexistence dynamics can be modelled within groups.

Within-group Dynamics: Competitive Exclusion Versus Coexistence

Classical models of group selection consider a scenario where a selfish type ultimately drives its cooperative counterpart extinct at within-group equilibrium. In particular, fitness functions of the following form, first proposed by Wright (1945) but subsequently used in a plethora of other models (Williams and Williams, 1957; Maynard Smith, 1964; Charnov and Krebs, 1975; Wilson, 1980, 1987), are typically used to model within-group selection:

$$f_s = 1 + p_c g \quad (1)$$

$$f_c = (1 + p_c g)(1 - a) \quad (2)$$

In the above equations, f_s and f_c denote the per capita fitness of selfish and cooperative individuals within a group, respectively. Cooperators, whose proportion within the group is denoted by p_c, confer a fitness benefit g on every group member. Crucially, both types receive this benefit, while only cooperators pay a cost, represented by the selection coefficient against cooperation, a. It is then clear that if

these equations are iterated until equilibrium is reached then the selfish type will be driven to fixation within the group.

Both competitive exclusion and stable coexistence within-group dynamics can instead be modelled using the standard two-species symmetric Lotka-Volterra competition equations (e.g. (May, 1976)). For implementation purposes, we use the following difference equation as a discrete approximation:

$$N_{i(t+1)} = N_{i(t)} + \left[M_i N_i \left(\frac{K_i - \alpha_{ii} N_i - \alpha_{ij} N_j}{K_i} \right) \right]. \quad (3)$$

In the above equation, $N_{i(t)}$ is the biomass of species i at time t, and M_i the intrinsic per capita growth rate. Each species has an intrinsic carrying capacity K_i, which is then modified through interspecific interactions. Specifically, the per capita effect of species j on species i is given by α_{ij}, the coefficient of interaction. All such interactions are competitive in the above equation, as ensured by the negative sign and the stipulation that all $\alpha > 0$. Similarly, α_{ii} denotes the negative density-dependant effect of species i on itself that prevents unbounded exponential growth. This coefficient can be seen as representing crowding and can vary for different species.

We define selfish (s) and cooperative (c) strategies in the above equation through settings of the within- and between-type interaction coefficients. Specifically, a selfish type is defined as having a large negative per capita effect on both itself (α_{ss}) and the other type (α_{cs}). A cooperative type is then defined as having correspondingly smaller per capita negative effects (α_{cc} and α_{sc}). A pure group of cooperators will therefore grow to a larger size than a pure group of selfish individuals, creating a group productivity differential on which selection can potentially act. However, within mixed-groups the selfish type will reach the larger frequency (provided that α_{ss} is not too large), since $\alpha_{cs} > \alpha_{sc}$. In other words, cooperators are favoured by between-group selection, while selfish individuals are favoured under within-group selection, exactly as in a classic group selection scenario. It should be noted that our definition of cooperative behaviour corresponds to weak, rather than strong, altruism (Wilson, 1980). This follows because although cooperation confers a relative fitness disadvantage compared to a selfish individual within the same group, it nevertheless increases the absolute fitness of all group members, including the cooperator.

It is well known that a stable coexistence of both types occurs in such a model when competition for resources (space, food, etc.) between individuals of the *same* type is stronger than competition between individuals of different types. Such a case corresponds at the ecological level to species occupying different niches, i.e. only partially overlapping in their resource requirements (May, 1976). Conversely, if between-type competition is stronger than within-type competition then competitive exclusion of one type will occur. Between- and within-type competition are both modelled in the Lotka-Volterra equations through the settings of the interaction coefficients. Throughout this paper, we assume the following:

1. $\alpha_{cc} < \alpha_{ss}$ and $\alpha_{sc} < \alpha_{cs}$, i.e. that cooperators have lower negative density-dependant effects on themselves and others;
2. $\alpha_{cs}\alpha_{sc} \leq 1$;
3. M and K, the intrinsic per capita growth rates and carrying capacities respectively, are the same for both types.

Given these assumptions, competitive exclusion of the cooperative type occurs when $\alpha_{ss} < \alpha_{cs}$, producing qualitatively similar dynamics to those of the traditional within-group selection equations (1) and (2). However, when $\alpha_{ss} > \alpha_{cs}$ then the cooperative type is maintained at within-group equilibrium at an above-zero frequency, i.e. a stable coexistence of types occurs. When the interaction coefficients of the cooperative strategy are fixed, the equilibrium frequency at which it is maintained then depends upon the settings of α_{ss} and α_{cs}, i.e. the magnitude of the negative density-dependant effects of selfish individuals on themselves and cooperators, respectively. In addition, the within-group equilibrium is reached more quickly the greater the effects of the selfish type. Although a Lotka-Volterra model such as this is typically interpreted at the ecological level as representing species interactions, it could also be interpreted as a model of allelic competition dynamics within a single species group. In particular, coexistence Lotka-Volterra dynamics are analogous to balancing selection for a stable allelic polymorphism within a group. Conversely, competitive exclusion of one species by another is analogous to directional selection driving one allele to fixation. The motivation for using the language of allelic competition is to facilitate comparison with classical group selection models, which consider competition between selfish and cooperative alleles.

Our use of the Lotka-Volterra equations in this paper should be contrasted from their use in community or ecosystem selection models (Wilson, 1992; Penn, 2003). Such models do not consider explicit cooperative and selfish types in the fashion of traditional group selection models. Instead, they examine the complex within-group dynamics that arise when a larger number of types are present. These complex dynamics can give rise to multiple within-group attractors, which can then provide a source of variation in their own right upon which selection can act (Penn, 2003). By contrast, in this paper we consider simple two-type within-group dynamics, where only a single group attractor exists (either coexistence or competitive exclusion, as discussed above). As far as we are aware, our use of the Lotka-Volterra equa-

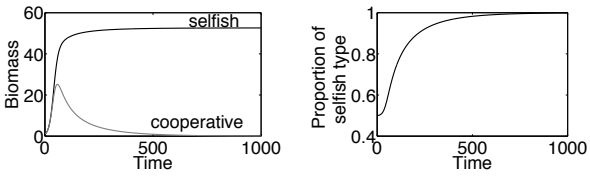
tions to define explicit selfish and cooperative strategies is novel.

Results

The parameter settings used for the Lotka-Volterra equations throughout this paper are shown in Table 1. Changing between competitive exclusion and coexistence within-group dynamics is achieved by simply switching over the values of α_{ss} and α_{cs}, since that determines whether $\alpha_{ss} < \alpha_{cs}$ and hence whether competitive exclusion occurs. The values of the interaction coefficients in Table 1 produce representative within- and between-group dynamics; other settings produce the same qualitative trends. In this section, we first present results using classical competitive exclusion dynamics, and then contrast these to results from the coexistence case.

Group Selection Dynamics in the Competitive Exclusion Case

The within-group dynamics for a group initialised with unit biomass of each type are shown in Figures 2(a) and 2(b), for the competitive exclusion case. Initially, both types are in their growth phase; their biomass is below the intrinsic type carrying capacity of 100. However, the selfish type grows at a faster rate, despite the fact that their intrinsic growth rates, M, are the same. This is because of the greater negative density-dependant effect of the selfish type on cooperators, i.e. $\alpha_{cs} > \alpha_{sc}$. Finally, since $\alpha_{cs} > \alpha_{ss}$, the cooperative type is driven to extinction. Furthermore, as Figure 2(b) shows, the proportion of selfish individuals increases monotonically. Such behaviour is qualitatively identical to that of directional within-group selection for a selfish allele in classical group selection models (e.g. (Wright, 1945; Wilson, 1980)).



(a) Biomass of each type.

(b) Proportion of selfish type.

Figure 2: Competitive exclusion within-group dynamics.

Now let us consider global dynamics under group selection in this competitive exclusion case. In order for group selection to operate through an aggregation and dispersal process, a difference in group size at the dispersal stage must exist. Figure 3 illustrates how final group size varies as a function of the time spent in the group prior to dispersal, for various starting frequencies of cooperators in groups of initial size 10. It can be seen from this graph that, using the

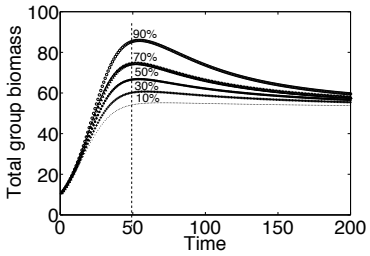


Figure 3: Final group size as a function of time spent reproducing within groups; initial group size 10 with various % of cooperators (competitive exclusion case shown; coexistence quantitatively similar). Dotted line shows time at which difference in group productivity is greatest.

parameters described in Table 1, groups with a greater proportion of cooperators do indeed grow to a larger size. In addition, the results for the coexistence case, where α_{ss} and α_{cs} are swapped, are quantitatively similar. These results therefore confirm that group selection can in principal operate, since there is a variation in group productivity on which selection can act.

To determine the magnitude of the effect of group selection, the aggregation and dispersal process was executed for 5000 iterations, which preliminary experimentation had shown to be a sufficient length of time for a global equilibrium to be reached, using the within-group parameter settings described in Table 1. Equation 3 was iterated 30 times in the reproduction stage, while the global population size was maintained at 5000. Initial group size was then varied from 1 to 100 inclusive, while the migrant pool was initialised with 50% of each type. The result of this process after 5000 aggregation and dispersal cycles is shown in Figure 4, where 'effect of group selection' on the y-axis is defined as the difference between the frequency of the selfish type at within-group equilibrium and the global frequency of the selfish type after 5000 aggregation and dispersal cycles. Since the within-group equilibrium in the competitive exclusion case is the selfish type at 100%, the y-axis equivilantely shows the global frequency of cooperators in this case. Furthermore, it should also be stressed that the within-group equilibrium is the equilibrium that would be reached in an unstructured population where there were no groups. The y-axis therefore shows the effect that group structure is having on the outcome of evolution compared to that in an unstructured population.

There are two points to note from Figure 4. Firstly, increasing the initial group size decreases the effect of group selection, and consequently the global proportion of cooperators. In particular, for small group sizes, the cooperative type reaches global fixation (and remains there because we do not reintroduce types by mutation). However, for group sizes above 10, it is driven extinct. This follows from the

Parameter	Value (competitive exclusion)	Value (coexistance)
α_{ss}	**1.9**	**2**
α_{cs}	**2**	**1.9**
α_{cc}	1	1
α_{sc}	0.5	0.5
K	100	100
M	0.1	0.1

Table 1: Parameter settings of the Lotka-Volterra equation. Note that the only difference between the competitive exclusion and coexistence settings is a swapping of the values of α_{ss} and α_{cs}.

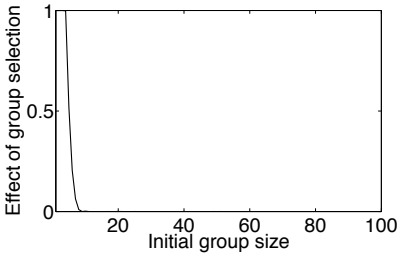
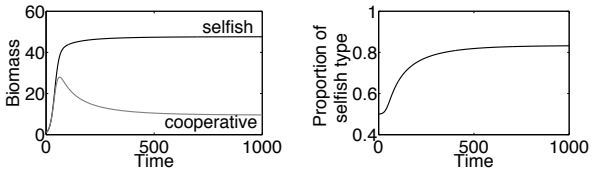


Figure 4: 'Effect of group selection' (see text) as a function of initial group size in the competitive exclusion case.



(a) Biomass of each type.

(b) Proportion of selfish type.

Figure 5: Coexistence within-group dynamics.

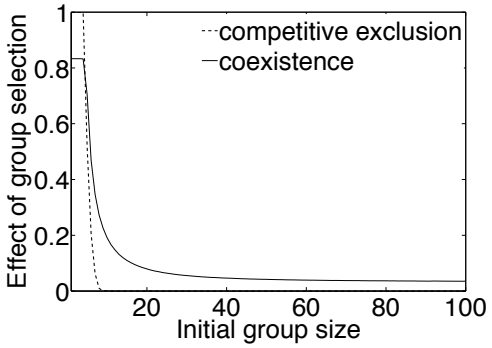


Figure 6: Comparing the range of group sizes over which an 'effect of group selection' (see text) can be seen between coexistence and competitive exclusion dynamics.

fact that the between-group variance necessary for group selection to act is generated by random sampling from the migrant pool, and therefore rests on the existence of a small initial group size, as previously discussed.

The second, and a key point for this paper, is that the effect of group selection rapidly tends to zero as initial group size increases. Specifically, above a group size of 10, there is no measurable effect at all. Such a result may therefore make the idea of group selection acting on randomly formed groups seem rather implausible as a significant evolutionary pathway. However, the above results only consider the competitive exclusion case; in the coexistence case, the results are somewhat different, as shown in the following section.

The Efficacy of Group Selection under Coexistence Dynamics

Let us now consider the coexistence dynamics that arise from redefining the selfish strategy as $\alpha_{ss} = 2$ and $\alpha_{cs} = 1.9$, i.e. by swapping the interaction coefficients over from the competitive exclusion case. Figures 5(a) and 5(b) show how the cooperative type is no longer driven extinct at the within-group equilibrium. In particular, the change in the frequency of the selfish type from an initialisation of 50% shows clear balancing selective dynamics resulting in the maintance of cooperation at an above-zero frequency. In other words, the result is a stable coexistence of cooperative and selfish types within a group.

Group selection dynamics under the aggregation and dispersal process are now as shown in the black curve in Figure 6. Crucially, in contrast to the competitive exclusion case (shown again in the dotted line), an effect of group selection can be seen over the entire range of group sizes examined. For example, in groups of initial size 50, group selection can be seen to still increase the global frequency of cooperation above the within-group equilibrium. The significance of this observation is that since the within-group equilibrium is the same equilibrium that would be reached in an unstructured population, these results show that group structure is having an effect on population dynamics across a wide range of group sizes.

Finally, to verify that this result is not an artefact of the particular values of α_{ss} and α_{cs} used, the same curves were plotted for a variety of other parameters. Figure 7 provides

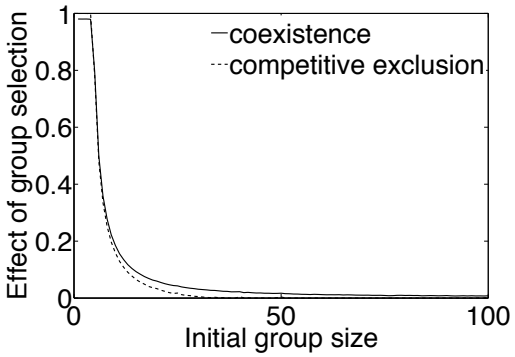


Figure 7: Demonstrating that the same qualitative trends arise where within-group selection towards selfish behaviour is stronger in the coexistence case than in Figure 6. Here, $\alpha_{cs} = 1.99$ and $\alpha_{ss} = 2$ in the coexistence case, vice versa for competitive exclusion; all other parameters as in Table 1.

an example of this, where $\alpha_{ss} = 1.99$ and $\alpha_{cs} = 2$ in the competitive exclusion case, vice versa for the coexistence case. These parameters were chosen since they represent stronger within-group selection towards selfish behaviour in the coexistence case than in the previous example. Specifically, the within-group equilibrium frequency of the selfish type in the coexistence case is 98.04%, compared to 83.3% previously. The results in Figure 7 show that while an effect of group selection is still seen over a larger range of group sizes in the coexistence case, the magnitude of the effect is reduced compared to Figure 6. The reason for this is that variance in group composition is proportional to the frequency of cooperators in the migrant pool in this case, and hence to the corresponding within-group equilibrium frequency, as discussed in detail in the following section.

Discussion

The results in the previous section demonstrate that where a stable coexistence of types occurs at within-group equilibrium, an effect of group selection on global frequencies can be seen over a much larger range of initial group sizes than in the competitive exclusion case. In particular, as group size increases in the competitive exclusion case, any measurable effect of group selection on the global frequency of cooperation rapidly tends to zero. By contrast, in the coexistence case, some effect on global frequencies is seen over the entire range of group sizes examined. It must be stressed that we do not make a particular claim about the magnitude of the effect for large group sizes. Rather, our model implies that there is *some* measurable effect on frequencies over a large range of group sizes; how large this effect may be will depend on the properties of the natural system under consideration. However, the fact that any effect of group selection still exists over a large range of parameters is significant, since it suggests that where within-group dynamics in nature are of the coexistence type, some effect of a group population structure may always be acting.

Coexistence dynamics allow group selection effects to be sustained over a larger range of group sizes because of the effect of migrant pool frequencies on between-group variance. In particular, because group formation constitutes random sampling from the migrant pool, initial between-group variance can be approximated by the binomial distribution, and is then given by $p_c p_s / S$, where p_c is the proportion of the cooperative type in the migrant pool, p_s the proportion of the selfish type, and S the initial group size (Wilson, 1980, p. 27). Since $p_c + p_s = 1$, it follows that between-group variance is proportional to the frequency of the least frequent type, i.e. variance is maximal when both types are of equal frequency, and zero when one type is at fixation. Therefore, where one type reaches global fixation then there can be no variance and hence no group selection. However, in the coexistence case, where one type cannot reach fixation, it follows that there must always be some variance and hence some possible effect of group selection. The fact that the variance is proportional to the frequency of the least frequent type is illustrated by the difference between Figures 6 and 7, where the lower within-group equilibrium frequency of cooperators in Figure 7 results in a reduced effect of group selection for large group sizes.

A further observation from Figure 6 is that a gradient towards an increased effect of group selection also exists over a larger range of group sizes in the coexistence case. Specifically, decreasing group size by a small amount yields an increase in the effect of group selection for groups of size 20 in the coexistence case. However, there is no gradient at this size in the competitive exclusion case. The significance of this is that increasing the effect of group selection increases average absolute individual fitness in the population, due to an increased global level of cooperation. If group size can be partly determined by individual traits (Powers et al., 2007) then this may provide an adaptive gradient towards smaller groups, increased levels of cooperation, and greater fitness. In the competitive exclusion case, however, such a gradient only exists over a much smaller range of group sizes. While numerical experimentation investigating whether either gradient can be followed by a series of small mutations will be the subject of a future study, the results presented here do suggest that the concurrent evolution of group size and cooperation is more plausible in cases where a stable coexistence of types within groups exists.

Conclusions

Any group selection process requires there to be a variation in group composition. In aggregation and dispersal style models, this variation arises through the random assignment of individuals from the migrant pool into groups. Consequently, it is often suggested that an effect of group selection on the global frequency of types will only be seen for very

small initial group sizes. However, the models on which this claim is based typically only consider within-group dynamics that lead to the competitive exclusion of a cooperative type by its selfish counterpart.

In this paper, we suggest that within-group competitive exclusion dynamics may be an unnecessary assumption in a number of situations, including the modelling of multi-species consortia in bacterial biofilms and during egalitarian major transitions. A model has been presented which shows that where such coexistence dynamics are present within groups, the range of initial group sizes over which an effect of group selection can be seen is much larger. Consequently, the potential for an adaptive gradient towards smaller groups and increased cooperation exists over a much larger range of group sizes under coexistence dynamics. This is in sharp contrast to the competitive exclusion case, where the effects of group selection rapidly reach zero as initial group size increases, excluding the possibility of such a gradient for a large range of parameters. Our results suggest that, where group size can be influenced by individual traits, the evolution of smaller groups and increased cooperation is more plausible under coexistence dynamics. Such increased group cooperation is a vital component of many major transitions in evolution (Maynard Smith and Szathmary, 1995; Michod, 1999).

We have shown in this paper that the conventional conclusion that group selection effects can only be seen for very small groups rests on the assumption that within-group dynamics lead to competitive exclusion. If a within-group coexistence of competing types is instead permitted, then the range of group sizes over which an effect can be seen is much larger. This result follows from the fact that the variance in group composition upon which group selection acts is dependant not only on group size but also on the frequencies of types in the migrant pool. In particular, since neither type can be driven extinct under coexistence dynamics, there will always be some variance in group composition when sampling from the migrant pool, which can then be acted on by group selection. Thus, since it is not necessary to assume that within-group dynamics lead to competitive exclusion, this result shows that group selection can operate in a wider range of conditions than previously realised.

References

Bedau, M., McCaskill, J., Packard, N., Rasmussen, S., Adami, C., Green, D., Ikegami, T., Kaneko, K., and Ray, T. (2000). Open questions in artificial life. *Artificial Life*, 6:352–357.

Burmolle, M., Webb, J. S., Rao, D., Hansen, L. H., Sorensen, S. J., and Kjelleberg, S. (2006). Enhanced biofilm formation and increased resistance towards antimicrobial agents and bacterial invasion caused by synergistic interactions in multi-species biofilms. *Appl Environ Microbiol*, 72:3916–23.

Charnov, E. L. and Krebs, J. R. (1975). The evolution of alarm calls: Altruism or manipulation? *The American Naturalist*, 109(965):107–112.

Darwin, C. (1859). *On the origin of species by means of natural selection, or the preservation of favoured races in the struggle for life*. John Murray, London.

Darwin, C. (1871). *The Descent of Man, and Selection in Relation to Sex*. John Murray, London.

Hamilton, W. D. (1964a). The genetical evolution of social behaviour. i. *Journal of Theoretical Biology*, 7(1):1–16.

Hamilton, W. D. (1964b). The genetical evolution of social behaviour. ii. *Journal of Theoretical Biology*, 7(1):17–52.

Hammerstein, P. (2003). *Genetic and Cultural Evolution of Cooperation*. MIT Press.

Keller, L., editor (1999). *Levels of Selection in Evolution*. Monographs in behavior and ecology. Princeton University Press.

May, R. M., editor (1976). *Theoretical Ecology: Principles and Applications*. Blackwell Scientific Publications, Oxford.

Maynard Smith, J. (1964). Group selection and kin selection. *Nature*, 201:1145–1147.

Maynard Smith, J. and Szathmary, E. (1995). *Major Transitions in Evolution*. Spektrum.

Michod, R. E. (1999). *Darwinian Dynamics: Evolutionary Transitions in Fitness and Individuality*. Princeton University Press.

Odling-Smee, F. J., Laland, K. N., and Feldman, M. W. (2003). *Niche construction: the neglected process in evolution*. Monographs in population biology; no. 37. Princeton University Press.

Okasha, S. (2006). *Evolution and the Levels of Selection*. Clarendon Press.

Penn, A. S. (2003). Modelling artificial ecosystem selection: A preliminary investigation. In *Proceedings of the 7th European Conference on Artificial Life (ECAL 2003)*.

Powers, S. T., Penn, A. S., and Watson, R. A. (2007). Individual selection for cooperative group formation. In *Proceedings of the 9th European Conference on Artificial Life (ECAL 2007)*.

Rissing, S. W., Pollock, G. B., Higgins, M. R., Hagen, R. H., and Smith, D. R. (1989). Foraging specialization without relatedness or dominance among co-founding ant queens. *Nature*, 338:420–422.

Wade, M. J. (1978). A critical review of the models of group selection. *The Quarterly Review of Biology*, 53:101–114.

Williams, G. C. and Williams, D. C. (1957). Natural selection of individually harmful social adaptations among sibs with special reference to social insects. *Evolution*, 11:32–39.

Wilson, D. S. (1980). *The Natural Selection of Populations and Communities*. Benjamin/Cummings, California.

Wilson, D. S. (1987). Altruism in mendelian populations derived from sibling groups: The haystack model revisited. *Evolution*, 41(5):1059–1070.

Wilson, D. S. (1992). Complex interactions in metacommunities, with implications for biodiversity and higher levels of selection. *Ecology*, 73(6):1984–2000.

Wright, S. (1945). Tempo and mode in evolution: A critical review. *Ecology*, 26:415–419.

Mechanism as Mind: What Tensegrities and Caterpillars Can Teach Us about Soft Robotics

John Rieffel[1] , Barry Trimmer[1] and Hod Lipson [2]

[1] Biomimetic Devices Laboratory, Tufts University, Medford MA
[2] Computational Synthesis Laboratory, Cornell University, Ithaca MA
john.rieffel@tufts.edu

Abstract

With recent advances in materials, interest is being applied to the idea of robots with few if any rigid parts, able to substantially deform themselves in order to flow around, and even through objects. In order to accomplish these goals in an efficient and affordable manner, space and power will be at a premium, and so soft robots will most likely be both under-actuated and under-controlled. One approach to actuation and control lies in embodying portions of both tasks within the structural dynamics of the robot itself. Such "morphological computation" is known to exist throughout the biological world, from the behavior of cellular cytoskeletons up to the tendinous network of the human hand. Here we present two examples of morphological computation - one from biology, the *manduca sexta* caterpillar, and one from engineering, a modular tensegrity tower - and explore how ideas from these realms can be applied toward locomotion and control of a highly articulate, under-controlled, soft robot.

Introduction

Imagine a robot that can squeeze through holes, climb up walls, and flow around obstacles. Once the domain of science fiction, thanks to modern advances in materials such as polymers (Huang et al., 2007), and nanocomposites (Capadona et al., 2008) such a "soft robot" is becoming an increasing possibility. This ability to significantly deform and alter shape, at a much higher level of detail than discrete "modular" robots (such as Yim's Polybot (2000) and Rus's Molecubes (1998)) makes accessable new and increasingly important environments such as mine fields and collapsed buildings.

However, this incredible flexibility and deformability brings with it considerable constraints in terms of actuation, power, and payload. Striving not to have any rigid or fully solid elements means that servo motors and batteries - the bread and butter, so to speak, of conventional robotics - are far from ideal. As a consequence, soft robots will in all likelihood be under-actuated, under-powered, and under-controlled. Therefore, the onus lies upon soft robotics researchers to discover ways of controlling these highly articulate systems.

Fortunately, nature itself has provided us with several quite viable prototypes, among them many of the large invertebrates such as the octopus, squid, and the *manduca sexta* caterpillar, the latter of which we will draw particular inspiration from in this work. As we discuss in detail below, the *manduca* is able to achieve incredible flexibility and control, despite having relatively few muscles and astonishingly few motoneurons in each of its segments. It is conjectured therefore that, in *manduca*, the interaction of hydrostatics, body wall tension, and muscles, all contribute to a degree of neuromechanics, or *morphological computation* (Trimmer, 2007). That is to say, a large amount of the work normally attributed to the neural system is instead "outsourced" and embodied directly into the mechanics of the structure. Similar types of morphological computation have been observed in other biological systems, such as the tendonous network of the human hand (Valero-Cuevas et al., 2007), wallabies Biewener et al. (2004), and cockroaches Ahn and R.J.Full (2002).

In this paper we review some details of *manduca*'s anatomy and locomotion as it pertains to morphological computation. We then present related work in which a highly complex mechanical system – a tensegrity structure – is able to achieve locomotion by exploiting the dynamical coupling between modules as an emergent data bus. Finally, we bring these aspects together when describing the design and control of a completely soft robot modeled loosely on the *manduca*. More broadly, we hope to present morphological computation – the use of mechanism as mind – as the best approach to solving the issues of actuation and control inherent in soft robotics.

The *Manduca Sexta* Caterpillar: a Model Species for Soft Robotics?

Caterpillars such as *manduca sexta* (**Figure 1**) are some of the most successful climbing herbivores in the world. They are incredibly flexible, with a large multi-dimensional workspace, are able to cantilever themselves over gaps up to 90% of their length, and perform a u-turn inside spaccesless than twice their body diameter(Trimmer, 2007). They are able to accomplish all of this because of, and at the same time despite of, the fact that they are completely soft-bodied and lack any rigid elements such as a skeleton. Unlike the beam-and-lever mechanics of vertebrates, *manduca* move through a complex dynamical interplay of hydraulics, body wall tension, and muscles.

Most remarkably, they are able to accomplish all of these complex behaviors despite a relatively simple anatomy. Most locomotion is performed by the co-ordination of their abdominal segments, each of which contains on the order of 70 muscles. Furthermore, each such segment contains relatively few motoneurons (one, or maximally two per muscle), and no inhibitory motor units (Trimmer, 2007). **Figure 2** contains an illustration of the major muscles within a single such segment. These muscles however, are in of themselves rather complex, exhibiting nonlinear and pseudo-elastic responses to load cycling (**Figure 3**) which are quite different than vertebrate muscles (Dorfmann et al., 2007).

All of these properties come into play when observing the crawling kinematics of the animal (**Figure 4**). Under normal locomotion, waves of motion pass from the rear segment of the animal

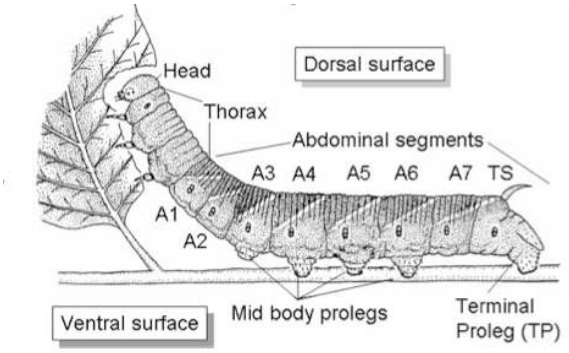


Figure 1: The external anatomy of the *manduca sexta* caterpillar. (adapted from Mezoff et al. (2004))

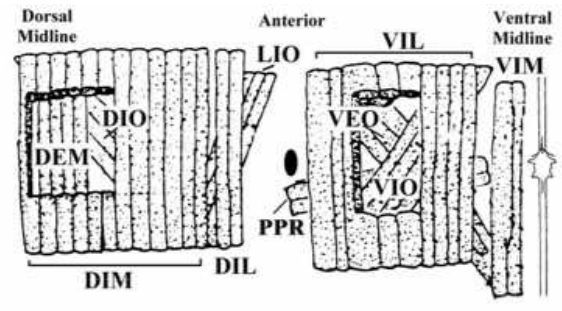


Figure 2: The major muscles from one side of an abdominal segment of manduca (from Levine and Truman (1985))

(TS) toward the head. As the wave propagates, each segment compresses then re-extends, with the dorsal and ventral parts remaining in phase. Somewhat puzzlingly, the length and radius of each segment co-vary - they narrow and shorten simultaneously . This suggests that fluid volume is not conserved during crawling - rather tissue and fluid are moved and compressed throughout the animal during locomotion (Trimmer and Issberner, 2007).

Combined, these complex mechanical properties, and the limited neural control, lead to the conclusion that the dynamics of the system is itself responsible for control tasks that would otherwise be attributed to neural circuitry. We explore and describe how a complex mechanical structure can exploit this kind of coupled dynamics in order to achieve co-ordinated motion in the following section.

Morphological Communication in Tensegrity Robots

Traditional engineering approaches strive to avoid, or actively suppress, the kind of nonlinear dynamic coupling among components exhibited in the anatomy of the *manduca*. Especially near resonant frequencies, these couplings tend to produce undesirable vibrations and oscillations that are difficult to predict and may sometimes be catastrophic. A variety of passive and active damping techniques have been developed to diminish these effect across many fields ranging from robotics to structural engineering.

Biological systems, by contrast, are often rife with complex dynamics. Beyond the examples from *manduca*, consider the principle of tensegrity, which can be found at many scales of life, ranging from the cellular cytoskeleton and the structure of proteins (Ingber,

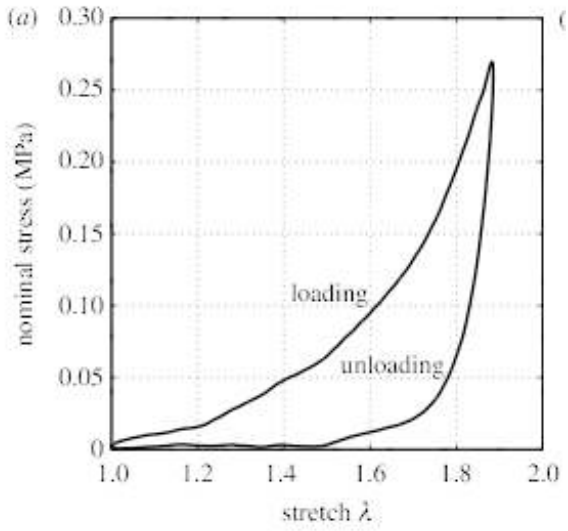


Figure 3: The pseudo-elastic response of a manduca muscle (from Dorfmann et al. (2007)). The muscle exhibits a large degree of non-linear pseudo-elasticity when subjected to load cycling.

.

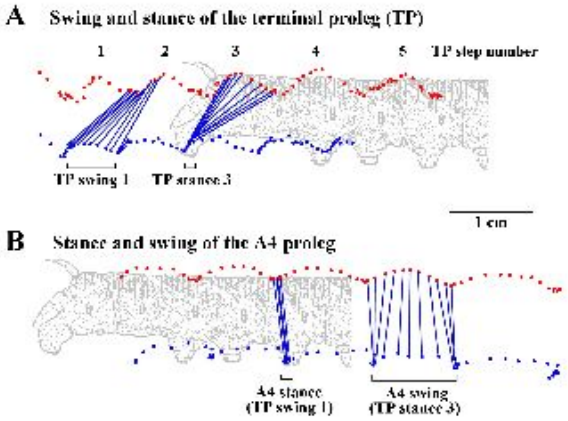


Figure 4: Movement of the *manduca* terminal and midbody segments during crawling (from Trimmer and Issberner (2007)).

1998) to the tendinous network of the human hand (Valero-Cuevas et al., 2007). At every scale, these systems contain the type of coupled mechanical and dynamical linkages which are so assiduously avoided in engineering design. Could there be in fact, an advantage to such high degrees of dynamical coupling?

The word tensegrity, a concatenation of tensile integrity was coined by Buckminster Fuller to describe structures first popularized by the sculptor Kenneth Snelson in 1948 (Fuller, 1975). Broadly speaking, a tensegrity structure is a set of disjoint rigid elements (rods) whose endpoints are connected by tensile elements (strings), and which maintains its shape due to the synergy between the compressive forces on its rods and the complementary forces in its cables. Such structures are pre-stress stable, in the sense that in equilibrium each rigid element is under pure compression and each tensile element is under pure tension. The structure therefore has a tendency to return to its stable configuration after subjected to any moderate temporary perturbation (Connelly and Back, 1998; Motro, 2003).

Unfortunately, these qualities which make tensegrities so attractive, largely pre-stress stability, carry with them complex nonlinear dynamics, even for relatively small tensegrity structures (Skelton et al., 2001), and as a result, active control is needed to dampen the vibrational modes of relatively modest structures. In almost all cases, deformation and control are achieved by changing the rest lengths of the tensile elements, for instance by attaching attaching strings to a reeled servo motor. In this manner, Skelton *et al*. have been able to demonstrate both active vibration damping (2004) and open-loop control of simple structures. Efforts such as these, however, seek to minimize and control the complex dynamics of tensegrity structures, and no effective model exists for the control of the complex dynamics of relatively large tensegrity structures. Rather than attempting to scale these control schemes to arbitrarily large and complex structures, our interest, by contrast, lies in harnessing and exploiting these dynamics in the same way that biological systems seem to.

Modular Tensegrity Robotics

Constructing robots from tensegrities is a double-edged sword. On one hand the homogeneity of the rigid elements allows for a high degree of modularity: each rod can contain identical sets of sensors and actuators – the parts of a 10-bar tensegrity are identical to those of a 3-bar one. On the other hand, any solution which relies upon centralized control of the robot faces a crucial problem: that of communication between modules. As the number of modules increases, the lines of communication (quite literally) increase, bringing both the challenge of coordination and the risk of tangles. Consider, for instance, the tensegrity shown in **Figure 6**. Even with a single sensor and actuator at each end of each bar, a centralized controller would need to synthesize, and co-ordinate the actions of thirty sensors and thirty controllers.

We implement a simpler alternative to the problem of control and locomotion by doing away with the notion of explicit inter-modular communication completely. In our model we consider each rod of the tensegrity to be a simple module with a small controller capable only of sensing, and affecting the tension on a single string at each end. Each strut module consists of a rigid tube with a single servo motor mounted at each end. While, in principle, multiple strings could be actuated by multiple servos at each end, we have chosen to keep the design simple by limiting actuation on each end to a single string. **Figure 5** contains a photograph of a representative tensegrity robot which contains four strut modules

In order to add time sensitivity we use a variant of ANNs called spiking neural networks. Spiking neural networks (SNNs) were developed to model more continuous processes: input and outputs

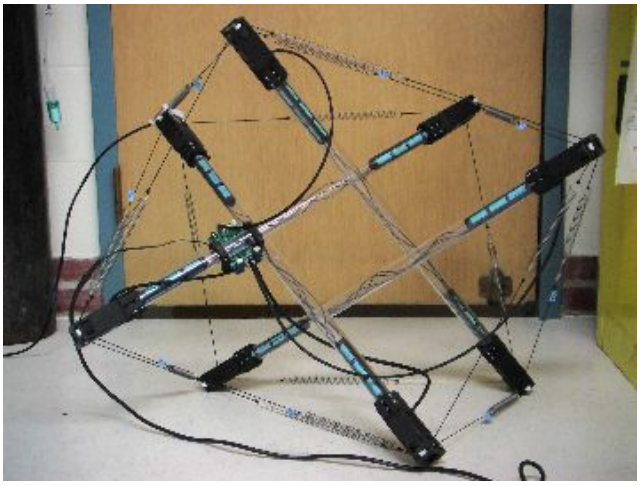


Figure 5: A tensegrity robot consisting of four strut modules and 16 strings

are both represented as single-value spikes (as opposed the sigmoid outputs of a conventional ANN) (Maass and Bishop, 1999). Instead of a sigmoid function, every SNN node contains a simple persistent counter, with adjustable offset and limit. At every time step, an SNN node sums its weighted inputs with the current counter value, and if the sum surpasses the limit the node fires a single "spike" to its output; otherwise the contents of the counter are decremented by a fixed decay rate, and persist until the next time step.

Each strut module in our tensegrity robot contains a single spiking neural network with two inputs, corresponding to the tension sensed at the single actuated string on each end, two hidden nodes, and two outputs. At every simulation time step, each module measures its inputs and feed them through the SNN. Output spikes are converted into string actuations by measuring the duty cycle of network spikes. Any spike rate above 30% over a 100 step period is considered "active", and the corresponding string is pulled by halving its rest length. Our choice of relatively simple binary actuation in this regard is an effort to simplify overall control, and to reduce the difficulty in translating simulated results into physical servo values.

Evolving Dynamic Gaits

In order to evolve gaits for tensegrity robots, the 15-bar tensegrity shown in **Figure 6** was reproduced within the Open Dynamics Engine (ODE) Simulation environment , the widely used open-source physics engine which provides high-performance simulations of 3D rigid body dynamics. Rigid elements were represented as solid capped cylinders of fixed length with a length-to-radius ratio of 24:1. Tensile elements were represented as spring-like forces acting upon the cylinder ends.

With only 30 actuators available (one at the end of each strut module), and a choice of 78 strings to actuate, we chose to evolve both the unique weights of the SNN within each strut module, and also which particular string at each end to actuate. Genotypes of individuals within the population therefore consisted of two subgenes. The first contained 180 floating point numbers corresponding to the collective weights of all 15 strut module controllers within the structure. The second consisted of a pairing of actuated strings with strut endpoints. A single point mutation could therefore either change a weight within the SNN or change which string was actuated at a particular endpoint.

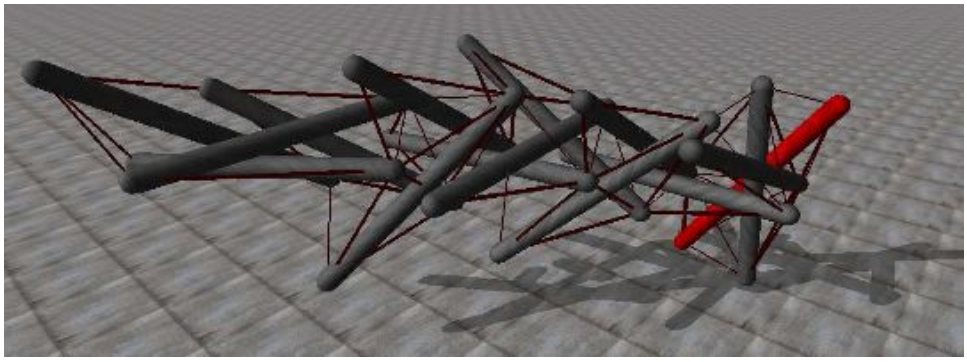


Figure 6: A complex and highly dynamically coupled fifteen-bar tensegrity structure

Using this framework, we were able to evolve the weights within the separate SNNs such that the structure as a whole was able to locomote. Each experiment consisted of a population of 150 individuals initialized with random SNN weights evolved over the course of 1000 generations. Individuals were evaluated within our simulated environment by measuring the travel of the center of mass over the course of 20,000 simulator time steps. Members of the population were then ranked by their fitness, and the bottom scoring half of the population culled. 75 new individuals were then created as offspring of the remaining population via fitness proportional selection, in which 30% of offspring were produced with two-parent crossover, and the remainder with single-point single-parent mutation.

Figure 9 shows the string activations of one successful evolved gait during a single gait cycle, and **Figure 7** contains snapshots of the movement of the ensuing locomotion. The path of the red sphere above the structure tracks the center of mass of the structure (vertically displaced for visualization). It is worth noting that this locomotion is accomplished despite the fact that the activity of the strings shown in Figure 9 is so low. The movement of the entire structure is, in fact, caused largely by the oscillation of just two of its 78 strings. This provides some indication of the extent to which the gait is exploiting the dynamics of the tensegrity itself, and its vibrational modes.

We can further qualitatively measure the coupling between evolved gait and system dynamics by observing the behavior of the structure when the speed of the evolved gait is adjusted while keeping the dynamics of the system unchanged. As shown in Figure 8 both the distance traveled *and the path traversed* vary significantly under varying speeds.

Toward A Soft Robot

Our aim is to create a completely soft, articulate and deformable robot modeled and inspired by the *manduca*. Like manduca (which grows 10000-fold without any changes in musculature or nervous system), we hope to arrive at a highly scalable solution - changing materials and actuators as necessary, while maintaining highly similar control schemes. The constraints any such system which strives to be fully soft means that space and power for actuation will be at a premium. It is clear that, much like the biological system, our soft robot must leverage every aspect of its morphology in order to offload what are normally considered computational tasks. The material properties of the body wall and associated actuators will need to be, in effect, part body and part brain.

Figure 10 contains a photograph of a prototype of such a system. The main body of the robot is cast from a soft silocone elastomer and actuated using SMA wires. Our aim in this case is to attempt

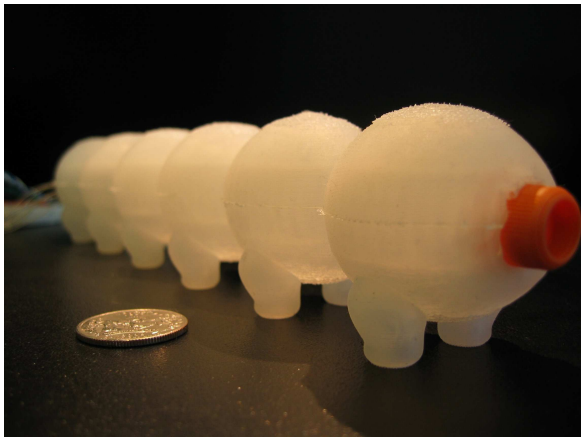


Figure 10: A prototype of the soft robot inspired by *manduca*.

to mimic, albeit at coarser grain, the placement of muscles within *manduca*. Both the elastic properties of the silocone and the tension of the SMA springs can be tightly controlled, which will be vital for exploiting dynamics between them. It is no coincidence that the pseudo-elasticity demonstrated in the *manduca* muscles is very similar to that exhibited in rubber doped with carbon-black particles (Dorfmann et al., 2007).

Our results with tensegrities demonstrate that it is possible to model and evolve dynamically complex systems which are capable of exploiting effects such as mechanical coupling in order to achieve locomotion. These results do not directly translate to a soft robot however: the use of supple, deformable materials with such complex dynamics means that rigid body simulations, such as those provided by the commonly used Open Dynamics Engine (ODE) physics engine are insufficient. Instead, we will use the PhysX engine developed by Ageia Technologies, which is capable of providing realistic simulations of deformable soft bodies such as cloth and rubber. Within this system, we hope to be able to have tightly control over specific material properties at particular points of the body, such as stiffness, elasticity, without needing to resort to full Finite Element Analysis, which might be more accurate, but at the cost of significantly longer evaluation times.

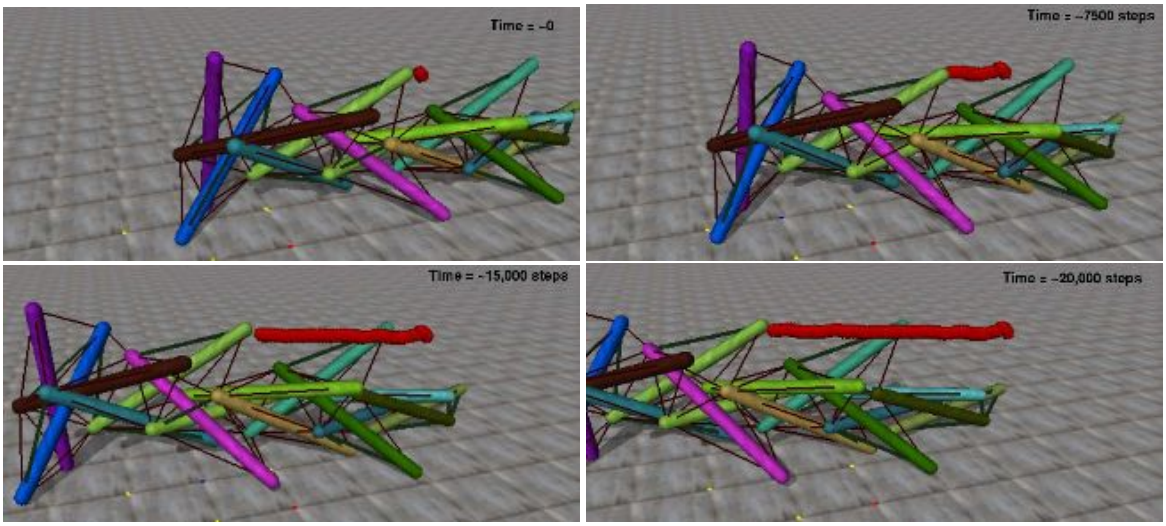


Figure 7: Snapshots of the motion of an evolved gait over 20,000 time steps

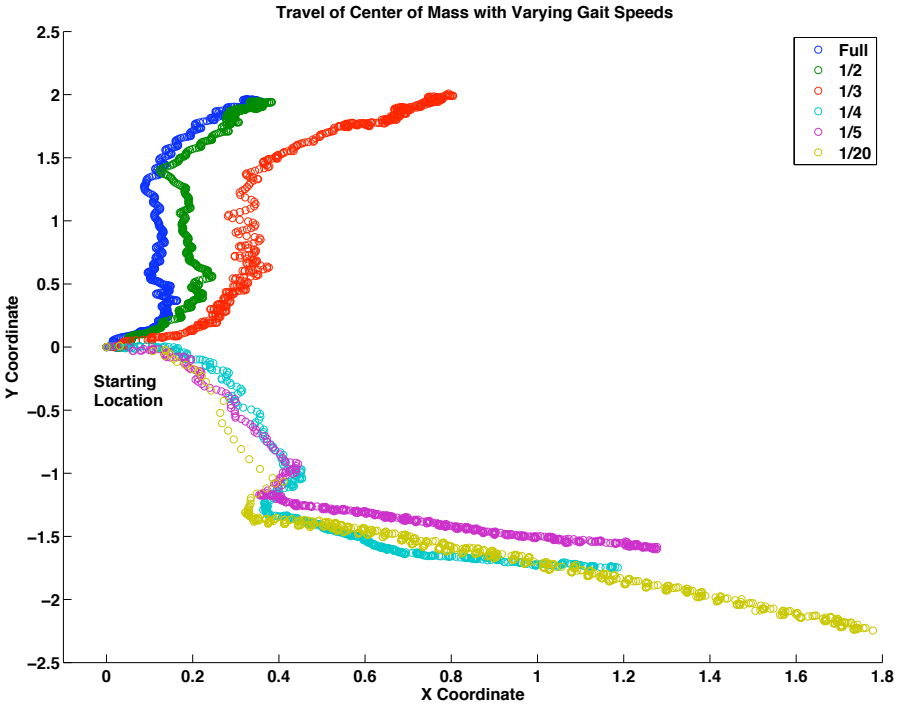


Figure 8: As the speed of the evolved gait is decreased both the distance traveled and the path traversed vary significantly.

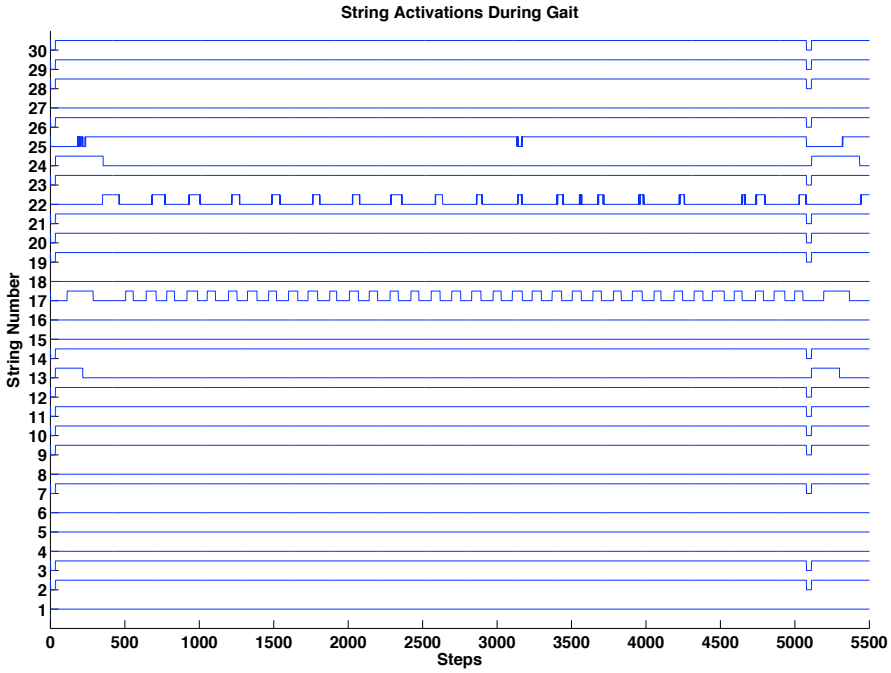


Figure 9: String activations of the evolved gait. Notice how very few of the strings are higly active. This indicates a high degree of efficiency in the gait, as the dynamic coupling between modules discributes the actuation throughout the structure.

Concluding Remarks

Advances in material science are bringing the promise of soft, flexible robots closer to reality. With the benifits of these new abilities and behaviors come new challenges in design and control. How can you actuate, much less control, a floppy amorphous structure that lacks any rigid elements? Fortunately a solution exists in the forms of biological invertebrates such as the octopus and the *manduca sexta* caterpillar. It is becoming clear that much of the ability of these animals lies in the particulars of their morphology – smart structures, in essence, which reduce the amount of neural computation required to perform complex tasks. Since we know it occurs in nature, we hope to reproduce simmilar effects in a soft robot. Here we have shown how one such form of *morphological computation* can arise in a complex mechanical system as well - in our case a large irregular tensegrity structure. With what we know, and what we hope to soon learn about *manduca*, and with the methodologies employed in making our modular tensegrity robots walk, we hope to shed light on how to build smart, resilient, and sophisticated soft robots. Regardless of the final appearance, it is clear that any successful soft robot's body will be at once both mechanism and mind.

References

Ahn, A. and R.J.Full (2002). A motor and a brake: two leg extensor muscles acting at the same joint manage energy differently in a running insect. *Journal of Experimental Biology*, 205.

Biewener, A., McGowan, C., Card, G., and Baudinette, R. (2004). Dynamics of leg muscle function in tammar wallabies (m. eugenii) during level versus incline hopping. *Journal of Experimental Biology*, 207.

Capadona, J., Shanmuganathan, K., Tyler, D. J., and Rowan, S. (2008). Stimuli-responsive polymer nanocomposites inspired by the sea cucumber dermis. *Science*, 319(7).

Chan, W. L., Arbelaez, D., Bossens, F., and Skelton, R. E. (2004). Active vibration control of a three-stage tensegrity structure. In *Proceedings of SPIE 11th Annual International Symposium on Smart Structures and Materials*.

Connelly, R. and Back, A. (1998). Mathematics and tensegrity. *American Scientist*, 86.

Dorfmann, A. L., Jr, W. A. W., and Trimmer, B. (2007). Muscle performance in a soft-bodied terrestrial crawler: constitutive modelling of strain-rate dependency. *Journal of the Royal Society Interface*.

Fuller, R. B. (1975). *Synergetics—Explorations in the Geometry of Thinking*. Macmillan Publishing Co.

Huang, J., Foo, C. W. P., and Kaplan, D. (2007). Biosynthesis and applications of silk-like and collagen-like proteins. *Polymer Reviews*.

Ingber, D. E. (1998). The architecture of life. *Scientific American*.

Kotay, K., Rus, D., Vona, M., and McGray, C. (1998). The self-reconfiguring robotic molecule. In *IEEE International Conference on Robotics and Automation*.

Levine, R. and Truman, J. W. (1985). Dendritic reorganization of abdominal motoneurons during metamorphosis of the moth, manduca sexta. *Journal of Neuroscience*, 5:2424–2431.

Maass, W. and Bishop, C. M. (1999). *Pulsed Neural Networks*. MIT Press.

Mezoff, S., Papastathis, N., Takesian, A., and Trimmer, B. A. (2004). The biomechanical and neural control of hydrostatic limb movements in manduca sexta. *J. Exp. Biol.*, 207(3043-53).

Motro, R. (2003). *Tensegrity: Structural Systems for the Future*. Kogan.

Skelton, R. E., Pinaud, J. P., and Mingori, D. (2001). Dynamics of the shell class of tensegrity structures. *Journal of the Franklin Institute*.

Trimmer, B. (2007). New challengees in biorobotics: incorporating soft tissue into control systems. In *IEEE International Conference on Robotics and Automation*.

Trimmer, B. and Issberner, J. (2007). Kinematics of soft-bodied, legged locomotion in manduca sexta larvae. *The Biological Bulletin*.

Valero-Cuevas, F., Yi, J., Brown, D., McNamara, R., Paul, C., and Lipson, H. (2007). The tendon network of the fingers performs anatomical computation at a macroscopic scale. *IEEE Trans Biomed Eng.*, 54:1161–6.

Yim, M., Duff, D., and Roufas, K. (2000). Polybot: a modular reconfigurable robot. In *IEEE International Conference on Robotics*.

"Embryo": an autonomic co-operative service management framework

Fabrice Saffre[1] and Mark Shackleton[1]

[1] BT Group plc, Pervasive ICT Research Centre, Adastral Park, Martlesham Heath, Ipswich IP5 3RE, United Kingdom
fabrice.saffre@bt.com

Abstract

In this paper, we present "Embryo", a fully decentralized service management framework inspired by morphogenesis and capable of installing components and modifying the topology of a peer-to-peer (P2P) interaction overlay network so as to meet the needs of the majority of all participating peers. Co-operation is an emergent property of the self-organisation process, which is underpinned by purely "selfish" decision-making based on incomplete information gathered through gossiping (local messaging). We provide a detailed description of the local reasoning loop governing the behavior of individual peers, as well as Monte Carlo simulation results that demonstrate the system's ability to converge to a stable state in which most peers have direct access to all the components they require via one of their first neighbors.

Introduction

The last few years have seen a huge number of papers on so-called "complex networks" in both natural (e.g. social, ecological, genetic) and artificial (e.g. power grid, communication) systems [1]. The main reason for the "vitality" of the field is arguably that many scientists have come to realise that the combination of graph theory and complexity science could provide them with powerful tools, provided that the specific problem they are trying to solve is translated into a set of vertices/nodes connected by edges/links. To a large extent, whether the nodes are genes, species or routers and the links chemical reactions, predator-prey interactions or fibre optic cables can be considered as irrelevant when the system is described as a complex network, which makes the associated investigation techniques very widely useful indeed.

It has been argued however that the dynamical properties of complex networks have been somewhat neglected and that maybe too much emphasis was being put on descriptive analysis (e.g. the search for power laws [2]). It is a fact that the mechanisms leading to the emergence of the various network structures have not always been studied in appropriate detail, and that some fundamental aspects of network growth, evolution and decay have yet to be explored. To some extent, the influence of dynamic node properties on network genesis is the focus of this paper.

Understanding the "history" of networked systems (i.e. of how they came to exhibit a particular structure) is of particular significance in the case of overlays, i.e. virtual or logical networks superimposed on (and usually not paralleling) the underlying physical infrastructure. Indeed, being "immaterial" entities, overlays can reorganise themselves very quickly and easily, potentially resulting in macroscopic topological changes occurring over a short period of time. Since in many cases, rewiring in overlays is based exclusively on locally available information and only involves P2P interactions, understanding how a given structure can emerge requires careful examination of the local rules governing decision making and/or information sharing. If the objective is to build an overlay network to serve a particular purpose, then the problem becomes to engineer those rules so as to generate and maintain a desirable configuration.

Some remarkable work has recently been done in this area by Jelasity and Babaoglu [3]. These authors propose a fully decentralised, gossip-based rewiring method (christened "T-Man") to organise a vast set of nodes so that each one of them rapidly "migrates" to the correct place in the whole (initially random) web of relationships. In T-Man, the "correct place" is entirely determined from the start by some static, node-specific information (which can be assimilated to a unique ID in an addressing system). There are many situations however in which the local property that the self-organising process is concerned with will simply not be a constant, due to individual elements changing all or part of their characteristics at the same time that the web of interactions is being reconfigured.

This problem is very similar to that found in morphogenesis (biological development), where individual stem-cells simultaneously differentiate (i.e. specialise) and move in space (equivalent to rewiring in a network) until cells of the right type occupy the right location in the developing organism. Indications are that the type of each cell is not assigned from the onset, but that differentiation is the result of signalling: in effect, neighbours influence each other's "choice" through a dynamic web of positive and negative feedbacks, the structure of which varies due to physical relocation of individual cells [4].

We used this aspect of the developmental process as a source of inspiration because we find it to share many characteristics with co-operative peer-to-peer (P2P) service provision, a type of application that is very different from content-sharing, as it is typically characterised by lower diversity (there are fewer different service components than there are unique files in a typical file-sharing community) and more subtle interaction patterns (peer activity isn't limited to propagating queries and uploading/downloading content). Note that biologically inspired approaches have been applied to content distribution by other authors (see e.g. [5]), but that this problem is outside the scope of our paper.

In P2P service provision, components are distributed across a collection of processing nodes, each of them hosting only a sub-set of all services locally and relying on different nodes to satisfy other needs, via remote access. In terms of the developmental biology metaphor, deciding which service to host is equivalent to differentiation (with the same complex, non-deterministic nature, due to the quality of a choice being primarily a function of other units' decisions), while identifying and selecting suitable providers through rewiring of the overlay network is somewhat similar to physical migration.

Motivation and related work

The service management framework presented in this paper is motivated by the need for autonomic (self-managing) solutions that support future service component-based systems. Two areas where such autonomic solutions are required include Web Services and Pervasive Computing.

Web Services [6, 7] encapsulate both a technology and an industry trend towards distributed, component-based business solutions, where an application is realised by linking many individual services together into complicated workflows and higher-level composite services.

Related to the Web Services domain is the Open Grid Services Architecture (OGSA) architecture that is being developed by the Global Grid Forum (GGF). The aspiration of this work is to support the distributed interaction and interoperability of large numbers of component services in order to meet the needs of users, especially in the eCommerce and eScience domains [8]. Some of the technical challenges arising in realising such solutions are highlighted in Ian Foster's seminal paper "The Physiology of the Grid" [9].

Pervasive Computing is similarly a combination of an overarching vision [9], industry trends and supporting underpinning technologies, and is of sufficient importance to merit its own journal [11]. The vision here is that of huge numbers of communicating devices embedded into the physical fabric of everyday life, all of which need to be marshaled to provide services effectively and efficiently. An interesting example of an adaptive approach to realising a service provision framework in a pervasive computing context is given in [12, 13].

Taken together, these technology trends provide huge opportunities, but they also exacerbate problems which already plague ICT systems, namely the complexity of such systems arising from huge numbers of interacting component parts and the cost of deploying and managing the behaviour of such systems. IBM have probably captured the challenges most clearly and succinctly by launching the Autonomic Computing initiative and associated challenges [14].

Given that future applications and services will almost certainly be realised by coordinating the actions of relatively fine-grained component services, the question arises as to how this might be achieved. Clearly the component services need to be encouraged to cooperate together to provide higher-level services, and given the envisaged scale and complexity, much of this cooperation needs to be inherent, or "engineered into", the underlying system as self-organising principles. This paper presents a solution that is a step along that path.

The need for self-managing approaches for Web Services has been acknowledged by the Open Grid Services Architecture (OGSA) in their stated objectives for "self-management services" and "service level managers" but it is recognised that this work is at an early stage and is currently more aspirational than actual [8]. The approach currently being taken is clearly motivated by that advocated by IBM as the "autonomic management" control loop [15] - in essence this relies on a "generic control loop pattern" comprising monitoring, analysis, projection and action phases [8]. While this pattern will clearly form an important element in many autonomic systems, in this paper we advocate autonomic approaches that are more "inherent" in the design and behaviour of the system as a whole. This has the additional benefit of producing a more lightweight architecture.

In fact our approach is much more closely allied with the techniques and algorithms arising from a rather different community whose research is sometimes labelled as "self*" and often draws upon highly interdisciplinary concepts and models, such as biologically inspired autonomic solutions [16, 17]. Of particular interest to us is the work of Babaoglu who has drawn heavily on biological inspiration to realise a number of highly effective self-organised solutions to P2P data storage and routing, such as T-Man [3]. This and related work provides decentralised solutions and algorithms that are capable of maintaining reliable "overlay networks" over which service can be delivered, even when many component nodes may be individually unreliable [18].

This paper builds on solutions such as those advocated by T-Man, using similar self-organising principles to maintain an overlay network, but also allows individual nodes (service providers/consumers) to dynamically change their "type" in response to perceived demand in a fashion that provides useful autonomic features that support desirable system-level behaviour. While much of the existing research has focussed on data sharing over P2P overlay networks [19, 20], our focus is rather on the provision of a richer set of services via an overlay network that helps coordinate the cooperative behaviour of many individual service components. In this sense it is more closely related to the Chameleon system for self-organised and decentralised P2P web services [21].

Finally, we should stress that ultimate aspirations of both the Self* and Web Service research communities are in fact quite closely aligned. In practice we expect the real benefit will be most effectively realised when solutions such as those presented in this paper begin to be combined with those arising from initiatives such as the Open Grid Services Architecture in a truly interdisciplinary fashion.

Basic Principles

Like T-Man, Embryo uses exclusively gossiping to propagate information throughout the network and individual nodes can choose to swap neighbours based on whatever they learn about their counterparts through this process. Unlike in T-Man, they do not select neighbours based on a static identifier, but by trying to establish a set of symbiotic relationships with partners whose "specialty" complements their own at the time when the link is created. Because individual nodes can subsequently choose to stop hosting a given service and start hosting another (i.e. change type), their

relationships can become unsuitable which will eventually initiate a rewiring process. These dynamics, resulting from multiple nodes changing type and neighbourhood links concurrently, are very analogous to those found in morphogenetic systems.

For simplicity, even though these limitations would probably not apply as strictly in any real co-operative P2P system, Embryo cells are assumed to host only one service at a time (i.e. they cannot simultaneously perform several functions) and to have a fixed maximum degree (i.e. they cannot create additional links, meaning that establishing a new connection requires terminating another one). We also make the implicit assumption that link cost and/or capacity is homogeneous throughout the system, and so that rewiring of the overlay doesn't in any way affect performance (i.e. two nodes hosting the same service are equally capable of providing that service to any other peer)..

The local reasoning loop is similar to the one we used in previous work [22]. Whenever a node requires a service that it is not hosting itself, it first determines if it already knows a provider. If it does, it just sends a request to this particular neighbour. If it doesn't (or if the known provider fails to provide the service due to having changed type since the link was established) the originator of the request tries to identify an alternative provider. Unlike in our previous work however, this is not done by broadcasting a request, but by searching a locally kept stack of "adverts".

Adverts form the basis for the gossiping system of Embryo. Basically, every time that a node fails to identify a suitable provider, it prepares an advert specifying its own identification, which service is needed and which one it can provide in return (i.e. its present type). Adverts are periodically exchanged between neighbours, and propagated for a fixed amount of time (as in many P2P systems, this "time-to-live" mechanism is used to stop traffic from increasing indefinitely, by ensuring that outdated requests are discarded). Before forwarding adverts, every node keeps a local copy, building itself a partial but expanding and regularly updated picture of the offer and demand throughout the system (the adverts stack).

When searching for a provider, a node sequentially examines the adverts (most recently received first), looking for one that contains a requests for its own current type and offers the service that it needs in return. If such an advert is found, the node contacts its poster and a handshake procedure is initiated, which will only succeed if:

- Both nodes still have symmetrical needs (i.e. the poster hasn't changed type or found another provider since the advert was created).
- Both nodes have a spare (i.e. currently unallocated) or useless (i.e. connected to a peer that doesn't provide a necessary service) link.

If it does, the new co-operative link is created.

If a node consistently fails to identify a provider for a particular service, it may choose to take the radical action of discontinuing the service that it is currently hosting and replace it with the one for which it has not been able to find a "collaborator". This obviously creates a crisis for the node and for its neighbours, as it instantly makes all existing symbiotic links obsolete (since the node that has just changed type will no longer be capable of providing the service for which they were established, making the relationship useless for its partners). The underlying assumption is that this crisis can and will be resolved by a cascade of other modifications (of the overlay's topology and/or of individual nodes' specialty), and that the change of type will contribute to increase availability of the incriminated service.

The key difference between Embryo and T-Man is therefore that in the former, the self-organisation process involves both rewiring and modification of local properties. In other words: a node can migrate towards a location in the network where its current type is needed, change type in an attempt to turn itself into a kind of unit more appropriate to its current location, or even combine both procedures.

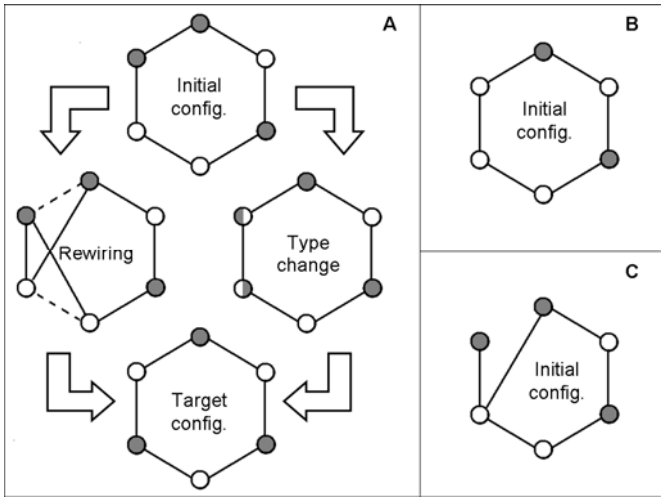


Fig. 1. Illustration of how "rewiring" and "differentiation" processes can lead to the same target configuration (A) and potentially problematic initial conditions (B and C).

Figure 1a illustrates, in a particularly trivial case, how the two processes can produce the same result. If the system was managed exclusively via T-Man "rewiring", it could only follow the left-hand path, whilst if it was relying on differentiation only, as in our own previous work [23, 24], it could only follow the right-hand path. The advantage of being able to combine both methods as in Embryo is made clear by the fact that, had the initial configuration been the one shown in fig. 1b, it would have been impossible to reach the target by rewiring only. Similarly, if it had been 1c, type change alone would have been insufficient.

Only with Embryo is it guaranteed that the exact target configuration can be reached from any of the initial configurations shown on fig. 1, and it is clear that the further the initial conditions are from the desired system state (topologically and/or in terms of node type distribution), the more potentially useful it is to be able to combine rewiring and differentiation.

Detailed algorithm

Our simulated implementation is based on a P2P architecture in which every node is expected to provide and request services to/from its counterparts. As a result, all units are currently governed by the same decision rules, even though it is relatively straightforward to introduce variants in

which parameter values (or the rules themselves) would be specific to an individual and reflect its unique constraints and/or requirements.

Decision making

In the present state of Embryo, every node is assumed to perform only one function at a time, i.e. it cannot simultaneously host/provide more than one set of services (assimilated to belonging to a specific "node type"). As a result, acquiring the ability to perform a new function requires changing type (which implies losing the ability to perform the previous one). However, this simplification was introduced for clarity only and is not a fundamental limitation of the proposed algorithm (qualitatively similar collective decision dynamics could be obtained even if every node was capable of performing several functions, i.e. belonging simultaneously to multiple types).

The functions to which a node needs access are assumed to be completely identified locally (i.e. every participant "knows" its own needs). Whenever it requires a function performed, the node behaves as follows:

- If it already knows (i.e. is connected through the overlay to) a provider, i.e. another node belonging to the right "type", it sends a service request.
- If it knows no provider (or if the provider fails to answer the request for whatever reason), it looks through its locally kept list of adverts to identify a suitable candidate and initiate contact (see "messaging").
- If it doesn't find a compatible advert or if the handshake fails (for whatever reason, see "messaging" for details) it has 2 options:
    - prepare a new "advert" to offer a partnership (see "messaging")
    - turn itself into the requested type and become its own provider.

The last action is the basis of the specialization process. The decision by the node to turn itself into the requested type is probabilistic and is based on its perception of the corresponding function's availability. The probability P of changing type obeys:

$$P = 1-1/(1+(x/x_c)^{\alpha}) \qquad (1)$$

where x is the number of failed requests for that particular service, decremented (incremented) by one at each successful (unsuccessful) attempt, and re-initialised to zero if the node turns itself into the corresponding type, and xc and α are parameters. The choice of function is arbitrary and another could have been used instead (preliminary results only suggest that it should be a sigmoid, which confirms intuition for those familiar with similarly self-organising systems found in nature). For the purpose of the proof-of-concept simulations, we used $\alpha = 2$ and $x_c = 4N$ (where N is the number of types).

We have also experimented with variants in which the probability that a node changes type also decreases with the number of such "metamorphoses" that it has undergone already. For that purpose, we used the following transformation from P into P*:

$$P^* = P\, e^{-\beta y N} \qquad (2)$$

where y is the number of previous type changes, N is the total number of types and β is a (positive) parameter. However, to facilitate interpretation of the results, this modification was de-activated in the version used for the simulations.

Messaging

Embryo relies on gossiping along co-operative links in the overlay network to propagate information about system state. Every node keeps a local list of the so called "adverts" that have reached it, indexed first by the function that they offer, second by their "age" (newest on top). An advert contains 4 distinct pieces of information:

- The type of the sender at the time when the advert was generated (i.e. the service/function on offer)
- The type/function requested in exchange (i.e. the service needed by the sender at the time when the advert was generated)
- The unique ID of the sender (e.g. an IP address or computer name)
- A timestamp

At irregular intervals (probabilistic decision), a node opens its "inbox", where new incoming messages are stored between inspections. It is a rule that, when opening an advert message, the recipient immediately checks whether the local list already contains an entry from the same provenance. If it does and the new advert's timestamp designates it as more recent (which isn't necessarily the case, as a newer advert could have arrived first if following a different gossiping route), it replaces the older one. As a result, there can never be more than one entry per node (sender) in the local list of adverts.

Whenever a node receives an advert that modifies its own local list (i.e. new provenance or new offer/request from an already identified source), it also creates a copy in its "outbox". The content of the outbox is forwarded to the node's neighbours (in the co-operative overlay network), also at irregular intervals (probabilistic decision). Constraints can be imposed on the number of messages that can be sent to every neighbour in order to accommodate link capacity. Also, a time limit can be added so that possibly "outdated" adverts do not unnecessarily clog the network.

When in need of a service for which it either knows no provider or has failed to contact it, a node will look through its locally maintained list of adverts. If it finds one with the right characteristics, i.e. if:

- The service on offer is the one that has just been identified as being currently unavailable (i.e. the one that triggered consultation of the adverts list)
- The service requested in exchange matches its own type (i.e. it can honour its part of the deal)

then the node makes an attempt at contacting the sender of that particular advert.

Assuming that this attempt is successful, a handshake procedure follows whereby both nodes examine the opportunity of forging a new cooperative relationship. This will succeed only if:

- The sender of the advert hasn't changed type since it was sent (and so is still capable of providing the service advertised).
- It still hasn't identified another provider for the service hosted by the initiator of the handshake.

Other factors would probably need to be taken into consideration in a real implementation (e.g. QoS level, service charge…) but these are not modelled in the proof-of-concept simulation. Note that, according to this procedure, a node will never maintain more than N-1 links at a time (i.e. the number of services that it cannot provide to itself), but see the discussion about "volunteering" in conclusions and future work.

If the handshake fails (i.e. communication with the sender was successfully established but one or both of the nodes rejected creating a co-operative link), then the adverts is identified as obsolete and cleared from the local list maintained by the initiator of the negotiation. In this case, the requesting node resumes its search through the list of adverts offering the desired service until either it reaches the last advert in the list (newest first) or finds a suitable provider (i.e. handshake succeeds). The failure to establish a workable partnership has two possible causes:

- A service imbalance (i.e. there isn't any node hosting the required service that needs access to the function corresponding to the type of the node trying to initiate a new co-operative link), in which case the best corrective action to take is to change type.
- The local information about availability is inaccurate (out-of-date or somehow corrupted) or incomplete, in which case the best corrective action is to (re-)advertise the desired co-operative relationship, so that potential candidates that are either unknown or wrongly identified as unsuitable (i.e. assumed to host a different service or not to be "interested" in the initiator's type/offer) become aware of the opportunity.

The decision to follow either course of action is made based on exp. (1), whereby the probability of choosing the "metamorphosis" option increases with the number of failed attempts (which can be interpreted as an indication that service imbalance, not lack of information, is indeed the cause of the repeated inability to forge a partnership).

Results

We used Monte Carlo simulation to gather evidence about the overall efficiency (speed, scalability, robustness…) of Embryo. We do not have space here to present all of our data, so we will focus on showing how system behaviour is affected by the value of the two main parameters, i.e. population size and number of services (N).

All results are for 100 independent realisations. The simulation stops when all peers are either "fully satisfied" (i.e. they have one first neighbour of every type different from their own) or belong to a disconnected sub-graph that cannot reach steady state (size <N). Cases in which this condition was not met before reaching an arbitrary time limit were only encountered in networks of less than 64 peers in the N = 17 scenario and were discarded.

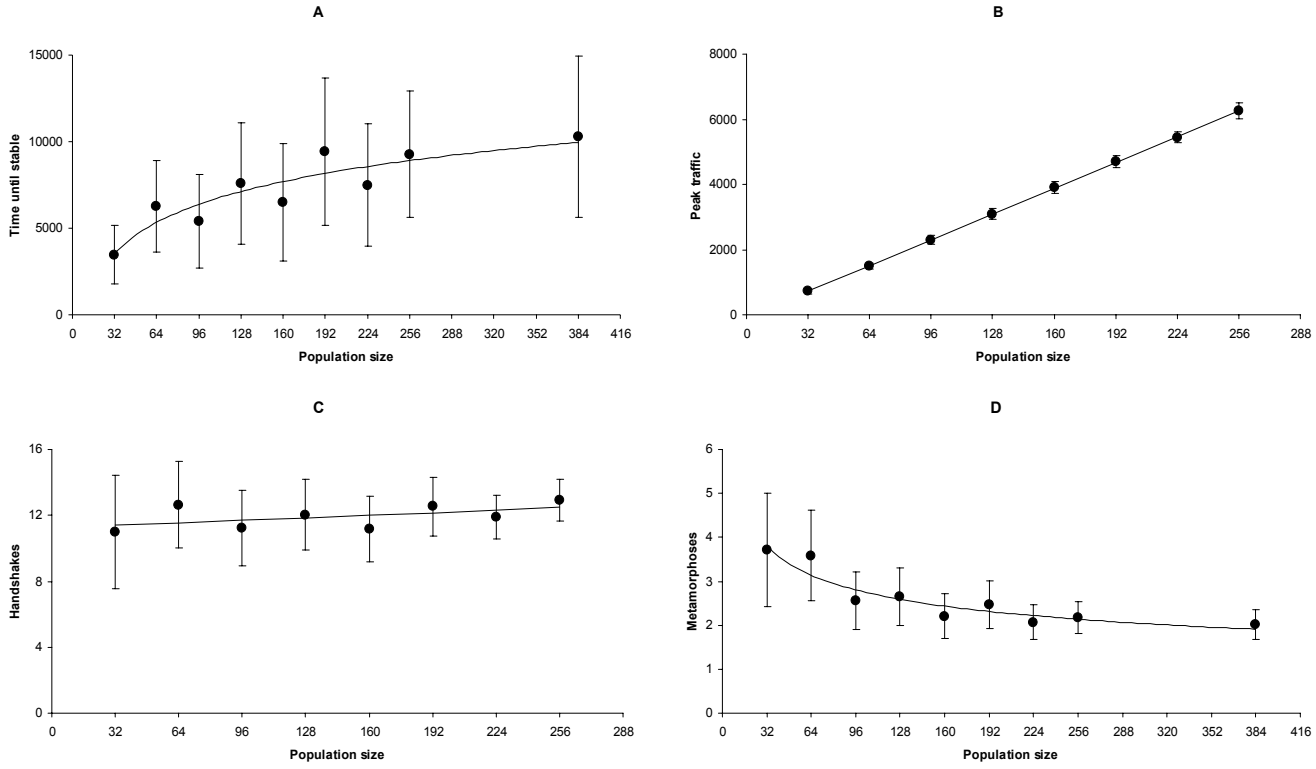


Fig. 2. Evolution of four key variables as a function of population size for N = 9 types or services. (A) Time to steady state. (B) Peak traffic. (C) Number of successful handshakes per node. (D) Number of metamorphoses per node. Error bars indicate standard deviation

Note that since our objective is to demonstrate the fundamental properties of the algorithm and present generic trends revealed by the simulation, units are arbitrary (e.g. "time" is simply the number of simulation steps, "traffic" is the number of adverts being propagated per time-step). For the same reason, we focus on average values, though we also provide the standard deviation as an indication of variability. A more detailed statistical analysis would only be justified if the data consisted in experimental measurements, or if we had simulated system operation at a considerably lower level (e.g. by emulating realistic communication protocols featuring the equivalent of latency, packet loss etc.).

Figure 2 shows the evolution of some key variables, namely the time to reach steady state (A), the peak traffic (B), the number of successful handshakes (C) and the number of metamorphoses (D) for N = 9 services and a variable number of peers. Clearly, the time to reach steady state is a logarithmic function of the population size, while the peak traffic (maximum number of adverts being propagated per time unit) grows linearly with the value of that same parameter. The number of successful handshakes per peer (corresponding to rewiring of the overlay) only increases very slowly with population size, while the number of metamorphoses (corresponding to type change) appears to obey an inverse power law.

These trends all emphasize Embryo's scalability with respect to population size. In particular, the drop in the average number of metamorphoses per peer (likely to be the most "costly" operation) seems to indicate that our algorithm would actually perform better in large systems than in small ones.

As for the influence of the number of services, we ran a number of tests with N = 17. The results are shown in figure 3. We discarded the results for 32 peers as this proved to be a pathological case, with less than 50% of simulation runs converging before reaching the time limit. We attribute this fact to the comparatively low population size / number of services ratio (which implies that the only steady state involves one fully connected graph of 17 peers). Otherwise, the information shown on fig. 3 is identical to the one shown on fig. 2.

Overall, scalability with respect to the number of services is confirmed, though the noise level is comparatively high for the time to reach stable state (fig. 3a). For intermediate to large population sizes, the time to converge doesn't appear to increase much compared with the N = 9 scenario, which is in accordance with the logarithmic trend. As for peak traffic it appears almost unaffected by the number of services. Interestingly, though higher in absolute terms, the number of successful handshakes per peer (fig. 3c) now decreases with population size, which again tends to indicate that performance actually increases as the system becomes larger.

.

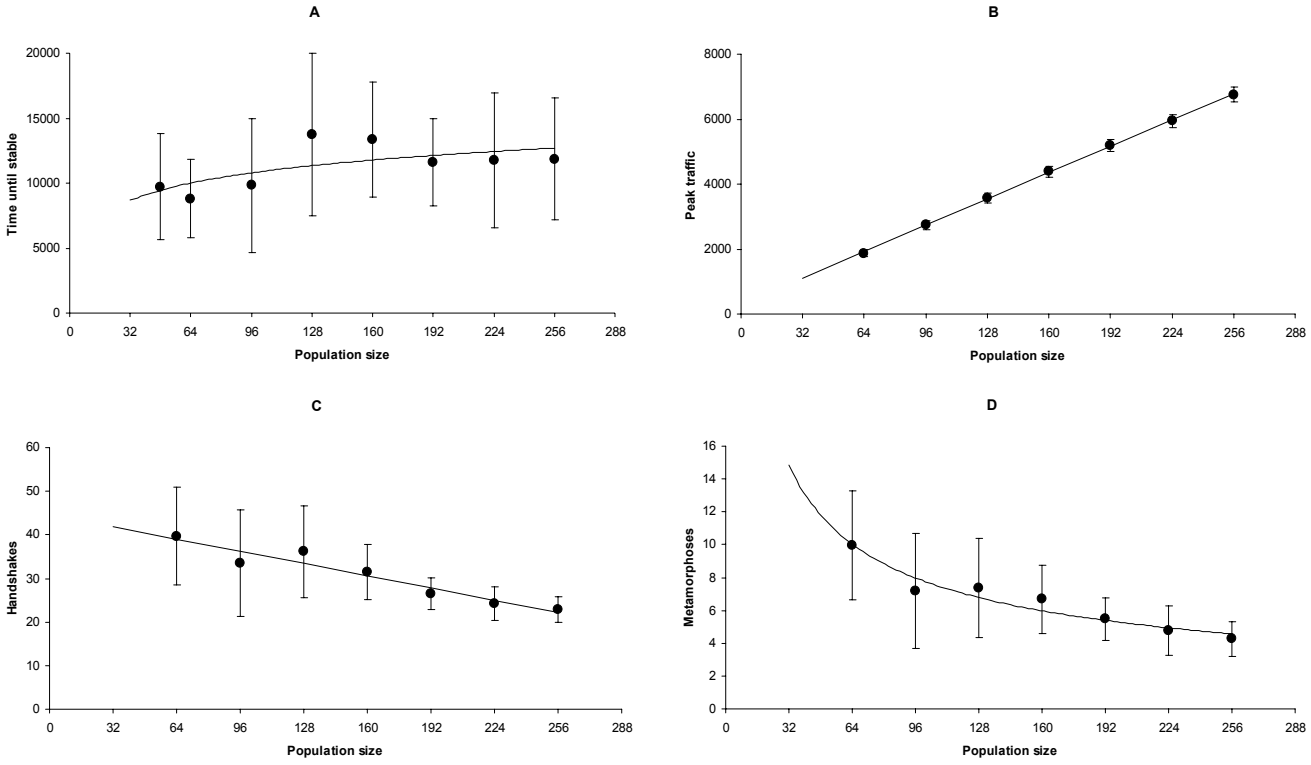


Fig. 3. Evolution of four key variables as a function of population size for N = 17 types or services. (A) Time to steady state. (B) Peak traffic. (C) Number of successful handshakes per node. (D) Number of metamorphoses per node. Error bars indicate standard deviation.

Conclusions and future work

All our results seem to designate Embryo as a suitable design philosophy to build an efficient and reliable P2P service provision infrastructure, especially in large and unpredictable resource-sharing communities. We argue that this makes our algorithm a strong candidate for autonomic deployment and maintenance of ICT systems at the interface between Service-Oriented Architecture (SOA) and Grid computing. This however would require taking into account some additional characteristics that, for the sake of clarity and completeness, were not included in the present study.

For example, the procedure for establishing a relationship described in this paper implies reciprocity/symmetry, in the sense that a co-operative link is only established if each partner decides that it is in its own best interest to choose the other as a provider for a required service. On the contrary, a node can change type (which renders all of its co-operative relationships immediately obsolete) or terminate a link without consulting (or even notifying) its counterpart(s). This simultaneously offers a guarantee against cheating (since any node can unilaterally decide to withdraw from a relationship that it judges unsatisfactory) and makes it obvious that "selfishness" is not an obstacle to the self-organisation process.

We have however experimented with a modified version of the decision and messaging infrastructure in which the initiator of the handshake procedure may accept to perform a function for another node even if it has no personal interest in doing so (we call this procedure "volunteering"). In this version, we assume that a node is able to provide a service to more peers than it has needs (i.e. it has more links than the minimum necessary to meet all of its own requirements). Preliminary findings suggest that such "volunteering" can be highly beneficial to the community as it appears to speed up the self-organisation process and leave fewer or no nodes "excluded" from the final (stable) overlay.

Finally, in biological development, the "preferred neighbourhood" varies from one cell type to the other, leading to the formation of functional organ and tissues, which are basically specialised structures made of a an aggregation of cells belonging to a small sub-set of all possible types. Interestingly, this is also the case in P2P service provision, as not all peers need access to all services, and so the "homogeneous full coverage" scenario described in this paper is obviously a simplification. The more complex collective dynamics likely to emerge in an extended version of Embryo taking into account these various other aspects will be the subject of future work.

References

[1] S. H. Strogatz, "Exploring complex networks" Nature 410, 268-276 (2001).

[2] A.-L. Barabasi, R. Albert. "Emergence of scaling in random networks" Science 286, 509-512 (1999).

[3] M. Jelasity, O. Babaoglu "T-Man: Gossip-based overlay topology management" In: Proceedings of the 3rd International Workshop on Engineering Self-Organising Applications, Utrecht (2005).

[4] S. Kumar, P.J. Bentley. (eds.) "On Growth, Form and Computers" Elsevier Academic Press, London (2003).

[5] J.-J. Suh, S. G. Quan, S.-H. Park, Y. Y. Kim, "Adaptive File Distribution in P2P Network Using Ant Colony Optimization for Smart Home Environment" International Conference on Hybrid Information Systems, Nov. 2006.

[6] Michael Stal, "Web services: beyond component-based computing" Communications of the ACM, vol. 45, no. 10, Oct. 2002.

[7] L. F. Cabrera, C. Kurt, D. Box. "An Introduction to the Web Services Architecture and Its Specifications" Version 2.0, Oct. 2004. http://msdn.microsoft.com/webservices/webservices/understanding/advancedwebservices/default.aspx?pull=/library/en-us/dnwebsrv/html/introwsa.asp

[8] I. Foster, H. Kishimoto, A. Savva, D. Berry, A. Djaoui, A. Grimshaw, B. Horn, F. Maciel, F. Siebenlist, R. Subramaniam, J. Treadwell, J. Von Reich. "The Open Grid Services Architecture, Version 1.0" Informational Document, Global Grid Forum (GGF), January 29, 2005. http://www.gridforum.org/documents/GWD-I-E/GFD-I.030.pdf

[9] I. Foster, C. Kesselman, J. Nick, S. Tuecke "The Physiology of the Grid: An Open Grid Services Architecture for Distributed Systems Integration" Open Grid Service Infrastructure WG, Global Grid Forum, June 22, 2002. http://www.globus.org/alliance/publications/papers/ogsa.pdf

[10] M. Weisner (1991) "The computer for the 21st century" Scientific American, 265(3):94—104 (1991).

[11] IEEE Pervasive Computing, IEEE Computer Society, ISSN: 1536-1268.

[12] L. McNamara, C. Mascolo, L. Capra "Trust and Mobility Aware Service Provision for Pervasive Computing" Workshop on Requirements and Solutions for Pervasive Software Infrastructures (RSPSI) held at 4th International Conference on Pervasive Computing, May 2006.

[13] L. Capra, S. Zachariadis, C. Mascolo "Q-CAD: QoS and Context Aware Discovery Framework for Mobile Systems" ICPS, July 2005.

[14] J. O. Kephart, D. M. Chess "The Vision of Autonomic Computing" IEEE Computer, Jan. 2003.

[15] N. Chase "An autonomic computing roadmap" Oct 2004 http://www-128.ibm.com/developerworks/library/ac-roadmap/

[16] A. Keller, J. P. Martin-Flatin (eds). Second IEEE International Workshop on Self-Managed Networks, Systems & Services (SelfMan 2006 Proceedings), June 2006, Dublin, Ireland; published by Springer Verlag as Volume 3996 of the Lecture Notes in Computer Science (LNCS) Series.

[17] O. Babaoglu, M. Jelasity, A.Montresor, C. Fetzer, S.Leonardi, A. van Moorsel, M van Steen “Self-Star Properties in Complex Information Systems", Lecture Notes in Computer Science, Hot Topics, vol. 3460, Springer-Verlag, 2005.

[18] G. P. Jesi, A. Montresor, O. Babaoglu “Proximity-aware Superpeer Overlay Topologies” Second IEEE International Workshop on Self-Managed Networks, Systems & Services (SelfMan 2006), June 2006, Dublin, Ireland.

[19] K. Aberer, A. Datta, M. Hauswirth, "P-Grid: Dynamics of Self-organizing Processes in Structured P2P systems", In: Peer-to-Peer Systems and Applications, LNCS 3485, Springer, Aug 2005.

[20] K. Aberer, L. Onana Alima, A. Ghodsi, S. Girdzijauskas, M. Hauswirth, S. Haridi "The essence of P2P: A reference architecture for overlay networks", The Fifth IEEE International Conference on Peer-to-Peer Computing, Konstanz, 31 Aug - 2 Sep 2005.

[21] C. Adam, R. Stadler “Implementation and Evaluation of a Middleware for Self-Organizing Decentralized Web Services” Second IEEE International Workshop on Self-Managed Networks, Systems & Services (SelfMan 2006), June 2006, Dublin, Ireland

[22] F. Saffre, H. R. Blok, “SelfService: a Theoretical Protocol for Autonomic Distribution of Services in P2P Communities” In: Proceedings of the 12th IEEE International Conference and Workshops on Engineering of Computer-Based Systems, Greenbelt (2005).

[23] F. Saffre, J. Halloy, J. L. Deneubourg. “The Ecology of the Grid” In: Proceedings of the 2nd IEEE International Conference on Autonomic Computing, Seattle (2005).

[24] F. Saffre, J. Halloy, M. Shackleton, J. L. Deneubourg. “Self-Organised Service Orchestration through Collective Differentiation” IEEE Trans. on Systems Man and Cybernetics, 36(6), 1237-1246. (2007)

The Limited Utility of Communication in Simple Organisms

Matthias Scheutz and Paul Schermerhorn
Cognitive Science Program and Complex Systems Group
College of Arts and Science and School of Informatics
Indiana University, Bloomington, IN 47406, USA
{mscheutz,pscherme}@indiana.edu

Abstract

Many forms of communication have evolved in the animal kingdom for different purposes. In this paper we investigate the limits of communication for simple reactive organisms and show that communication has only limited benefits in biologically inspired foraging tasks and can even have detrimental effects in certain environments. Based on these results, we argue that simple agents with simple architectures need very special environmental conditions for communication to benefit them and thus to evolve.

Introduction

Various forms of communication have evolved in the animal kingdom, ranging from broadcasting simple signals, to the complex linguistic exchanges of humans. Much work in ALife has attempted to demonstrate *that*, *when* and *how* communication can evolve, but paid little attention to cases where communication *did not* or *will not* evolve. Yet, we believe that a full appreciation of the utility of communication for natural and artificial agents is not possible without understanding both its potential *and* limitations.

In this paper we attempt to delineate the kinds of circumstances that would limit the evolution of communication for biologically inspired tasks. We start with a few methodological points about the notion of "communication" and lay out the argument structure we will use here for investigating the limitations of communication. Then we define a biologically inspired task called t-MATES (for "timed Multi-Agent Territory Exploration Task with Satiation") and introduce various agent models for that task. Simulation results will paint a surprising picture, showing that communication is of very little utility for t-MATES tasks. We discuss implications of the results for the evolution of communication and relate them to previous findings in the literature, concluding with a brief summary and suggestions for future work.

Communication and Mechanism

Biological agents have evolved different forms of communication for different purposes, ranging from signaling danger (e.g., danger caws of lookout crows), to indicating readiness for mating (e.g., mating calls of frogs), to reporting locations of food (e.g., food dances of honey bees), to initiating joint action (e.g., dogs' bows to initiate social play), and to sharing mental states (e.g., human reports of their beliefs).

These different forms of communication require different functional capabilities of the agents' architectures. Processing simple signals emitted from another agent that only indicate the agent's presence in some location does not require much more than a perceptual system that can pick up those signals as such and determine the direction from which they originated (e.g., female frogs can determine the signal strength and direction of male callers in a swamp; similarly, ants can sense the gradient of pheromones left in the environment by other ants). In fact, simple signals indicating a particular state of affairs as perceived by an agent can be construed as *indexicals* (in the Peircian sense), i.e., a food call effectively communicates indexical information of the form "I see food here now", see also Perconti (2002). Note that this message containing three indexicals is different from the message "Agent A sees food at location X at time t" even if the content of the message is the same (i.e., the variables A, X, and t are replaced by the respective names so that agent, location and time agree with the utterance of the indexical message). For messages of the latter sort can realistically not be encoded in simple indexical signals (unless one has a large number of distinct signals for all occasions of interest at hand, which is practically almost never feasible). Hence, representational devices are needed to represent agents, places, and times in the second case. Those representational devices, in turn, require a systematic encoding (i.e., representations with formal rules defining well-formed expressions) and mechanisms that can encode and decode information (i.e., parsers). Moreover, to determine times, locations, and agents (as in the above case), agreed-upon scales (e.g., clocks, maps, and naming conventions) are required together with "measuring devices" (i.e., algorithms and possibly tools) to determine that determine where instances fall on the scale (i.e., what time it is, where in the map an item is located, and who the speaker is). All of this, in turn, requires much more sophisticated functional capa-

bilities in agent architectures that allow agents to determine what to communicate and how to use the communicated information. Processing expressions that can encode and thus communicate mental states like beliefs, for example, might require representational capabilities such as those used in modal or (fragments of) first order logic (e.g., to represent the belief that at least one member of a group has already had dinner).

More complex architectures that can handle the added sophistication of more complex messages (such as their syntax and semantics) comes at a price, however: the cost of building/growing and maintaining it.[1] In contrast, simple signals like "I see food here now" might not require much additional processing at all: the receiving agent could just move towards the perceived signal if it needs food, or ignore it otherwise.

Aside from the computational/architectural costs, the costs of communicating can also be substantive. An agent that continuously sends broadcast signals (like alarm, food, or mate calls) might use up a significant portion of its energy, possibly without any benefit if no other agent can hear the signal (the calls of male treefrogs, for example, are much more expensive than navigation, limiting them to participate in the calling chorus during mating season for only a few days out of several weeks, e.g., see Fellers (1979)).

From all of the above it is clear then that claims about the evolvability of communication or about the likelihood of communication evolving need to be very specific with respect to the form of communication they target, as all the above differences (with respect to communication schemes, functional, representational and computational capabilities of agent architectures, and the various costs) are typically subsumed under the general term "communication".

For example, only social insects seem to have evolved different ways to communicate information about food sources in partly non-indexical ways among their respective groups (from annotating the environment by leaving marks like ants do with pheromone trails, to using intricate dances like those of honey bees that encode direction and distance to food sources as well as food quality), despite the large number of different species of insects. For most other forms of insects, indexical mate signaling is the most that has been described (but see Cocroft (2005) for an example of food signaling in treehoppers).

We will in the following investigate two forms of communication about food: the simplest form of indexicals signaling "I see food here now", and a more complex form of "I see food in location *X* now", which removes one of the three indexicals and replaces it with an explicit value (namely the location of the food item). Since we are interested in determining the limits of communication, we will have to follow a different strategy from most work on the evolution of communication, which we will describe next.

[1] In humans, for example, the brain consumes up to 25% of the body's energy, in infants even up to 75%, e.g., see Cunnane (2006).

Method

Most research on the evolution of communication (see the Related Work section below) attempts to demonstrate that communication is *beneficial* and thus *can* evolve in some agent in some given task. The logical form of these "evolvability claims" is typically an existence claim: for a given agent type and task there exists an initial distribution and an evolutionary trajectory from this distribution that leads to communication in those agents. The existential quantifiers here are often the result of a common strategy to establish claims about the evolvability of communication by examining the outcomes of runs of genetic algorithms or similar evolutionary computational tools. It is, however, important to note that the existential quantifiers critically limit the scope of the claim: it only says that for some initial conditions there are trajectories that lead to communication being beneficial. This leaves open whether communication could have or would have evolved for all trajectories, or the vast majority of trajectories, and thus whether it was *likely* for communication to evolve. For the likelihood of a property evolving in a task, we need to determine the conditional probability of communication evolving *given* as set of initial conditions. The conditional probability formulation can then be used both to confirm and disconfirm that a property $\mathcal{P}$ such as communication is likely to evolve in a set of agents by comparing the performance of agents with $\mathcal{P}$ and without $\mathcal{P}$ for each initial condition. If there is no *absolute performance* difference, then there will likely be no evolutionary trajectory resulting in agents with $\mathcal{P}$, for having and using $\mathcal{P}$ would at best incur an additional cost without yielding any gain in task performance. If there is no *relative performance* difference between agents with and without $\mathcal{P}$ in the given task (i.e., when the cost of having and using $\mathcal{P}$ is taken into account in the performance measure), then the answer to the question whether there is an evolutionary trajectory leading to $\mathcal{P}$ will depend on additional information about intermediary stages of the trajectories, e.g., what evolutionary operations are used and how frequently they are employed, whether these operations can produce viable architectures at any point along the trajectory, etc. Typically, it is difficult (if not infeasible) to obtain this kind of information.

Hence, we will aim at establishing that there is no *absolute performance* difference between agents with and without communication. While it is impossible to do this exhaustively for the sheer size of the set of initial conditions (even in our limited experimental setup), it is possible for a small, but representative subset of initial conditions randomly drawn from the set of all initial conditions. Statistical significance tests can then be used to reject the null hypothesis that there is an absolute performance difference between communicating and non-communicating agents. And the *p*-

value of the significance test can be taken as an upper bound on the conditional probability that communication evolves in environments of the given type (a more detailed exposition of the employed experimental methodology can be found in Scheutz and Schermerhorn (2005)).

Task and Agent Models

To be true to the question about when and why communication evolved in nature, we define a generic biologically plausible territory exploration task that is intended to measure the efficiency with which agents can negotiate their environment (e.g., how they determine where to go in their environment based on their survival goals such as finding food).

Definition *t*-MATES: A *timed multi-agent territory exploration task with satiation* (*t*-MATES) $T(t,C,A,R,D,S)$ requires a group of identical agents A each with sensory range R to visit as many checkpoints in C as possible in a 2D environment within the allotted time t, where agents and checkpoints are placed according to a probability distribution D and each agent can visit up to S checkpoints (the "satiation level").

D is typically unknown to the agents, hence it cannot be *a priori* exploited by them. Agents neither know their own locations in the environment nor those of the checkpoints. Rather they can only detect relative locations of checkpoints based on their perceptions (e.g., the location of a checkpoint relative to the agent's heading). All checkpoints are marked so agents can perceive them when they are within sensory range. Whenever a checkpoint is visited by an agent, the agent removes the mark, thus effectively removing the checkpoint from the environment.

One way to conceptualize *t*-MATES tasks is to think of them as "foraging episodes" (of duration t) taken from an ongoing evolution of populations of biological agents: checkpoints can be viewed as food sources, and visiting can be taken to be consuming them, with the satiation level determining the maximum food intake an agent can have within the foraging period t. Performance of different agent types during t reflects the agent types' *foraging efficiency* (i.e., the efficiency with which agents can find food), which in turn is a fitness measure of their performance in the larger evolutionary context of survival and procreation. That is, if an agent kind K_1 has a higher foraging efficiency than another agent kind K_2 as measured in the *t*-MATES task, where foraging efficiency is given in terms of "average number of visited items per time unit", then one would expect K_1, on average, to perform better than K_2 in *t*-MATES tasks in an evolutionary setting.[2]

[2]The qualifier "on average" is critical here as there can always be special circumstances that punish normally fitter agents and can even lead to their extinction.

Next, we define a simple reactive, yet biologically plausible non-communicating base agent model (e.g., at the level of insect behavior) that meets the minimum requirements for the *t*-MATES task of being able to move about the environment, detect a checkpoint within the given sensory range R and move towards it. For simplicity's sake, we do not employ a particular sensory model (e.g., sonar or visual sensors), which would introduce complicating perceptual effects such as interference or visual occlusions, but rather assume that agents can detect any number of checkpoints within the circular *Area* of radius R around them. Given that checkpoints have an extension in space (1 square unit), the maximum number of detectable non-overlapping checkpoints is limited by $Area = R^2 \cdot \pi$.

The behavior of a *non-communicating agent* is then determined solely by its sensory information (which is limited to checkpoints, other agents are not perceived) based on the following three rules:

Rule 1: if no checkpoint is sensed, perform a random walk $RW(rwd,\beta)$ (i.e., move in the direction of the current heading θ for *rwd* cycles, then change heading randomly to some value in $[\theta-\beta,\theta+\beta]$)

Rule 2: if some checkpoints are sensed and are not within visiting range (i.e., they are not within the extension of the agent's body of 8 units), go directly towards the closest checkpoint (the direction is given by α such that $\min_d\{\langle d,\alpha\rangle | \langle d,\alpha\rangle$ is within sensory range)

Rule 3: if some checkpoints are sensed, at least one checkpoint C is within visiting range, and the agent's count of checkpoints visited c is less than its satiation level S, remove the mark(s) of up to $S-c$ of the checkpoints (if it is/they are still there)

When an agent achieves satiation, it continues to execute the rules above (i.e., it will search for another checkpoint and move to it, but upon arrival will simply remain there until the checkpoint is removed by another agent).

Note that the basic model is parameterized by $RW(rwd,\beta)$ and R, hence its performance will critically depend on those parameters. In a sense, $RW(rwd,\beta)$ is an agent-internal parameter that should be chosen so as to maximize an agent's performance with a given sensory range in a given environment if we want to investigate the utility of communication. However, to be able to choose the best values for $RW(rwd,\beta)$, we need to understand how the random walk interacts with other parameters such as the agent's sensory range, the number of participating agents in the task, and the structure of the checkpoints in the environment (e.g., a random distribution). Hence, we conducted a large set of *calibration experiments* to determine the best random walk distance (*rwd*) for base agents for each *sensory range* $R \in \{25\cdot n | 1 \le n < 10\} \cup \{300+50\cdot n | 0 \le n \le 6\}$ and *group*

size $|A| \in \{2,3,4,5\}$, in both random and clustered environments.[3]

We extend the base agent model in two different ways to allow for two kinds of communication: purely indexical communication – call it *signaling agent* – and partly "representational" communication – call it *messaging agent*. We also allow for two reactions to signals: a typical *approach* behavior (e.g., like those exhibited by Toque Macaques when they hear a food signal Dittus (1984)) and, for contrast, an *avoidance behavior* that will cause agents to walk away from the direction of the food signal. The effect of the avoid behavior should contrast to potential clustering effects engendered by the approach behavior, potentially leading to better agent distribution, particularly in random environments. Thus, we will define four different types of communicating agents.

For the signaling agent we add the following two rules:

Sending: whenever a checkpoint is sensed, the agent turns on its "checkpoint" signal

Receiving: whenever no checkpoint is sensed but one or more checkpoint signals are sensed, the agent either approaches or moves away from the direction of the closest signal.

Similarly, we add two rules for the messaging agents:

Sending: whenever a checkpoint is sensed, the location of the checkpoint is communicated as the $\langle d, \alpha \rangle$ of distances $d \in [0,R]$ and angles $\alpha \in [0,2\pi]$ relative to the sending agent's position

Receiving: whenever no checkpoint is sensed but one or more checkpoint messages are received, the agent either approaches or moves away from the closest checkpoint[4]

Note that messaging agents will at any given time know the locations of all checkpoints that are perceived by any agent within communication range, while signaling agents will only know the locations of checkpoints they themselves perceive, even though they will know where other agents are that perceive checkpoints. Satiated agents will continue to send and receive according to the communication rules for their agent type.

Because we are interested in determining the limitations of communication, we will consider two different communication ranges: an (unrealistic) unlimited communication range (as a control condition) and a biologically plausible limited communication range that is the same as the agent's sensory range.

We thus arrive at nine different agents, which we will label from 0 to 8 for ease of presentation. Agents of type 0 do not use communication, while odd-numbered agents use messaging and even-numbered agents use signaling. Agents types 1 through 4 use *unlimited* communication, while agents types 5 through 8 use *limited* communication. Finally, agents types 1, 2, 5, and 6 use *approach* behavior, while agents types 3, 4, 7, and 8 use *avoidance* behavior.

Experiments and Results

All simulation experiments were conducted in the artificial life simulator SWAGES, which is a distributed agent-based artificial life simulation environment that consists of the parallelizable SIMWORLD simulator and an experiment gridserver used to schedule experiments on heterogeneous clusters of computers, automatically parallelize and distribute simulations over multiple hosts, collect statistics, and perform preliminary data analysis (Scheutz and Schermerhorn, 2006; Scheutz et al., 2006).

One simulation experiment consists of 100 experimental runs, each using different randomly generated initial conditions from a given distribution D (of initial conditions) in a continuous 2D world, which is limited to an 800 by 800 square region (in comparison, each agent occupies a circular region of diameter 8).[5] Two different distributions of checkpoints are used: *random* and *cluster*. In the random distribution, checkpoints are placed at random locations within the whole environment, while in the cluster distribution all checkpoints are placed according to a Gaussian distribution (with a radius of 150 units and a standard deviation of 75 units) centered in one 200 by 200 quadrant (with all checkpoints contained within the quadrant). We consider two different numbers of checkpoints, $|C| = 10$ and $|C| = 40$, and four group sizes of agents, $|A| = 2$ to $|A| = 5$ to investigate the possible effects of food density and group size on the utility of communication. Moreover, we fix the agents' satiation thresholds at $S = 10$, but vary their sensory ranges from 25 to 600. The same set of 100 initial conditions of checkpoint and agent placements is used for all variations of group size and sensory/communication range for a given number of checkpoints and checkpoint distribution to allow for a

[3] For space reasons we omit a detailed description of the results.

[4] The details of exactly how agents extract the exact location of a checkpoint relative to an agent's own heading from another agent's message are not straightforward; they usually involve additional communicated parameters such as heading of agents relative to each other or relative to a fixed coordinate system (e.g., as measured by a compass, etc.). Here we simply assume that the agent can compute the angle and distance to the communicated checkpoints based on where the message came from, and that they somehow have access to the source location. For it will turn out that messaging agents do in general not have better absolute performance than signaling agents, hence the details of the control architecture and the buried complexities and costs do not have to be considered explicitly (as would typically be the case for conditions where messaging agents performed absolutely better than signaling agents).

[5] Whenever an agent reaches the boundary of the environment, it will "bounce" off (similar to a billiard ball bouncing off the cushion) with some very small random error (this is to make sure that agents will not be able to leave the area within which checkpoints are located).

	$C = 10$		$C = 40$	
	Random	Cluster	Random	Cluster
0	6.93 (2.62)	8.07 (3.13)	20.80 (7.55)	20.69 (10.74)
1	6.65 (2.71)	8.41 (2.99)	20.32 (7.95)	25.22 (10.48)
2	6.62 (2.70)	8.37 (2.99)	20.29 (7.93)	24.89 (10.44)
3	6.60 (2.70)	7.82 (3.13)	19.98 (7.99)	18.57 (10.90)
4	6.60 (2.71)	7.81 (3.20)	19.99 (8.03)	18.58 (10.90)
5	6.90 (2.63)	8.08 (3.14)	20.74 (7.58)	21.11 (10.87)
6	6.89 (2.62)	8.07 (3.13)	20.72 (7.58)	21.08 (10.86)
7	6.95 (2.62)	8.05 (3.13)	20.85 (7.55)	20.36 (10.61)
8	6.95 (2.62)	8.05 (3.13)	20.86 (7.55)	20.36 (10.61)

Table 1: Average performance of all nine agent types across all sensory ranges (from 25 to 600) in both types of environments (random and cluster) with both numbers of checkpoints (10 and 40) for all four group sizes (from 2 to 5).

		10 Checkpoints			
		Random		Cluster	
	Df	F	p	F	p
Type	8	59.00	$< .001$	55.07	$< .001$
Range	16	5167.12	$< .001$	3390.00	$< .001$
Type:Range	128	3.93	$< .001$	3.24	$< .001$
Error	61047				
		40 Checkpoints			
		Random		Cluster	
	Df	F	p	F	p
Type	8	25.17	$< .001$	499.00	$< .001$
Range	16	2610.12	$< .001$	1756.89	$< .001$
Type:Range	128	3.82	$< .001$	15.13	$< .001$
Error	61047				

Table 2: Two-way 9×16 Analysis of variance in performance with the independent variables *agent type* and *sensory range* and the dependent variable *checkpoints visited*.

	10 Checkpoints								
		Unlimited Communication Range				Limited Communication Range			
		Approach		Avoid		Approach		Avoid	
	0	1	2	3	4	5	6	7	8
0		+++	+++	+++	+++				
1	---					---	---	---	---
2	---					---	---	---	---
3	---					---	---	---	---
4	---					---	---	---	---
5		+++	+++	+++	+++				
6		+++	+++	+++	+++				
7		+++	+++	+++	+++				
8		+++	+++	+++	+++				
0		---	---	+++	+++				
1	+++			+++	+++	+++	+++	+++	+++
2	+++			+++	+++	+++	+++	+++	+++
3	---	---				---	---	---	---
4	---	---	---			---	---	---	---
5		---	---	+++	+++				
6		---	---	+++	+++				
7		---	---	+++	+++				
8		---	---	+++	+++				

	40 Checkpoints								
		Unlimited Communication Range				Limited Communication Range			
		Approach		Avoid		Approach		Avoid	
	0	1	2	3	4	5	6	7	8
0		+++	+++	+++	+++				
1	---			+	+	--	--	---	---
2	---					---	---	---	---
3	---	-				---	---	---	---
4	---	-				---	---	---	---
5		++	+++	+++	+++				
6		++	+++	+++	+++				
7		+++	+++	+++	+++				
8		+++	+++	+++	+++				
0		---	---	+++	+++				
1	+++			+++	+++	+++	+++	+++	+++
2	+++			+++	+++	+++	+++	+++	+++
3	---	---	---			---	---	---	---
4	---	---	---			---	---	---	---
5		---	---	+++	+++				
6		---	---	+++	+++				
7		---	---	+++	+++				
8		---	---	+++	+++				

Table 3: Comparison of the nine agent kinds in 10 checkpoint (top two tables) and 40 checkpoint (bottom two tables). Within each checkpoint condition, the upper table is in random and the lower table is in cluster environments. "+" and "-" denote significant performance differences (between mean performance of the agent type in the row minus mean performance of the agent type in the column), where the number of symbols indicates the significance level based on Tukey's honestly significant difference (HSD) multicomparison post-hoc test: one symbol for $p < .05$, two symbols for $p < .01$, and three for $p < .001$.

direct performance comparison among the different agent kinds and parameters. We use *the number of checkpoints visited within t* as performance measure and fix $t = 500$, which turned out to be long enough to highlight foraging differences among agent types.

The overall performance results for the nine agent types in the four environmental conditions averaged over all sensory ranges and group sizes are shown in Table 1, the results of ANOVAs for each environmental condition are shown in Table 2, and the statistically significant performance differences are shown in Table 3. The results in Table 2 show that the differences in average performance between agent types are significant (as is the effect of sensory range on performance, unsurprisingly). The interaction between agent type and sensory range is due to performance differences between types found in medium sensory ranges; when sensory range is very low, agents have a very hard time locating checkpoints about which to communicate, whereas when sensory range is high, shared information is seldom novel.

The results in Table 3 demonstrate that in random environments, regardless of the food density, communication does not pay off, not even in the simplest form of signaling (as demonstrated by the lack of minus symbols in the row with agent type 0). Quite to the contrary, unlimited communication can significantly hurt agent performance (see the plus symbols in the 0 agent row). In cluster environments, there is some benefit to communication: agents with unlimited communication range using approach behavior perform better than non-communicating agents (see the minus symbols in the first two columns of the 0 agent row), but not if they use avoid behavior, as expected (see the plus symbols in the third and fourth columns of the 0 agent row). Note that there was no performance difference between the two forms of communication. Messaging agents with unlimited communication range using approach behavior in ran-

dom high density environments do, however, have a slight advantage over those using avoid behavior (see the single plus symbol in columns 3 and 4 of the 1 agent row). Overall, there is no statistically significant performance difference in any of the four environmental conditions between non-communicating and communicating agents with limited communication range (as evidenced by the lack of any symbols in columns 5 through 8 in the 0 agent row).

Discussion

The above results make a strong case for the limited utility of communication for simple insect-like agents in *t*-MATES tasks, especially since there was no statistically significant performance difference between non-communicating and communicating agents with limited communication range. With unlimited communication range, the question about the utility of communication becomes surprisingly dependent on the type of environment: in random environments, performance actually *decreases* due to agents wasting cycles pursuing checkpoints that will likely have been visited by other agents before them, while in cluster environments performance increases due to agents being attracted to the cluster quickly as soon as one agent has discovered it. The performance difference between communicating and non-communicating agents is particularly pronounced in the high density condition (of 40 checkpoints), where the satiation level limits agents to 10 visits (thus 4 agents are required to visit all checkpoints in the cluster; in the non-communication conditions this means that the cluster needs to be discovered independently at least four times, which can take a while). The performance improvement is less pronounced in the 10 checkpoint cluster (given that one agent could visit them all). In the random condition, the trend is in the opposite direction: the performance decrease is higher in the low density condition than in the high density condition, again for the reason that agents will chase checkpoints that other agents are likely to get first.

Note that the above results are based on absolute performance differences, as communicating agents are not charged any penalties for their communication mechanism (including processing and representational resources and computation time, additional sensors and effectors, etc.). The costs involved in communication, especially the cost for sensitive sensors with large sensory ranges (as is required for communication to be beneficial) can be quite expensive (e.g., see (Schermerhorn and Scheutz, 2006, 2007b) for comparison of the various tradeoffs). Hence, whether communication based on large communication ranges will evolve for high density cluster environments is an open question, but we can already say that if it evolves then it will use signaling and not messaging, given that there was no performance difference between signaling and messaging, but messaging requires and incurs much greater costs.

It is curious, then, that a small number of insect species – the social insects – did evolve messaging communication to communicate the locations of resources to their peers. This could be because these agents depart from and return to a common location which makes a difference in their foraging patterns that could favor communication. Moreover, honey bees (Capaldi and Dyer, 1999; Menzel et al., 1998) could not use the signaling mechanism for food employed in this study when they are at the hive. Interestingly, Dornhaus and Chittka (2004) provide evidence that bees can survive just fine *without communication* (i.e., when their ability to communicate is suppressed) depending on the food distribution and food quality in their environment. Hence, we would expect to observe this contingency of communication being beneficial depending on the distribution of food in the environment in modified *t*-MATES tasks if the additional constraint of always having to return to a common "base checkpoint" after visiting a "field checkpoint" is taken into account; and in fact Schermerhorn and Scheutz (2005) provides preliminary evidence from a related task that suggests that this might indeed be the case.

Related Work

Several authors have investigated the utility of communication or signaling in various tasks. There are, for example, purely game-theoretic studies that explore the role of communication in *coordination games* and show that non-binding, pre-play communication can can increase the probability of playing the Pareto-dominant strategy (e.g., Cooper et al. (1992)). Arkin et al. find that communication can aid coordination in robotic retrieval tasks (Arkin and Hobbs, 1992; Wagner and Arkin, 2004). Conversely, Werger et al. (Werger and Mataric, 2001) and Quinn et al. (2003) found communication to be unnecessary to achieve task formation in a system which uses behavior-based mechanisms to generate cooperative behaviors. However, the employed tasks are sometimes very different from *t*-MATES tasks making it difficult to compare the outcomes.

MacLennan found that communication will evolve in a task requiring coordinated behavior when agents are rewarded for agreeing on the meaning of a signal (MacLennan, 2002). However, this rewards agents directly for communication rather than demonstrating that communication can be beneficial to performance of a separate task. Similarly, Levin (1995), using a genetic algorithm approach with a fitness function that explicitly favors the evolution of communication, found it possible to progressively improve the ability of agents to correctly interpret other agents' communications. Noble and Cliff (1996) extend MacLennan's work to show that a structured language will evolve based on the benefits of communication.

Quinn (2001) describes experiments in which artificial agents evolve a signaling mechanism in the absence of predetermined communication channels. Pairs of simulated robotic agents starting within sensor range of one another are

given the task of moving in their environment while staying within sensor range. Here, a signaling system evolved which was not part of the fitness function, but rather measured absolute task performance and behavior coordination.

Marocco et al. (2003) and Cangelosi et al. (2004) describe experiments with simulated robots which are required to recognize a sphere and a cube in order to maximize contact with the sphere and minimize contact with the cube. Once agents identify an object, they can communicate that information to other agents, allowing them, for example, to avoid contact with the cube without using first-hand proprioceptive information. Communication between parents and offspring was found to evolve.

Ackley and Littman (1991) note that models in which the speaker as well as the listener benefits from communication produces an unrealistic environment in which many observed phenomena related to communication do not make sense. In their model, agents can share information about nearby food and predators. They found that, in some conditions, communication can improve performance on the survival task (i.e., locating food and avoiding predators).

Noble (1999) examines various communication games to determine under what circumstances communication will evolve. Agents have the opportunity to communicate during encounters between a signaler and a receiver, and they are rewarded when the receiver responds appropriately to the signal. In this study, communication was found to evolve when the signaler receives a net reward. However, when signalers are not rewarded for receivers' successes, communication did not evolve.

Grim et al. (2002) examine the benefit of communication in a survival task requiring agents to consume food when present and hide from predators. Agents can share information about food and predators with neighboring agents. They find that communication will evolve in the absence of a cost for signalling, but that adding such a cost, even just to the level of 2% of the benefit of eating or the cost of predation, affects the viability of communication.

Reggia et al. (2001) found similar results to those presented above with regard to the effect of checkpoint (food) distribution. Their study examines only indexical signalling, and they do not examine the effect of sensory range or communication range. However, different from our study, their model includes predators, an important risk factor that should further decrease the utility of communication.

Conclusion

We investigated the limitation of communication for improving the performance of simple agents in timed multi-agent territory exploration tasks with satiation. Different from most work on the evolution of communication, our results paint a nuanced picture of the utility of communication. In environments with no structure communicating agents with limited communication range do not perform better than non-communicating agents, and unlimited communication range can result in a significant performance drop. In cluster environments, only communication with unlimited communication range (that covers the whole environment) leads to better performance (which may or may not be biologically plausible depending on the type of environment and type of sensory modality). More importantly, there was no significant performance difference between signalling and messaging agents suggesting that if communication were to evolve, then it would be of the simplest possible form using only indexical broadcast signals rather than non-indexical messages.

While the above results might seem largely negative from the perspective of someone arguing for the utility of communication and the likelihood of it evolving for biologically plausible foraging tasks, the results about the utility and high level of performance of simple agents with at best simple means of signalling in t-MATES tasks is highly relevant for and might find direct applications in a variety of engineering tasks where low-cost solutions or solutions with a high (if not the best) relative performance are of major interest (for example, in energy-efficient mobile rovers that explore the surfaces of planets, expandable autonomous mine-sweeping robots that search an area for mines and explode with the mine by driving over it, or unmanned surveillance vehicles that need to check various locations in an environment as they are dynamically reported as quickly as possible).

In addition to the already mentioned constraint of imposing a hive-like home base for foragers, we see at least two promising directions for further investigating the benefits and limits of communication in t-MATES tasks. The first concerns the idea of "structure in the environment" and the degree to which communication can benefit from it. Specifically, it would be interesting to define a measure of "structure" (ideally information-theoretic) for environments that gets at the kinds of distributions of checkpoints that would favor communication.

Another direction concerns the coordination of agent behavior, in which communication could play an facilitatory role. The idea here is to impose additional task constraints on the t-MATES task such as requiring multiple agents to visit the same checkpoint at the same time (as would be required for mating) to isolate scenarios where coordination can be significantly improved via communication (Schermerhorn and Scheutz (2007a) already started an exploration of these tradeoffs in a related task).

Acknowledgments

The authors would like to thank Colin Allen and three anonymous reviewers for their helpful suggestions for improving the paper.

References

Ackley, D. and Littman, M. (1991). Altruism in the evolution of communication. In Brooks, R. and Maes, P., editors, *Artificial Life IV: Proceedings of the Fourth Workshop on Artificial Life*, pages 40–48, Cambridge, MA. MIT Press.

Arkin, R. C. and Hobbs, J. D. (1992). Dimensions of communication and social organization in multi-agent robotic systems. In *Proceedings of the Second International Conference on Simulation of Adaptive Behavior*, pages 486–493.

Cangelosi, A., Riga, T., Giolito, B., and Marocco, D. (2004). Language emergence and grounding in sensorimotor agents and robots. In *First International Workshop on Emergence and Evolution of Linguistic Communication*, Kanazawa, Japan.

Capaldi, E. A. and Dyer, F. C. (1999). The role of orientation flights on homing performance in honeybees. *The Journal of Experimental Biology*, 202:1655–1666.

Cocroft, R. B. (2005). Vibrational communication facilitates cooperative foraging in a phloem-feeding insect. *Proceedings of Biological Science*, 272(1567):1023–1029.

Cooper, R., DeJong, D. V., Forsythe, R., and Ross, T. W. (1992). Communication in coordination games. *The Quarterly Journal of Economics*, 107(2):739–771.

Cunnane, S. C. (2006). *Survival of the Fattest: The Key to Human Brain Evolution*. World Scientific.

Dittus, W. P. (1984). Toque macaque food calls: Semantic communication concerning food distribution in the environment. *Animal Behaviour*, 32(2):470–477.

Dornhaus, A. and Chittka, L. (2004). Why do honey bees dance? *Behavioral Ecology and Sociobiology*, 55(4):396–401.

Fellers, G. M. (1979). Mate selection in the gray treefrog, Hyla versicolor. *Copeia*, 1979(2):286–290.

Grim, P., Kokalis, T., Tafti, A., and Kilb, N. (2002). Evolution of communication with a spatialized genetic algorithm. *Evolution of Communication*, 3(2):105–134.

Levin, M. (1995). The evolution of understanding: A genetic algorithm model of the evolution of communication. *BioSystems*, 35:167–178.

MacLennan, B. (2002). Synthetic ethology: A new tool for investigating animal cognition. In Bekoff, M., Allen, C., and Burghardt, G. M., editors, *The Cognitive Animal: Empirical and Theoretical Perspectives on Animal Cognition*, pages 151–156. MIT Press.

Marocco, D., Cangelosi, A., and Nolfi, S. (2003). The emergence of communication in evolutionary robots. *Philosophical Transactions: Mathematical, Physical and Engineering Sciences*, 361(1811):2397–2421.

Menzel, R., Geiger, K., Joerges, J., Müller, U., and Chittka, L. (1998). Bees travel novel homeward routes by integrating separately acquired vector memories. *Animal Behavior*, 55(1):139–152.

Noble, J. (1999). Cooperation, conflict and the evolution of communication. *Adaptive Behavior*, 7(3-4):349–370.

Noble, J. and Cliff, D. (1996). On simulating the evolution of communication. In Maes, P., editor, *Proceedings of the Fourth International Conference on Simulation of Adaptive Behavior*, pages 608–617, Cambridge, MA. MIT Press.

Perconti, P. (2002). Context-dependence in human and animal communication. *Foundations of Science*, 7:341–362.

Quinn, M. (2001). Evolving communication without dedicated communicatin channels. In *Proceedings of ECAL 2001*, pages 357–366.

Quinn, M., Smith, L., Mayley, G., and Husbands, P. (2003). Evolving controllers for a homogeneous system of physical robots: Structured cooperation with minimal sensors. *Philosophical Transactions of the Royal Society of London, Series A: Mathematical, Physical and Engineering Sciences*, 361:2321–2344.

Reggia, J. A., Schultz, R., Wilkinson, G. S., and Uriagereka, J. (2001). Conditions enabling the evolution of inter-agent signaling in an artificial world. *Artificial Life*, 7(1):3–32.

Schermerhorn, P. and Scheutz, M. (2005). The effect of environmental structure on the utility of communication in hive-based swarms. In *IEEE Swarm Intelligence Symposium 2005*, pages 440–443. IEEE Computer Society Press.

Schermerhorn, P. and Scheutz, M. (2006). Social coordination without communication in multi-agent territory exploration tasks. In *Proceedings of the Fifth International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS-06)*, pages 654–661, Hakodate, Japan.

Schermerhorn, P. and Scheutz, M. (2007a). Investigating the adaptiveness of communication in multi-agent behavior coordination. *Adaptive Behavior*, 15(4):423–445.

Schermerhorn, P. and Scheutz, M. (2007b). Social, physical, and computational tradeoffs in collaborative multi-agent territory exploration tasks. In *Proceedings of the First IEEE Symposium on Artificial Life*, pages 295–302.

Scheutz, M. and Schermerhorn, P. (2005). Predicting population dynamics and evolutionary trajectories based on performance evaluations in alife simulations. In *Proceedings of GECCO 2005*, pages 35–42. ACM Press.

Scheutz, M. and Schermerhorn, P. (2006). Adaptive algorithms for the dynamic distribution and parallel execution of agent-based models. *Journal of Parallel and Distributed Computing*, 66(8):1037–1051.

Scheutz, M., Schermerhorn, P., Connaughton, R., and Dingler, A. (2006). Swages–an extendable parallel grid experimentation system for large-scale agent-based alife simulations. In *Proceedings of Artificial Life X*, pages 412–418.

Wagner, A. and Arkin, R. (2004). Multi-robot communication-sensitive reconnaissance. In *Proceedings of the 2004 IEEE International Conference on Robotics and Automation*.

Werger, B. B. and Mataric, M. J. (2001). From insect to internet: Situated control for networked robot teams. *Annals of Mathematics and Artificial Intelligence*, 31(1–4):173–198.

Analysing honeybees' division of labour in broodcare by a multi-agent model

Thomas Schmickl[1,2] and Karl Crailsheim[1,3]

[1]Department for Zoology, Karl-Franzens-University Graz, Universitätsplatz 2, A-8010 Graz
[2]thomas.schmickl@uni-graz.at
[3]karl.crailsheim@uni-graz.at

Abstract

We describe a multi-agent model of a honeybee colony and show several applications of the model that simulate experiments that have been performed with real honeybees. Our special emphasis was on the decentralized, self-organized regulation of brood nursing, which we successfully simulated: We found that brood manipulations, food-deprivation experiments and colony-size manipulations can be explained by the mechanisms we implemented into our model described here. Our agents can perform various tasks (foraging, storing, nursing). The model is spatially resolved, and contains a designated broodnest area as well as a designated honey/nectar storage area. All bees (and larvae) consume nectar/honey at a task-specific rate, allowing us to track the flow of nectar through the colony. Several kinds of stimuli, which are important for division of labour, were modelled in detail: dances, contact stimuli and chemical signals.

Introduction

The ability of social insects to divide the colony's work via specialisation, polyethism and task partitioning has fascinated scientists since decades. For example, early work of (Lindauer, 1952; Rösch, 1952; Sakagami, 1953) described the impressive ability of honeybees to specialize on different tasks based on an age-based scheme (temporal polyethism). In recent years, several conceptual models have been proposed to explain the basic proximate mechanisms that lead to division of labour in social insects in general, see Beshers and Fewell (2001) for a review, and for honeybees in detail: Age-based polyethism (Seeley, 1982; Johnson, 2005), regulation by queuing delays (Seeley, 1992), foraging for work theory (Franks and Tofts, 1994), threshold reinforcement (Theraulaz et al., 1998), and social inhibition (Beshers et al., 2001). Many of these concepts were also investigated by mathematical models and computer simulation (Anderson, 1998; Gautrais et al., 2002). One the one hand, these models focused very well on the specific key process that they were built to examine, on the other hand, they lack many specific details that are significantly affecting the behaviour of social insects. To fill this gap and to allow specific simulation of honeybees' division of labour, we constructed a multi-agent model of a honeybee colony that builds on the ideas of the before mentioned models and incorporates several important honeybee specific details:

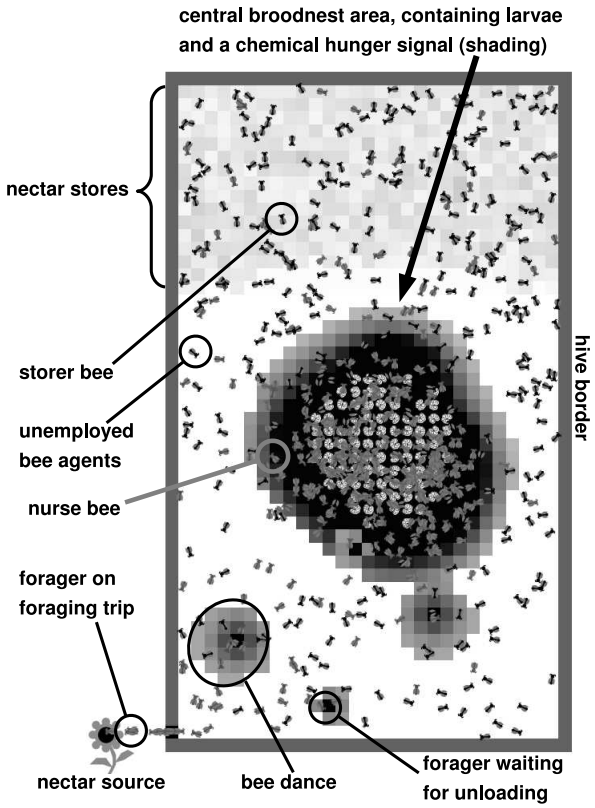


Figure 1: Typical screenshot of our multi-agent simulation at run-time. Bee agents move across the hive space and, depending on their history, emit several sorts of stimuli: waggle/tremble dances and offering signals. Hungry larvae also emit chemical hunger stimuli, which diffuse in the central broodnest area. Unemployed bees can react to all of these stimuli and switch to one of the modelled task cohorts.

1. A typical spatial distribution of brood and food in the hive.

2. Complex behavioural programs of specialized workers.
3. Characteristics of the spreading of different kinds of stimuli (chemicals, sounds/vibration, light).
4. Agents physiology (energetic expenditures).
5. Flow of nutrients among the agents and the combs.

Our multi-agent model (named TaskSelSim) is implemented in NetLogo (Wilensky, 1999). The implementation of the model (equations, parameter values) have been described in detail in (Schmickl and Crailsheim, 2008b). In this article, we describe the models implementation in a lower degree of details and concentrate on those details that are important for the focal questions described here: How does the brood status affect the division of labour in the simulated honeybee colony and how does the colony status affect the brood nursing. Other aspects of division of labour (effects of selective removals/additions of task cohorts) were already investigated in (Schmickl and Crailsheim, 2008a), thus we did not perform such experiments in the study presented here.

Brood nursing (feeding brood with honey, pollen and pollen derived gland products) is a distributed process in a honeybee colony: Each specialised nurse bee feeds many larvae sequentially and each larva is fed by many nurses. The brood is allocated in a central area in the hive, one larva is occupying one comb cell. We studied the nursing of brood in honeybees with several ethological studies, see Schmickl and Crailsheim (2002). These experiments showed that brood nursing is regulated in a homoeostatic, adaptive way. It was shown by (Huang and Otis, 1991b) that nurse bees preferentially inspect comb cells that are occupied by larvae and that artificially starved larvae receive preferential nursing (Huang and Otis, 1991a). The hunger state of a larva is communicated to nurse bees by emission of chemical substances (pheromones). All of these facts were incorporated in our model (together with an implementation of the foraging process and the nectar storing process), to generate a model that is able to integrate many (separately derived) hypothesises of honeybees' regulation of division of labour into one single consistent process.

The Model

Our model depicts one honeybee colony consisting of agents (adult bees, larvae), stimuli (dances, contact stimuli, chemical signals, light) and resource stores (nectar and honey is used synonymously). The hive space is modelled in discrete patches (31 x 52) but the adult bee agents can move across these patches in continuous motion. The intensity of local stimuli is modelled discrete, following the grid of patches that represent the comb cells. Figure 1 depicts the typical spatial distribution of agents, stimuli and resource stores.

Within each time step, the following functions are executed iteratively:

1. All patches update their status (decay of chemicals).
2. All agents emit stimuli, chemicals are diffused.
3. All agents consume nectar.
4. All adult agents decide to engage or to give up a task.
5. All adult agents perform behaviour according to their task.

Modelled Tasks

Depending on the task the bees are engaged in, they perform the following behavioural programs:

Unemployed bees: These bees move randomly in the hive. In our model, bees had to switch to this unemployed state at least for one time step before they could engage in a different task.

Forager bees: These bees leave the hive with a low (but sufficient) crop load. They fly to the nectar source, fill up their crop and fly back to the entrance. There they emit the unloading stimulus to attract nearby storage bees which take over the nectar load. After some time of random movement in the hive, they can perform a waggle or a tremble dance (see below for more details). Afterwards, they leave the hive again towards the nectar source.

Storer bees: These bees wait near the entrance for returning foragers. They take the crop load of returning foragers and head towards the storage area (see Figure 1). They drop their nectar load there and head back towards the entrance.

Nurse bees: These bees navigate (uphill) in the chemical stimulus emitted by hungry larvae. If they are located on a patch containing a hungry larva, they start to feed this larva until it is saturated or the nurse is almost empty. These feedings last for several time steps.

Larvae: The brood resides in cells (patches) in the central broodnest area (see figure 1). Larvae cannot move. If they have low nectar reserves, they emit a chemical hunger signal. See below for more details.

Modelling the Stimuli

In our model it is important that stimuli differ significantly in their dynamics and in their range: Contact stimuli are emitted by returning forager bees to attract storer bees to take over the nectar load. These signals have a short range ($r = 1$) only and stop immediately after the forager is entirely unloaded. Depending on the waiting period a forager searched for a storer bee, it then performs either a 'waggle dance' ($T_{search} <= 20$) to recruit more forager bees or a 'tremble dance' ($T_{search} >= 50$) to recruit more storer

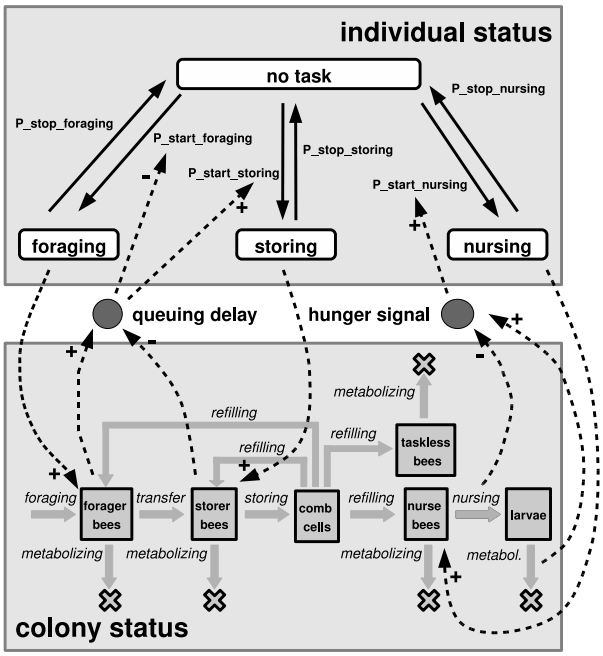


Figure 2: The flow of nectar, bees' metabolism and task selection of our agents. Top: Individual task selection depicted as a state automaton. Middle: Most important regulation feedbacks. Bottom: Task cohorts as compartments in the flow of nectar in the colony. Rounded boxes represent individual tasks. Solid arrows indicate task switches. Dashed arrows indicate dependencies ('A is affecting B'). Rectangular boxes represent worker cohorts, larvae or combs. Solid arrows indicate nectar flows. The flower represents a nectar source, the cross-like symbols represent sinks.

bees. Both stimuli spread wider ($r = 3$) and decay non-linearly ($\frac{1}{d}$) with increasing distance from the emitting bee. As soon as the dancing bee stops, the emitted dance signal disappears also from all other patches immediately. In contrast to that, the chemical stimuli emitted by larvae stay much longer and spread wider: They diffuse to all nearby patches and decay slowly:

$$\frac{\partial C(\mathbf{x})}{\partial t} = D\nabla^2 C(\mathbf{x}) - \mu C(\mathbf{x}) + \alpha_i(t) L_{hungry}(\mathbf{x}), \quad (1)$$

where $C(\mathbf{x})$ is the local concentration of hunger pheromone at position $\mathbf{x}$, μ is the rate of pheromone decay, α is the addition rate of pheromone produced by a hungry larva. $L_{hungry}(\mathbf{x})$ is set to 1 in case that there is a hungry larva at position $\mathbf{x}$, else it is set to 0.

In case that the larva at position $\mathbf{x}$ has a nectar reserve below the hunger threshold $cr_{low} = 0.25$, alpha scales linear from 1 down to 0, as described in equation 2.

$$\alpha_i(t) = 1 - \frac{v_i(t)}{cr_{low} \cdot capacity_{larva}} \quad (2)$$

If the larva has more nectar in its reserves, then the value of $\alpha_i(t)$ is set to 0. A hungry larva at position $\mathbf{x}$ is referred as larva i. The nectar reserve of this larva is described as $v_i(t)$, the maximum storing capacity of a larva was set to $capacity_{larva} = 0.33$. The 'diffusion term' was implemented numerically (and discrete): we used the build-in function "diffuse" available in the NetLogo programming environment. The light stimulus decreases linearly with increasing distance from the hive's entrance and is used for navigation of foragers for leaving the hive and for navigation of storer bees for approaching the entrance area and for approaching the honey area. Nurse bees navigate uphill in the chemical pheromone field to find hungry larvae to feed and move towards darker areas to find honey cells for refills.

Simulated physiology

An adult bee can hold a maximum of 1 unit of crop load. A larva can hold 0.33 units at maximum. Adult bees consume their nectar loads at a low rate of $cr_{low} = 0.0004\ units/step$, flying foragers consume at a higher rate $cr_{high} = 0.001\ units/step$. Larvae consume nectar at the rate $cr_{larva} = 0.0004\ units/step$. If an agent (bee or larva) runs out of nectar, it dies and is removed from the system. The bottom of figure 2 shows these consumption flows.

Modelling Division of Labour

The most important aspect in our model is the implementation of the task selection mechanism. We followed the approaches of Gautrais et al. (2002) and implemented a threshold based system. Each type of local stimulus can motivate an unemployed adult bee agent (task = 'no-task') to join one of the tasks $m \in \{$ 'foraging', 'storing', 'nursing' $\}$. See figure 2 (upper part) for the possible task transitions. Whenever one of these stimuli exceeds an individual threshold of an agent i located on that patch x, the agent engages in the associated task m. Each of these thresholds is modelled in a non-linear manner, as is shown by equation 3. $p_{i,m}$ models the likelihood to engage in task m in one time step. $s_{x,m}$ is the local intensity of the task-associated stimulus. $\Theta_{i,m}$ is used to shift the threshold individually up and down, n is used to express the degree of non-linearity in these behavioural decisions.

$$p_{i,m} = \frac{s_{x,m}^n}{s_{x,m}^n + \Theta_{i,m}^n} \quad (3)$$

Employed bees switch back to the unemployed state with probabilities of $\lambda_{'nursing'} = \lambda_{'storing'} = 0.005/step$ and $\lambda_{'foraging'} = 0.001/step$. To allow specialisation within this system, the levels of the thresholds are adapted individually during run time. In the case that an unemployed agent engages in task m', the $\Theta_{i,m}$ is reduced by ξ_m, making it

more likely that the agent will engage in this task in future. Whenever an unemployed agent does not engage in a task, the corresponding threshold is increased by φ_m, making it more unlikely that these behaviours will be triggered later on. In our simulations, all values of ξ were set to $\xi = 0.1$ and all values of φ were set to $\varphi = 0.001$. It was shown in (Schmickl and Crailsheim, 2008b) that these parameter values lead to plausible division of labour. In our simulations we used $n = 2$ for all agents and all agents initially started with Θ values of 0.001 for all tasks. During run time, values of Θ were confined between 0 and 1.

Initial Conditions

Our simulations were conducted with 700 adult bee agents and 100 larvae. The larvae were distributed randomly (normal distribution) around the center of the hive. All adult agents started in randomized positions and with randomized headings. Their initial task was set to 'no-task'. All agents had (uniformly) randomized crop loads.

Results

This article focuses on the aspects associated with the regulation of brood nursing, thus we manipulated the ratio of adult bees to larvae in our simulation experiments described here. We first simulated 10000 time steps of an undisturbed colony, to allow the colony to reach equilibrium in brood supply and in division of labour. At time step 10000, the whole simulation state was saved on hard disk. Starting with this saved configurations several perturbations were performed (addition of brood, removal of adult workers) and the resulting changes in task cohorts were measured. In these experiments, all adult bees started with Θ values of 0.

Perturbations of the adult-to-brood ratio

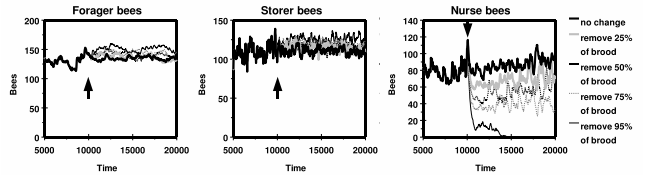


Figure 3: Removal of brood affects the size of the nursing cohort strongly. The additional workforce that gets available from abandoning brood care affects also the size of the other working cohorts. The arrows indicate the timing of the perturbation. Graphs show mean values (N=6).

The more brood was removed at time step 10000, the less bees performed the nursing task. This high abandonment from nursing made more bees available for the tasks of storing and for the foraging task, as can be seen in figure 3.

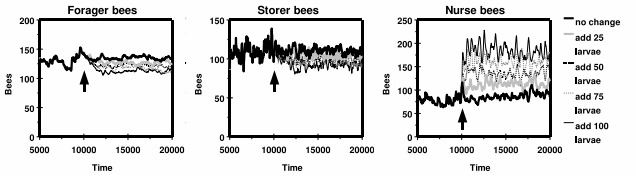


Figure 4: Addition of brood affects the size of the nursing cohort strongly. This binds additional workforce to the task of nursing, what in turn affects also the size of the foraging cohort and of the storing cohort. The arrows indicate the timing of the perturbation. Graphs show mean values (N=6).

Analogously we observed a significant increase of the size of the nursing cohort as we spontaneously added brood to the colony at time step 10000. This reduced the number of unemployed bees, in turn affecting also the task equilibrium of foraging bees and of storing bees, as shown in figure 4.

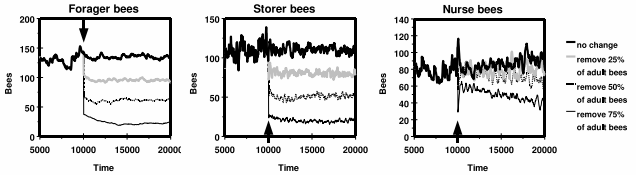


Figure 5: Removal of worker bees affected all task cohorts. The cohort of nurses bees was strongly affected only with the more extreme removal of worker bees. The arrows indicate the timing of the perturbation. Graphs show mean values (N=6).

As figure 5 shows, the removal of adult bees strongly affected all three task cohorts. The removal was a random pick across all task cohorts. While the cohorts of foragers and storers were affected significantly by all worker losses, the cohort of nurse bees was affected significantly only by the bigger losses of worker bees.

Nectar economics

During the experiments shown in the figures 3, 4, and 5, the colony structure was significantly altered at time step $t = 10000$. Since we also modelled the flow of nectar (nectar income and consumption), we could also investigate how these alteration affected the colony's nectar economics. Figure 6 shows these results: The removal of brood strongly enhanced the colony's net nectar gain, as a significant sink for nectar was decreased. In contrast to that, the addition of brood increased this important nectar sink, what had a detrimental effect on the colony's nectar economics. This effect was also observed by leaving the sink unchanged but

by decreasing the foraging workforce, as it happened by the removal of adult bees.

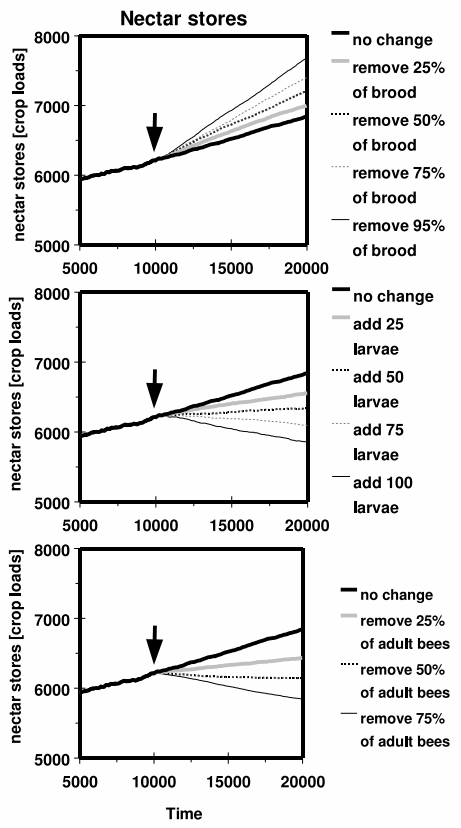


Figure 6: Removal of brood leads to strong increases of the colony's net nectar gain over time. Addition of brood or removal of workers lead to strong decreases of the colony's net nectar gain. The arrows show the timw of the perturbation. All graphs show mean values (N=6).

Scaling properties of division of labour

As shown above, colony manipulations affected the task cohorts and the colony's net nectar gain. As figure 7 shows, forager and storer cohorts are severely affected (high steepness of the regression curve) by removal of worker bees. Nurses and net nectar gain is affected by both, brood manipulation and by adult removal. Almost all correlations between perturbation strength and resulting cohort sizes were found to be linear. It has to be mentioned that the steepness of regressions varied significantly, what points towards different sensitivities of cohorts to perturbation types.

Nursing on the individual level

Empirical studies showed that nursing of brood is regulated in a supply-demand driven process. Huang and Otis (1991a) performed an interesting study, where they prevented a set

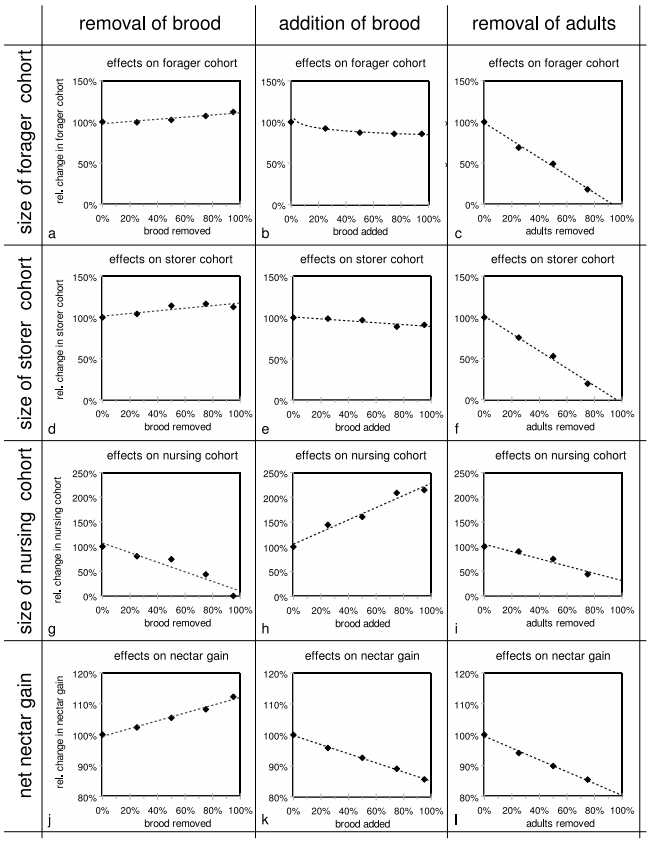


Figure 7: The effects of all perturbations scaled in most cases linearly with the strengths of the perturbations. The figures indicate the relative difference of the end result of the simulations ($t = 20000$) compared to the undisturbed control simulations. Graphs show mean values (N=6).

of 4-day old larvae from being fed by nurse bees with a cage that was placed around the larvae. They found that these starved larvae were fed preferentially by nurse bees after the cage was removed. We were interested whether or not such effects could be observed also in our model. Thus we (virtually) put a cage around a central spot in the center of the broodnest, what prevented the bees from entering this area. After 400 time steps with the cage preventing feedings, a fraction of the larvae died (see figure 8). The remaining larvae were fed preferentially during the first 1000 time steps after the cage was removed (figure 9). Later on, the formerly starved larvae were fed on average on the same level, but the mode of nursing was still altered due to experimental manipulation: Feedings were performed in a more oscillating manner, suggesting that disturbances of brood nursing could cause long-term alterations in the colonies nursing behaviour. Table 1 sums up the mean number of feedings per time-slot (which was 100 time steps wide) per larva for both

zones (central cage area, peripheral 'no-cage' area):

Phase (time steps)	Central area (cage)	Peripheral area (no-cage)
Pre (0 - 1500)	0.24 ± 0.16	0.24 ± 0.13
Experiment (1501 - 1900)	0 ± 0	0.24 ± 0.14
Post (1901 - 2900)	0.37 ± 0.64	0.22 ± 0.11
End (2901 - 5000)	0.24 ± 0.34	0.23 ± 0.08

Table 1: Statistical comparison of all 4 phases in both experimental zones. Means values were gained from all larvae in the corresponding zone per 100 time steps. $\pm$ indicates the corresponding standard deviations in these datasets.

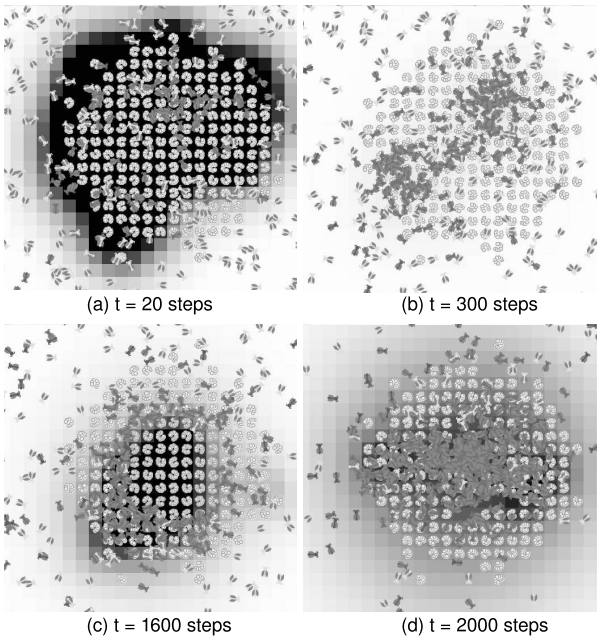


Figure 8: (A) Initially the brood starts hungry and emits a lot of hunger signals. (B) After 300 time steps, the nurses satisfied the brood and kept it on a rather fed status. (C) At time step $t = 1500$, the (virtual) cage was installed around the central brood nest area. At $t = 1600$, many hungry larvae can be found in the center, emitting strong hunger stimuli. (D) After removal of the cage at $t = 1900$, the central brood is either dead (removed) or very hungry. At $t = 2000$, nurses aggregate in this area and feed frequently.

Specialization in the nursing task

In additional simulation runs, the development of thresholds in big and small colonies with low and high brood state was observed. In these experiments, all Θ values were initially

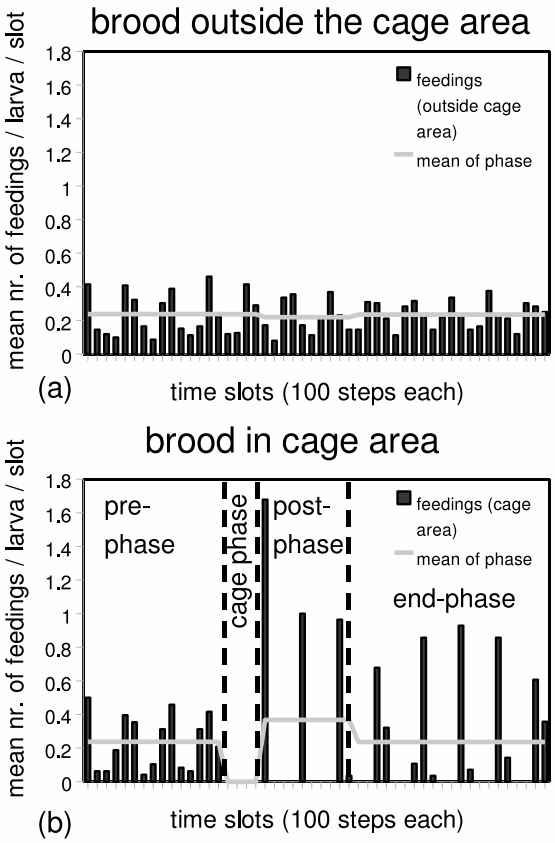


Figure 9: (A) In the area outside the cage-zone, larvae are fed in all experimental periods on the same average rate. (B) In the pre-experimental period, central and peripheral larvae are fed on the same level. During the cage-period, no feedings can occur. During the first 1000 time steps after the cage's removal, the remaining starved larvae are fed preferentially. Also the oscillations increased significantly. In the final period, the level of feedings returns to the initial value, but the rhythmicity stays on an increased level.

randomized uniformly between 0 and 1. After 10000 time steps, the values of $\Theta_{nursing}$ were measured. To speed up the specialization process, all ξ_m values were increased to 0.2 in these experiments.

The first simulation we performed was with the colony status we used also in the simulations described in the previous sections (700 adults, 100 larvae). As can be seen in figure 10, approx. 10% of the bees developed into highly specialized nurses. The majority of bees developed into highly specialized storage bees or into 'partly-specialized' storage bees. Although we observed between 65 and 120 forager bees throughout the run-time of the simulation, the observed degree of specialization for this task was not comparable to

the degree of specialization of the other tasks. In (Gautrais et al., 2002) it is predicted by another model, that the degree of specialization increases with colony size. To investigate this question also with our model, we scaled down the colony size (adults and brood) by the factor $\frac{1}{4}$. Please note that the nursing workload per bee was kept constant. As can be seen in figure 11, the degree of specialization decreased, especially with the nursing task and with the storing task.

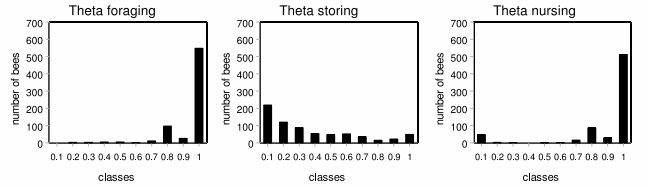


Figure 10: Degree of specialization to our modelled tasks in a simulated colony consisting of 700 adult bees and 100 larvae. Low theta values ($\Theta <= 0.2$) are interpreted as 'high degree' of specialization. Values of $0.2 < \Theta < 0.8$ are interpreted as partially specialized bees. Higher values of Θ are interpreted as bees not specialized to the specific task.

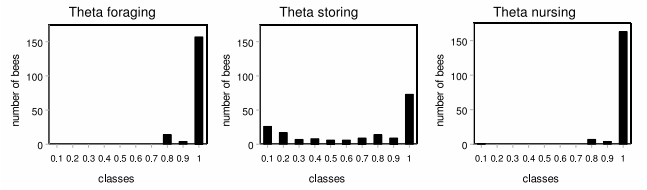


Figure 11: Degree of specialization to our modelled tasks in a simulated colony consisting of 175 adult bees and 25 larvae. Low theta values ($\Theta <= 0.2$) are interpreted as 'high degree' of specialization. Values of $0.2 < \Theta < 0.8$ are interpreted as partially specialized bees. Higher values of Θ are interpreted as bees not specialized to the specific task.

In a final simulation experiment, we doubled the work load per adult bee compared to the settings shown in figure 10. As can be seen in figure 12, this increased preferentially the degree of specialization of nurse bees, as the number of highly-specialized nurses more than doubled.

Discussion

We showed that a threshold-based model can suffice to simulate honeybee-specific division of labour. By performing and analyzing our simulations, we learned that having a reactive system of task cohorts, which is able to react plausibly to perturbations of colony structure, is not a guarantee

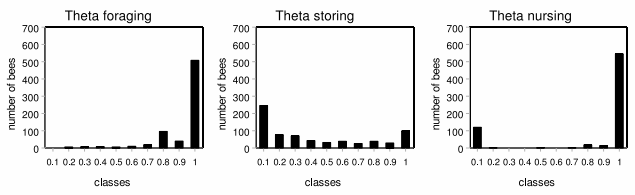


Figure 12: Degree of specialization to our modelled tasks in a simulated colony consisting of 700 adult bees and 200 larvae. Low theta values ($\Theta <= 0.2$) are interpreted as 'high degree' of specialization. Values of $0.2 < \Theta < 0.8$ are interpreted as partially specialized bees. Higher values of Θ are interpreted as bees not specialized to the specific task.

for having task specialisation and de-specialization of workers: As our figures 3, 4, 5 and 6 show, our modelled colony reacts very plausibly to the induced perturbations. When we investigated whether or not the nursing received by individual larvae after a deprivation experiment is predicted plausibly by our multi-agent simulation, we found that the nursing regulation reflects empiric results very well.

Although we found division of labour and specialization of hive-bees (nurses, storers), figure 10 tells us, that the foraging cohort did not show the expected high degree of specialisation in our simulations. The threshold-response system was able to model specialisation of nurses by chemical brood stimuli at a very high degree. We found also many bees highly specialized to the task of storing. But also a high number of 'semi-specialized' storers ($0.2 < \Theta < 0.8$) was found. Most foragers showed only a low degree of specialization, indicating that foragers are not often re-recruited to the foraging task after they abandoned from foraging. We had between 65 and 120 foraging bees present at all times, but almost all foraging bees performed just one or two consecutive engagements, having several round-trips. The fact that we still observed task cohorts that reacted adaptively to perturbations can be reasoned by the equilibria that emerge: Foraging has a high turn-over number, that means foragers that quit the task once in our model are not often re-recruited. But simultaneously many other bees, that performed other tasks before, are recruited to the task of foraging. Obviously, this suffices to allow an adaptive equilibrium-based division of labour.

We conclude that threshold reinforcement (Theraulaz et al., 1998; Gautrais et al., 2002) is well suited to produce plausible specialization in tasks that are associated with very durable, time-persistent and spreading stimuli, like the pheromones that stimulate nursing behavior. Also the storing task has a high density of stimuli (tremble dances, every returning forager emits the 'storing' stimulus), but foraging is induced only by the (relatively rare) waggle dances.

As these dances do not occur at a comparably high frequency (only some foragers perform a waggle dance), re-recruitment is hard to explain by just this stimuli alone. In nature, foragers are a well specialized group in the honeybee society, thus we can assume that we will have to incorporate other additional factors to achieve the observed high specialisation of forager bees: shaking signals, stop signals and a (probably age-related) higher predisposition for the foraging task, as it can be easily implemented into our model by a slight downward-bias of $\Theta_{foraging}$ in a specific group of bees which represent 'older' bees. In addition, motivation for foraging can be also influenced by physiological properties of the bees, which reflect often characteristic hive conditions, as was demonstrated in (Camazine, 1993). We conclude that, incorporating additional regulatory systems and motivational aspects of bees, as for example temporal polyethism (Seeley, 1982; Johnson, 2005) can significantly improve the models predictions concerning foraging specialization.

For interpreting the observed differences in task specialization, we had to consider also the regions of the hive in which the recruited workers tend to stay. These regions include a specific mixture of stimuli, thus determining also the likelihoods to switch to other tasks. Such effects are discussed in the 'foraging-for-work'-theory, as it is described in (Franks and Tofts, 1994). By working with and on our model we learned that honeybee-specific division of labour cannot be modelled with the threshold-reinforcement model alone. We developed the idea that several of the discussed concepts of honeybees' division of labour have to be implemented into one single model, which then represents an integrative approach to understand honeybee's' division of labour. By extending our model in these directions, we will pursue this scientific goal.

Acknowledgements

This work was supported by: EU-IST FET project 'I-Swarm', no. 507006; EU-IST-FET project 'SYMBRION', no. 216342; EU-ICT project "REPLICATOR", no. 216240. Austrian Science Fund (FWF) research grants: P15961-B06 and P19478-B16.

References

Anderson, C. (1998). Simulation of the feedbacks and regulation of recruitment dancing in honey bees. *Advances in Complex Systems*, 1:267–282.

Beshers, S. N. and Fewell, J. H. (2001). Models of division of labor in social insects. *Annual Review of Entomology*, 46:413–440.

Beshers, S. N., Huang, Z. Y., Oono, Y., and Robinson, G. E. (2001). Social inhibition and the regulation of temporal polyethism in honey bees. *Journal of Theoretical Biology*, 213:461–479.

Camazine, S. (1993). The regulation of pollen foraging by honey bees: how foragers assess the colony's need for pollen. *Behavioral Ecology and Sociobiology*, 32:265272.

Franks, N. R. and Tofts, C. (1994). Foraging for work: How tasks allocate workers. *Animal Behaviour*, 48:470–472.

Gautrais, J., Theraulaz, G., Deneubourg, J.-L., and Anderson, C. (2002). Emergent polyethism as a consequence of increased colony size in insect societies. *Journal of Theoretical Biology*, 215:367–373.

Huang, Z.-Y. and Otis, G. W. (1991a). Inspection and feeding of larvae by worker honey bees (hymenoptera: Apidae): effect of starvation and food quantity. *Journal of Insect Behavior*, 4(2):305317.

Huang, Z.-Y. and Otis, G. W. (1991b). Nonrandom visitation of brood cells by worker honey bees (hymanoptera: Apidae). *Journal of Insect Behavior*, 4(2):177–184.

Johnson, B. R. (2005). Limited flexibility in the temporal caste system of the honey bee. *Behavioral Ecology and Sociobiology*, 58:219–226.

Lindauer, M. (1952). Ein Beitrag zur Frage der Arbeitsteilung im Bienenstaat. *Zeitschrift für vergleichende Physiologie*, 34:299–345.

Rösch, G. A. (1952). Untersuchungen über die Arbeitsteilung im Bienenstaat. *Zeitschrift für vergleichende Physiologie*, 2:571–631.

Sakagami, S. F. (1953). Untersuchungen über die Arbeitsteilung in einem Zwergvolk der Honigbiene. Beiträge zur Biologie des Bienenvolkes, *Apis mellifera* L. *Japanese Journal of Zoology*, 2:117–185.

Schmickl, T. and Crailsheim, K. (2002). How honeybees (apis mellifera l.) change their broodcare behaviour in response to non-foraging conditions and poor pollen conditions. *Behavioral Ecology and Sociobiology*, 51:415–425.

Schmickl, T. and Crailsheim, K. (2008a). An individual-based model of task selection in honeybees. In et al., M. A., editor, *Proceedings of the 10th International Conference on the Simulation of Adaptive Behavior (SAB'08)*, Lecture Notes in Artificial Intelligence 5040, pages 383–392, Springer-Verlag Berlin, Heidelberg.

Schmickl, T. and Crailsheim, K. (2008b). Taskselsim: A model of the self-organisation of the division of labour in honeybees. *Mathematical and Computer Modelling of Dynamical Systems*, 14(2):101–125.

Seeley, T. D. (1982). Adaptive significance of the age polyethism schedule in honeybee colonies. *Behavioural Ecology and Sociobiology*, 11:287–293.

Seeley, T. D. (1992). The tremble dance of the honey bee: message and meanings. *Behavioral Ecology and Sociobiology*, 31:375–383.

Theraulaz, G., Bonabeau, E., and Deneubourg, J.-L. (1998). Response threshold reinforcement and division of labour in insect societies. *Proceedings of the Royal Society of London B*, 265:327–332.

Wilensky, U. (1999). *NetLogo*. *http://ccl.northwestern.edu/netlogo/*. Center for Connected Learning and Computer-Based Modeling, Northwestern University. Evanston, IL.

Altruism Amongst Spatial Predator-Prey Animats

Chris Scogings and Ken Hawick

Computer Science, Institute of Information and Mathematical Sciences,
Massey University,
Private Bag 102-904, North Shore MSC, Auckland, New Zealand
{c.scogings, k.a.hawick}@massey.ac.nz
Tel: +64 9 414 0800 Fax: +64 9 441 8181

Abstract

Understanding the emergence or suppression of altruism is an important step towards understanding real-life many-agent systems. We explore the relative survival traits of spatial animats in our predator-prey model and find some quantifiable emergent advantages of altruistic behaviour on the part of individual animats.

Keywords: altruism, animats, predator-prey.

Introduction

"Although a high standard of morality gives but a slight or no advantage to each individual man and his children over the other men of the same tribe... an advancement in the standard of morality will certainly give an immense advantage to one tribe over another."

Charles Darwin, The Descent of Man, 1871

The evolution of altruism has been debated for several decades and is still a source of some argument. On the one hand we have those who aspire to the "selfish gene" theory of Dawkins (1976) and regard altruism as (at best) negligible. On the other hand we have the approach of Sober and Sloan Wilson (1998) who state that it is possible for "pure altruism" (helping unrelated individuals and receiving no payback) to evolve. In between lie a range of possibilities including kin selection and group selection (Maynard Smith, 1964). Of particular interest is the recent work on the evolution of strong altruism in randomly formed groups (Fletcher and Zwick, 2004) and this paper also provides an excellent review of the current altruism debate. Many of the theories mentioned above use models to strengthen their position. However, most of these models are analytical and none use the "animat" (Wilson, 1991) approach as we have done here.

It appears that Darwin, quoted above in (Wilson and Wilson, 2007), believed in unadulterated group selection. This is the idea that individuals within a group will behave altruistically towards others so that the group as a whole will prosper. More on Darwin's position can be found in (Sober and Sloan Wilson, 1998). The research we present in this paper uses an animat model to explore whether this approach is viable within an Artificial Life simulation. We believe that our findings add strength to Darwin's original suggestion – namely that an altruistic group will have an advantage over a selfish group even though selfish individuals have an advantage over altruistic individuals within the group. However, this will only occur under certain conditions.

Complex systems using spatial agents or animats have existed for some time – see for example (Tyrrell and Mayhew, 1991; Adami, 1994; Holland, 1994) – and have yielded rich insights into emergent group behaviour in physical, biological and sociological simulation settings. Experimentation in this area is ongoing as illustrated in (Ronkko, 2007).

We have refined our predator prey animat model over a number of years and it has been introduced and discussed in several previous publications including (Hawick et al., 2005b; Scogings et al., 2006). Unlike other models which focus on the evolution of animats and the emergence of new species, we concentrate on making explicit, well-defined changes to the microscopic control variables of the model and then analysing any new (emergent) animat collective behaviours.

In particular we have documented fascinating emergent macro-behaviours such as the defensive spirals and other features discussed in (Hawick et al., 2004). An example of these features is provided in Figure 1. This shows some typical (and highly robust) wave front and proto-spiral pattern generation behaviours in our model. We have been able to study these in a quantifiable manner by applying automatic feature detection techniques to the spatial animat patterns (Hawick et al., 2005a).

This paper consists of the following sections: A brief overview of our predator-prey simulation; a discussion of how we introduced altruistic behaviour into the model; experimental runs of the model simulating "good times" (i.e. a grass value conducive to high prey population growth); experimental runs of the model during "bad times" (with a much lower grass value); and finally, a brief summary and conclusion.

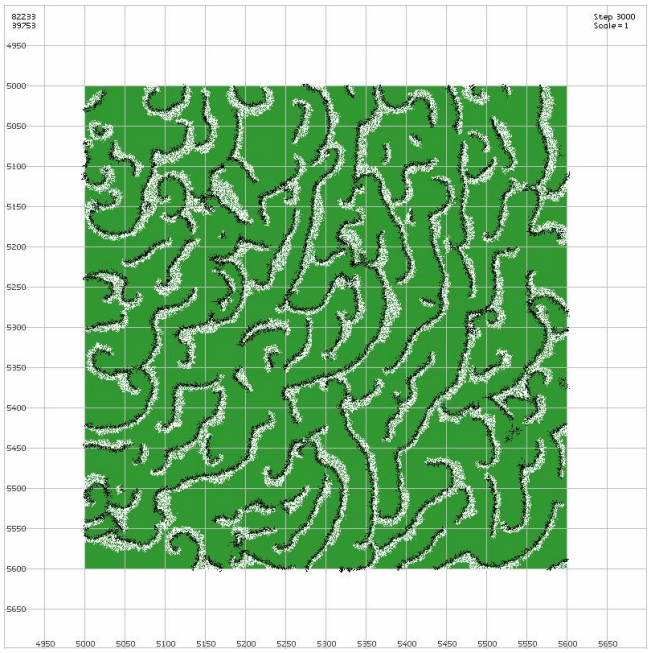


Figure 1: A typical run at step 3000. Predators are black and prey are white and all animats are selfish (the original model). The animats inhabit a square "grassy area" with a grass value of 60 which ensures healthy animat populations. Note the typical emergent clusterings, including spirals.

The Model

Our model consists of two species of interacting animats – the predators and the prey. Animats have a very simple state: a health variable; an age; and a spatial position in the 2-dimensional square mesh world. The food chain is therefore very simple as shown in Figure 2. Our system is an open one as far as energy is concerned. Prey consume "grass" which is assumed to be continually replenished although we can adjust this rate and can also adjust the spatial pattern of grass and hence the underpinning geometry of the flat world. Predators consume only prey and other things being equal we can reproduce the well known boom-bust limit cycles predicted by predator-prey models such as the Lotka-Volterra coupled differential equations (Lotka, 1925; Volterra, 1926) and their spatial variants (Gallego, 2003).

The simulation runs as a sequence of discrete and synchronous time-steps and in each time-step the following operations are performed for every animat:

- (Phase 1) health check
- (Phase 1) age check
- (Phase 1) locate neighbours
- (Phase 1) execute one rule
- (Phase 2) update all variables

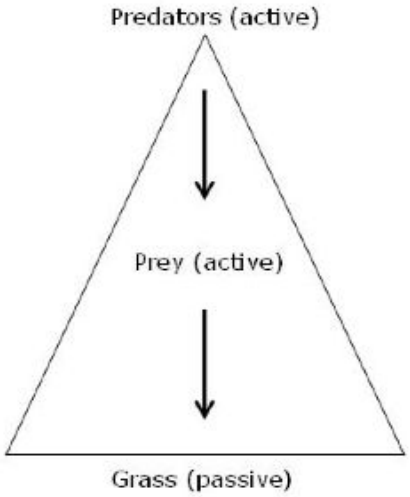


Figure 2: Model participants. Predators and Prey are spatial animat agents, whereas grass is a passive (continuously replenished) resource.

Although the model is synchronous animats are updated in a random order, which we found adequate to remove any spatial artefacts from sweep order. The process is a two-phase system in which the variables for all animats are updated after all checks have been made and all rules have been executed. The two-phase system was developed in order to ensure fairness across all the animats in the model and a full discussion of alternative updating systems is available in (James et al., 2004).

Every animat carries a small set of microscopic rules that govern its behaviour and this rule set is passed on unchanged to any offspring. It is possible to allow mutations to rules and to introduce genetic algorithms into the model but an important feature of our work is to make small, well-defined changes to the microscopic model and measure the effects of those changes. We have experimented with changing the order (priorities) of the rules and have investigated which rule sets generate the most successful animat groups (Hawick et al., 2005b). This approach is a good way of quantitatively investigating the microscopic rule space without getting lost in the combinatoric explosion that hinders a more simple minded evolutionary "suck it and see" investigation.

In this series of experiments, the rule set for predators is:

1. if well fed – breed with an adjacent predator
2. if hungry – eat an adjacent prey animat
3. if well fed – move towards another predator
4. if hungry – move towards prey
5. move randomly

and the rule set for prey is:

1. if well fed – breed with an adjacent prey animat
2. if hungry and not crowded – eat grass (if available)
3. if well fed – move towards another prey animat
4. if hungry and crowded – move away from other prey
5. move away from an adjacent predator
6. move randomly

Breeding only has a certain chance of success. This is a simple alternative to factoring in a host of complicated parameters including birth defects, nutrition, adequate shelter and so on. For these experiments the chance of a successful birth was set to 15% for predators and 40% for prey. The prey conditions involving crowding were introduced to prevent prey forming enormous clusters in any area of the grid that happened to be temporarily free of predators. If a prey animat has k or more adjacent neighbours, it is deemed to be "crowded" and can not eat grass (an abstract simulation of "over grazing"). For these experiments k was set to 10.

Rules are considered in a strict priority order. Each time-step, every animat attempts to execute the first rule in its rule set. However, most rules have conditions so can often not be executed. For example, prey will only move away from a predator if a predator is actually adjacent. If the conditions for the first rule can not be satisfied, the animat attempts to execute the next rule in the set and so on. This Markov chain mechanism of rules is described in detail in (Hawick et al., 2007).

All animats in the model have a "current health" value. This value (in some ways analogous to "internal energy") is reduced each time-step and if it reaches zero the animat "starves to death". If an animat eats something (predators eat prey and prey eat "grass") then the current health value will be increased by a certain amount, although it may never be increased past the maximum health value which is predetermined for each animat species. A "well fed" animat (see the conditions in the rules above) has a current health value of two-thirds or more of the maximum. A "hungry" animat has a current health value of less than one-third of the maximum. The concepts of health values and animats eating are discussed in (Scogings et al., 2007).

Early versions of the model did not require prey animats to eat anything and the concept of "grass" has been recently introduced. Grass can be placed in specific locations on the map and each grassy area carries a specific "grass value". When a prey animat eats the grass, its current health is increased by the grass value. This means that animats will do well on grass with a higher value and will struggle to survive on grass with a lower grass value. Grass therefore has a useful side effect in limiting the animat populations to the grassy area and preventing them becoming unmanageably large. This is important as our model boundaries are effectively open ones with no periodicity or hard reflecting walls. The wilderness is in fact a good way to limit the population without introducing spurious spatial artifacts due to boundaries. Animats can (and do) diffuse out into the wilderness and quietly starve which is less perturbative to the model's remaining and ongoing population than if they were corralled or transported spatially across a periodic boundary.

Figure 3 illustrates the effects of a low grass value on animat populations and should be compared with Figure 1 which has a high grass value. Note that we try to keep all microscopic parameters very simple and usually express them where possible as percentages. It is interesting to note that despite the low populations, clustering behaviour and spiral formation continue to emerge. Note that Figure 1 provides a snapshot at step 3000 whereas Figure 3 only shows step 1000 – in fact, with the low grass value, all animats were extinct by step 3000. The most interesting aspect of the model has been the emergence of macro-behaviours in the form of regular patterns of clusters comprising both species of animat. These clusters are persistent and recognisable across a range of conditions and control variables and are discussed in (Hawick et al., 2006).

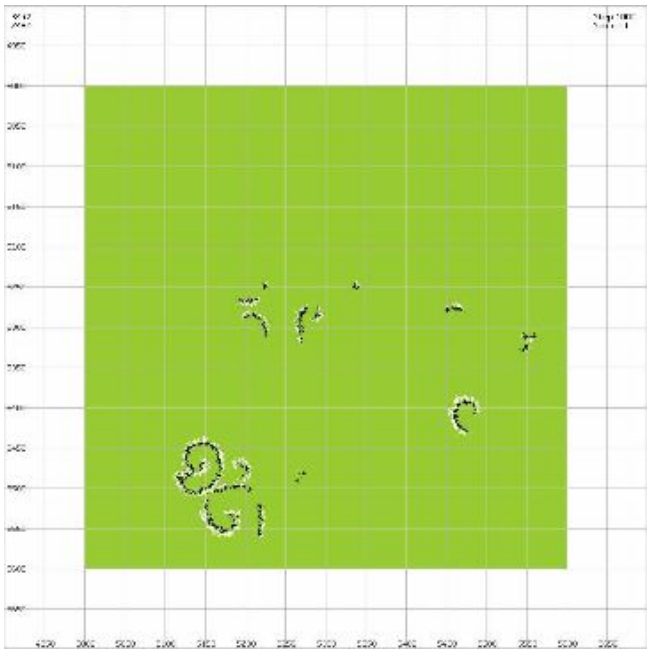


Figure 3: A run at step 1000 with a (uniform) grass value of 30. Predators are black and prey are white. This "low" grass value means low populations of both prey and predators, although emergent clustering behaviour (especially spiral formation) is still apparent. This situation should be compared with Figure 1 in which the grass value is doubled to 60.

Introducing Altruism into the Model

How to define altruistic behaviour is a central part of the general debate on altruism and a good review of this can be found in (Fletcher and Zwick, 2004). Most models use a variant of Hamilton's equation (Hamilton, 1964) but our model is not analytical so requires a form of altruism represented by a behavioural rule rather than an analytical equation. We decided that this should take the form of sharing food resources as these are most vital to animats' ongoing existence. (In fact, the sharing of food resources is probably a reasonable way to measure altruism among humans in the world today.) We also decided to only study altruistic predators as the "higher life form" in the model. As prey always have equal access to "grass" we felt that there was not much to be gained in studying the behaviour of altruistic prey. Therefore, in order to introduce altruism into the model we decided to regard predators' current health value as "transferable currency". Some predators are marked as altruistic whereas others are marked as selfish – i.e. they remain unchanged from the original model.

Just before the conclusion of phase 1 of each time-step an altruistic predator checks to see if it has an adjacent predator that has a lower current health value than its own. If this is the case, the current health values of the two predators are added together and each animat receives one half of this total value. In other words, an altruistic animat shares its current health with a neighbour (regardless of whether the neighbour is altruistic or selfish) such that the two animats receive equal shares and the current health of the altruistic animat will be reduced while the current health of the neighbouring animat will increase. Note that the altruistic animat will only share with an adjacent neighbour that is worse off than itself. No attempt is made to search for other neighbours to share with, and if several animats are both adjacent and worse off, one is chosen at random to share with.

When animats breed and produce offspring the new animats are precise clones of their "mothers". Two adjacent animats are required to breed but only one actually executes the rule to breed – this animat is known as the mother. This aspect of inheritance has existed in the model for some time and has been discussed in (Hawick et al., 2005b). Altruistic animats carry a "gene" that marks them as altruistic. Thus when altruistic animats breed they produce new altruistic animats (clones of themselves). This means that it is possible to track the success of the altruistic population versus the selfish population over time. We can then consider the question: is an altruistic society more successful than a selfish society?

Experiments 1, 2 and 3 : Good Times

In this section we describe the population variations that result from introducing altruistic predators into the system. Results are shown as animat populations plotted over time. The populations show the well known "boom-bust" phase variation that is typically found in any predator prey system. A boom in prey is followed (with a suitable phase lag) by a boom in predators. The cycle continues when predators "boom" and cause a subsequent "bust" in the numbers of prey. Spatial variations complicate this situation considerably from the simple limit cycles encountered in dimensionless Lotka-Volterra-like systems, and our model generally displays a equilibrating epoch (usually less than 1000 time steps) during which the overall population can grow or shrink drastically. We typically find however, for a remarkably wide range of microscopic animat parameter values, that a long term stable epoch subsequently arises during which the variations are due to emergence and interaction or spatial patterns such as clumps, wave fronts and spirals. It seems valid therefore to draw robust conclusions about altruism in the long term.

The first three experiments simulated a square grassed area with a grass value of 60 (which is regarded as above average). Thus the prey had no problem finding food and consequently predators always had easy access to prey. In experiment 1 there were no altruistic animats. Thus this simulation matched previous normal conditions and could be used as the control. The situation at step 3000 is depicted in Figure 1 and shows typical densities of predators and prey and the usual emergent clustering behaviour. The animat populations for this experiment are shown in the graph in Figure 4 as the second line (prey) and the fourth line (predators).

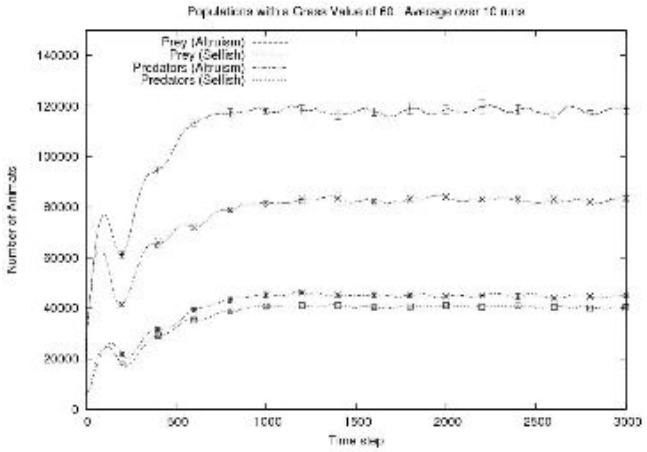


Figure 4: Plot showing total populations during experiments 1 and 2 when the grass value is 60. The populations from experiment 1 (all selfish) appear as the second line (prey) and the fourth line (predators). The populations from experiment 2 (altruistic predators) appear as the first line (prey) and the third line (predators). It is clear that the altruistic predators have succeeded in increasing their own population but more importantly are also "conserving" their prey food-source.

In experiment 2 all predators were designated as altruistic. This caused a slight increase in the total population of predators but an even greater increase in the population of prey, as shown by the first line (prey) and the third line (predators) in the graph in Figure 4. The situation at step 3000 is depicted in Figure 5 and shows much denser formations of prey animats because the altruistic predators are conserving their food source. It is interesting to note that the general pattern of emerging clusters, including spirals, has not been affected by the altruistic nature of the predators.

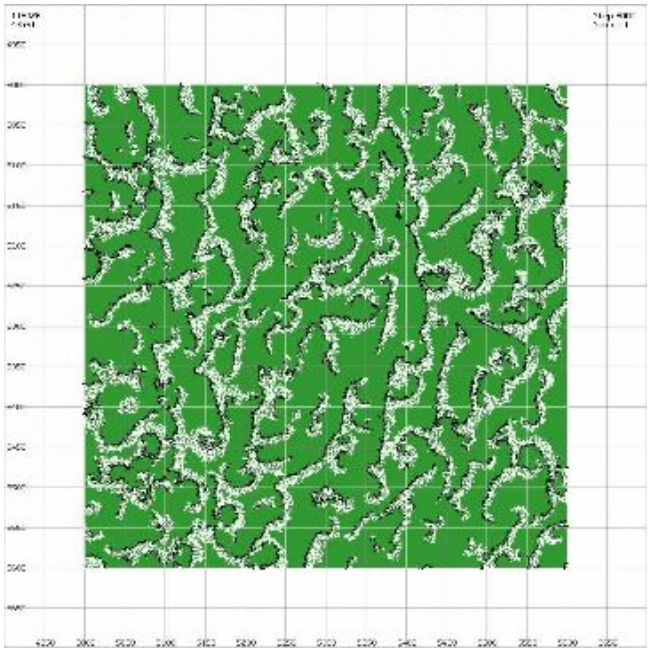


Figure 5: The situation at step 3000 during a typical run of experiment 2 with a grass value of 60. All predators (black) are altruistic and are conserving the prey population (white) which leads to formations of prey animats that are noticeably more dense than those in Figure 1 which depicts the situation for selfish predators at the same step and with the same grass value.

Remember that only predators are altruistic and there are no altruistic prey animats. Thus altruistic predators achieve two things: firstly they rescue some of their own number who may otherwise have starved to death; and secondly they also "conserve prey". This happens because predators only eat when hungry. Thus if a predator receives some health points from an altruistic neighbour, it has no need to eat prey itself. So it appears that an "altruistic society" not only ensures that its own population will be greater than the selfish equivalent, but it also conserves food reserves for future generations.

Alas, while an altruistic society may be more successful than a selfish one, this is not obvious to the individuals within that society. Experiment 3 commenced with 50% of the predators randomly designated as altruistic while the remaining 50% were designated as selfish. Thus two competing groups of predators were created. With the passage of time, it became clear that the altruistic predators could not compete with the selfish predators and the altruistic population died out completely. This is shown in Figure 6. Thus, although an altruistic society may be more successful as a group, altruistic individuals can not compete successfully with selfish individuals. However, perhaps this is only the case when times are good and food is plentiful.

All three experiments were run ten times with different random number seeds and the population graphs represent the averages from the ten simulations.

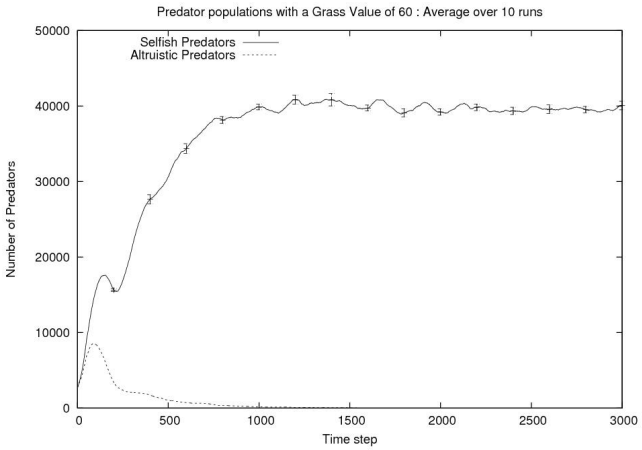


Figure 6: Graph showing predator population when the grass value is 60. The run starts with approximately equal numbers of altruistic and selfish predators but the selfish group soon comes to dominate the model.

Experiments 4, 5 and 6 : Bad Times

The second set of experiments used exactly the same procedure as the first but this time was conducted with a grass value of 30 which is below average and means that both prey and predators struggle to survive. This was made apparent in experiment 4 (the control) in which populations of both prey and predators are markedly decreased as shown in Figure 3 and in fact most animats had starved to death by step 3000. No animats were altruistic in this run. The populations of experiment 4 appear as the third line (prey) and the fourth line (predators) in Figure 7.

In experiment 5 all predators were once again designated as altruistic. The results of this run were significant and are shown in Figure 7 where the first line represents the prey population and the second line is the altruistic predator population. The altruistic animats once again were able to "conserve" the prey resource and thus were able to maintain a stable population of both predators and prey, whereas the

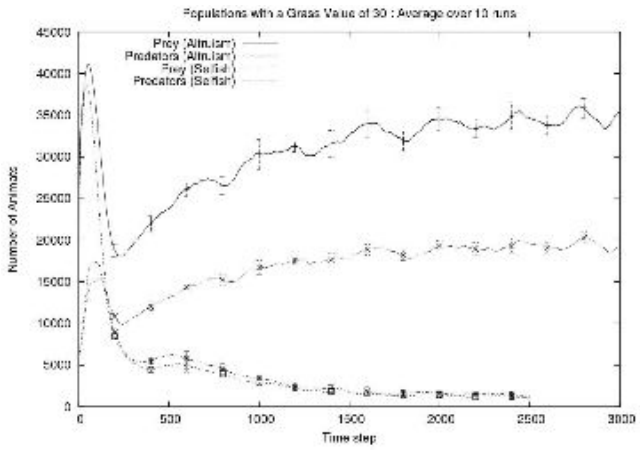


Figure 7: Graph showing total animat populations during experiments 4 and 5 with a grass value of 30. The lower two lines represent the prey and selfish predator populations in experiment 4. These lines are unreliable after step 2500 as the populations died out soon after that point in most of the runs. The top two lines represent prey and altruistic predator populations in experiment 5. This graph dramatically illustrates the success of altruistic predators at sustaining both their own population and that of their prey.

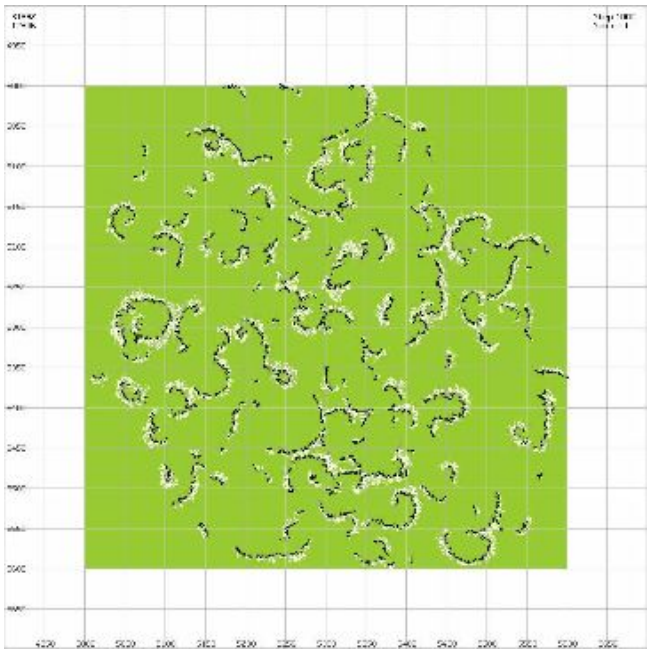


Figure 8: Experiment 5 at step 1000 with a grass value of 30. Predators are black and prey are white. The predators in this run are all altruistic and are doing remarkably better than the selfish predators (i.e. the original model) depicted in Figure 3 also at step 1000 with the same grass value.

behaviour of the selfish predators in experiment 4 caused both prey and predator populations to crash. Thus in times of adversity when food resources are scarce, the tendency of the altruistic society to conserve the prey population allows it to survive indefinitely and gives it an enormous advantage over the selfish society. The success of the altruistic society can be seen by comparing Figure 3 with Figure 8 (both at step 1000 with a grass value of 30) which clearly shows the advantages of conserving the food-source.

However we are still faced with the problem that individuals within a society appear to tend towards selfishness. We therefore performed experiment 6 with 50% of the predators randomly designated as altruistic while the remaining 50% were designated selfish. This time, because of the low grass value and the scarcity of prey, the altruistic group prospered and came to dominate the model. The population graph for experiment 6 is shown in Figure 9.

Once again, all experiments were run ten times with different random number seeds and the population graphs represent the averages from the ten simulations.

Summary and Conclusions

We have described our spatial predator-prey model in terms of its microscopic parameters and some of the emergent spatial patterns it generates. We have presented some experimental results based on averaged population trends over time, linked to some typical configuration snapshots. These suggest convergent and consistent results even over widely differing microscopic start conditions. Our model allows us to study the macroscopic properties of interacting "spatial herds" of individual animats. We are therefore able to study collective behaviour amongst the differing types of group or herd. We have tried in particular to explore the effects of altruism introduced at the microscopic level but manifesting itself as macroscopic success/failure levels.

We argue that this series of experiments supports quantitatively Darwin's assertion that an altruistic society will always do better than a selfish one even though selfish individuals within the society will tend to dominate. The experiments indicate that the main reason for this might be that altruistic societies conserve their (food) resources. This behaviour is indicative of true group selection and not kin selection as altruistic predators have no way of recognising kin and secondly, they share resources with any other predator including selfish ones.

The experiments also suggest that altruistic societies are particularly successful in times of hardship when they can continue to survive while selfish communities may vanish completely. In addition, during these times of adversity the selfish individuals within the altruistic society will not succeed and dominance will be retained by altruistic individuals. There are a number of shared resources issues related to "The Tragedy of the Commons" (Hardin, 1968). Our an-

imats do not yet have a direct sense of altruism that governs their commons, but we hope to incorporate this into a future version of the predator animats.

The use of an Artificial Life model in a study of an abstract notion such as altruism highlights the benefits that research in this area can deliver. The model was not originally developed to analyse concepts such as altruism but we have demonstrated that it can easily be adapted to this, and (we expect) to other similar uses.

The concept of the altruistic society is itself an emergent macro-behaviour as individual altruistic animats do not attempt to assist their entire society but merely one adjacent neighbour and even then, only when that neighbour has a lower health value than themselves. Thus is the concept of a complex altruistic society based on very simple rules executed by the individuals within that society.

We are presently considering how prey animats could also exhibit a form of altruism - for example by individuals sacrificing themselves to prey to save their fellows. We suspect this effect may give rise to an interesting interplay and further rich spatial behaviours.

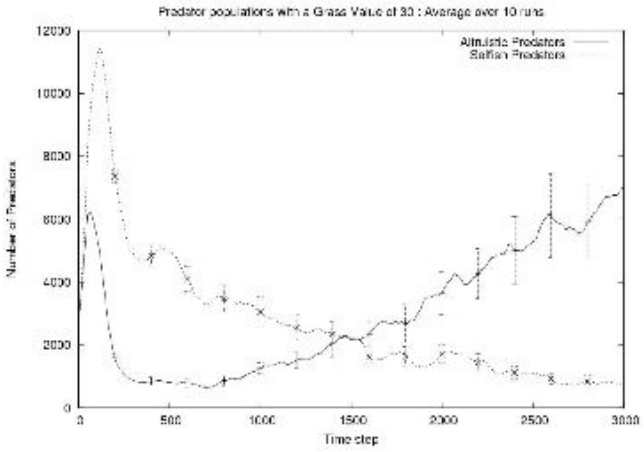


Figure 9: Graph for experiment 6 showing predator population when the grass value is 30. The run starts with approximately equal numbers of altruistic and selfish predators but eventually the altruistic group becomes dominant. The standard deviations (shown as error bars at regular intervals) are larger than usual for altruistic predators because the slope of the line after recovery (at about step 700) varied with each run. This does not change the general trends when comparing the two populations. This graph should be compared with that in Figure 6 where the selfish predators dominated because the grass value of 60 was significantly higher.

References

Adami, C. (1994). On modeling life. In Brooks, R. and Maes, P., editors, *Proc. Artificial Life IV*, pages 269–274. MIT Press.

Dawkins, R. (1976). *The Selfish Gene*. Oxford Univ. Press. ISBN 019857519X.

Fletcher, J. A. and Zwick, M. (2004). Strong altruism can evolve in randomly formed groups. *Journal of Theoretical Biology*, 228:303–313.

Gallego, S. (2003). *Modelling Population Dynamics of Elephants*. PhD thesis, Life and Environmental Sciences, University of Natal, Durban, South Africa.

Hamilton, W. D. (1964). The evolution of social behaviour. *Journal of Theoretical Biology*, 1:295–311.

Hardin, G. (1968). The tragedy of the commons. *Science*, 162(3859):1243–1248.

Hawick, K. A., James, H. A., and Scogings, C. J. (2005a). Manual and semi-automated classification in a microscopic artificial life model. In *Proc. Int. Conf. on Computational Intelligence (CI'05)*, pages 135–140, Calgary, Canada.

Hawick, K. A., James, H. A., and Scogings, C. J. (2005b). Roles of rule-priority evolution in animat models. In *Proc. Second Australian Conference on Artificial Life (ACAL 2005)*, pages 99–116, Sydney, Australia.

Hawick, K. A., James, H. A., and Scogings, C. J. (2006). A zoology of emergent patterns in a predator-prey simulation model. In Nyongesa, H., editor, *Proceedings of the Sixth IASTED International Conference on Modelling, Simulation, and Optimization*, pages 84–89, Gabarone, Botswana.

Hawick, K. A., Scogings, C. J., and James, H. A. (2004). Defensive spiral emergence in a predator-prey model. In Stonier, R., Han, Q., and Li, W., editors, *Proc. Complexity 2004*, pages 662–674, Cairns, Australia.

Hawick, K. A., Scogings, C. J., and James, H. A. (2007). Stochastic animats in complex spatial environments. Technical Report CSTN-040, Computer Science, Massey University.

Holland, J. H. (1994). Echoing emergence: Objectives, rough definitions, and speculations for echo-class models. In Cowan, G. A., Pines, D., and Meltzer, D., editors, *Complexity: Metaphors, Models and Reality*, pages 309–342. Addison-Wesley, Reading, MA.

James, H. A., Scogings, C. J., and Hawick, K. A. (2004). Parallel synchronization issues in simulating artifical life. In Gonzalez, T., editor, *Proc. 16th IASTED Int. Conf. on Parallel and Distributed Computing and Systems (PDCS)*, pages 815–820, Cambidge, MA, USA.

Lotka, A. J. (1925). *Elements of Physical Biology*. Williams & Williams, Baltimore.

Maynard Smith, J. (1964). Group selection and kin selection. *Nature*, 201:1145–1147.

Ronkko, M. (2007). An artificial ecosystem: Emergent dynamics and lifelike properties. *J. ALife*, 13(2):159–187.

Scogings, C. J., Hawick, K. A., and James, H. A. (2006). Tools and techniques for optimisation of microscopic artificial life simulation models. In Nyongesa, H., editor, *Proceedings of the Sixth IASTED International Conference on Modelling, Simulation, and Optimization*, pages 90–95, Gabarone, Botswana.

Scogings, C. J., Hawick, K. A., and James, H. A. (2007). Tuning growth stability in an animat agent model. In *Proceedings of the 16th IASTED International Conference in Applied Simulation and Modelling (ASM 2007)*, pages 312–317.

Sober, E. and Sloan Wilson, D. (1998). *Unto Others: The Evolution and Psychology of Unselfish Behaviour*. Harvard University Press.

Tyrrell, T. and Mayhew, J. E. W. (1991). Computer simulation of an animal environment. In Meyer, J.-A. and Wilson, S. W., editors, *From Animals to Animats, Proceedings of the First International Conference on Simulation of Adaptive Behavior*, pages 263–272.

Volterra, V. (1926). Variazioni e fluttuazioni del numero d'individui in specie animali conviventi. *Mem. R. Accad. Naz. dei Lincei, Ser VI*, 2.

Wilson, D. S. and Wilson, E. O. (2007). Evolution: Survival of the selfless. *New Scientist*, 2628:42–46.

Wilson, S. W. (1991). The animat path to AI. In Meyer, J.-A. and Wilson, S. W., editors, *From Animals to Animats 1: Proceedings of The First International Conference on Simulation of Adaptive Behavior*, pages 15–21. Cambridge, MA: The MIT Press/Bradford Books.

Measuring emergence via nonlinear Granger causality

Anil K. Seth

Department of Informatics, University of Sussex, Brighton, BN1 9QJ
a.k.seth@sussex.ac.uk, www.anilseth.com

Abstract

The concept of emergence is central to artificial life and complexity science, yet quantitative, intuitive, and easy-to-apply measures of emergence are surprisingly lacking. Here, I introduce a just such a measure, G-emergence, which operationalizes the notion that an emergent process is both *dependent upon* and *autonomous from* its underlying causal factors. G-emergence is based on a nonlinear time series analysis adapted from 'Granger causality' and it provides a measure not only of emergence but also of apparent 'downward causation'. I illustrate the measure by application to a canonical example of emergence, an agent-based simulation of bird flocking, and I discuss its potential impact on perhaps the most challenging of all scientific problems involving emergence: consciousness.

The maturation of artificial life and complexity science over recent years has given rise to renewed interest in emergence. Although the concept of emergence has a long philosophical history (Broad, 1925; Kim, 1999), its essence is simple enough: An emergent property is somehow 'more than the sum' of its component parts. Emergent properties appear rife within complex systems of all kinds: biological, cognitive, social, and technological. Broadly speaking, artificial life and complexity science focus on explaining phenomena that seem to involve emergence, and models constructed under these auspices are often described as emergent (Bedau, 2003). It is therefore surprising and significant that quantitative and easy-to-apply *measures* of emergence are mostly lacking. This is unfortunate because the ability to measure a phenomenon is an essential step towards its effective scientific description (Chang, 2004).

In this paper I will first differentiate several notions of emergence and by doing so briefly illustrate some relevant conceptual challenges. I will then introduce 'G-emergence', a new measure which operationalizes the intuition that an emergent process is simultaneously *autonomous from* and *dependent upon* its underlying causal factors. G-emergence is easy to apply, and I illustrate it by application to a canonical example of emergence: bird flocking. I end by discussing related measures, how it can defuse the metaphysically awkward notion of 'downward causation', and how it may shed new light one of the most recalcitrant problems in science: the relation between neural mechanism and phenomenal experience.



Figure 1: A flock of starlings about to roost.

Varieties of emergence

Intuitively, emergence refers either to a macro-level property that is 'more than the sum of' the micro-level parts ('property' or 'synchronic' emergence) or to the appearance of a qualitatively distinctive new phenomenon over time ('temporal' or 'diachronic' emergence). A striking example of property emergence is a flock of starlings wheeling in the sky before they roost: the flock seems to have a shape and trajectory of its own, which appears to exceed those of the individual birds (Figure 1). Temporal emergence is well illustrated by the appearance of new morphological features during embryogenesis and development. This paper focuses on measuring property emergence, though new opportunities for measuring temporal emergence are also identified.

Following Bedau (1997, 2003), both property emergence and temporal emergence can be differentiated into three categories: strong, weak, and nominal [similar decompositions can be found in (van Gulick, 2001; Bar-Yam, 2004)]. The least controversial of these is nominal emergence, which is simply the notion of a kind of property that can be possessed by macro-level objects or processes but not by their micro-level constituents. For example, a circle is nominally emer-

gent from the set of points from which it is constructed. Because nominally emergent properties can be derived trivially I will not discuss them any further.

Most challenging and controversial is the notion of strong emergence, which involves two closely related claims. First, a macro-level property is in principle not identifiable from micro-level observations. Second, macro-level properties have irreducible causal powers. The first claim rejects mechanistic explanations altogether, apparently calling a halt to scientific advance in the absence of new fundamental principles of nature (Chalmers, 2006). The second raises the difficult notion of 'downward causation'. Downward causation is problematic firstly because it contravenes the plausible doctrine that 'the macro is the way it is in virtue of the how things are at the micro', an idea that has been expressed variously as 'causal fundamentalism' (Jackson and Pettit, 1992) or 'supervenience' (Kim, 1999). A second challenge raised by downward causation is that of resolving conflicts between micro-level and macro-level causes (Bedau, 2003). Even so, the main problem with strong emergence may lie in its scientific irrelevance (Bedau, 2003). The only recurrent example of strong emergence in the scientific literature is that of the emergence of conscious states (e.g., qualia) from neurobiological processes (Sperry, 1969; Chalmers, 2006), which may speak more to our lack of understanding of consciousness than to our grasp of deep principles of emergence. I will return to this possibility later on.

In between strong emergence and nominal emergence lies the useful notion of weak emergence (Bedau, 1997, 2003), according to which a macro-level property is derived from the interaction of micro-level components but in complicated ways such that the macro-level property has no simple micro-level explanation. In contrast to strong emergence, weakly emergent properties are in principle identifiable from micro-level components, and in contrast to nominal emergence, the micro-to-macro inferential pathways must be non-trivial. According to Bedau, weakly emergent macro-level properties are *ontologically dependent* on and reducible to micro-level causal factors, but at the same time they are *epistemologically irreducible* due to the complexity of the micro-to-macro inferential pathways.

What exactly does it mean for a macro-level property to be epistemologically irreducible? Bedau's answer is that a weakly emergent (epistemologically irreducible) property is underivable from its micro-level parts *except by simulation*. This is an all-or-none classification. Either a macro-level property can be derived by some explanatory short-cut, in which case weak emergence does not apply, or it cannot, in which case the micro-level causal factors need to be simulated explicitly in order to derive the macro-level property.

In this paper I consider a continuous version of weak emergence, in which a macro property is weakly emergent *to the extent that* it is not identifiable from micro-level observations. This variation is valuable firstly because for many systems it may not be possible to prove 'underivability except by simulation', and secondly because from the perspective of measurement, a continuous value is much more useful than a binary classification.

Measuring weak emergence

To derive a continuous measure of weak emergence, I take as a starting point the idea that a weakly emergent macro-level property is simultaneously (i) *autonomous from* and (ii) *dependent upon* its underlying causal factors (Bedau, 1997). To operationalize this notion statistically, I propose that a macro-variable M can be measured as weakly emergent from a set of micro-variables $\mathbf{m}$ ($\mathbf{m} = m_1 \ldots m_N$) to the extent that: (i) past observations of M help predict future observations of M with greater accuracy than predictions based on past observations of $\mathbf{m}$ alone, and (ii) past observations of $\mathbf{m}$ help predict future observations of M with greater accuracy than predictions based on past observations of M alone.

The first condition provides an objective measure of the non-triviality of micro-to-macro inferential pathways, and the second checks for micro-to-macro causal dependence. This definition is relative to a choice of macro and micro description levels and is also relative to a choice of prediction method. As described below, an appropriate framework for prediction is provided by a statistical definition of causality first introduced by Granger (1969), in recognition of which the present measure is called *G-emergence*.

G-causality

In 1969 Granger introduced the idea of 'Granger causality' (G-causality) as a formalization, in terms of linear regression modelling, of Wiener's intuition that Y 'causes' X if knowing Y helps predict the future of X (Granger, 1969; Seth, 2007a). According to G-causality, Y causes X if the inclusion of past observations of Y reduces the prediction error of X in a linear regression model of X and Y, as compared to a model which includes only previous observations of X. Since its introduction, G-causality has found wide application in economics and many other fields including neuroscience and climatology (Ding et al., 2006; Seth, 2008a). To illustrate G-causality, suppose that the temporal dynamics of two time series, $X_1(t)$ and $X_2(t)$ (both of length T), can be described by a bivariate autoregressive model:

$$X_1(t) = \sum_{j=1}^{p} A_{11,j} X_1(t-j) + \sum_{j=1}^{p} A_{12,j} X_2(t-j) + \xi_1(t)$$
$$X_2(t) = \sum_{j=1}^{p} A_{21,j} X_1(t-j) + \sum_{j=1}^{p} A_{22,j} X_2(t-j) + \xi_2(t)$$

where p is the maximum number of lagged observations included in the model (the model order, $p < T$), A contains

the coefficients of the model, and ξ_1, ξ_2 are the residuals (prediction errors) for each time series. If the variance of ξ_1 (or ξ_2) is reduced by the inclusion of the X_2 (or X_1) terms in the first (or second) equation, then it is said that X_2 (or X_1) *G-causes* X_1 (or X_2). Assuming that X_1 and X_2 are covariance stationarity (i.e., unchanging mean and variance), the magnitude of this interaction can be measured by the log ratio of the prediction error variances for the restricted (R) and unrestricted (U) models:

$$gc_{2\rightarrow 1} = log\frac{var(\xi_{1R(12)})}{var(\xi_{1U})},$$

where $\xi_{1R(12)}$ is derived from the model omitting the $A_{12,j}$ (for all j) coefficients in the first equation and ξ_{1U} is derived from the full model. Importantly, G-causality is easy to generalize to the multivariate case in which the G-causality of X_1 is tested in the context of multiple variables $X_2 \ldots X_N$ ($X_i \neq X_j$ for all $X_{i,j}$). In this case, X_2 G-causes X_1 if knowing X_2 reduces the variance in X_1's prediction error when the activities of all other variables $X_3 \ldots X_n$ are also included in the regression model (see below). For a tutorial introduction to Granger causality, see Seth (2007a).

G-autonomy

A simple extension of G-causality allows quantification of the 'statistical autonomy' of a variable with respect to a set of other variables (Seth, 2007b). In this case, instead of asking whether the prediction error of X_1 is reduced by including past observations of X_2, we ask whether the prediction of error of X_1 is reduced by inclusion of its *own* past, given a set of external variables. That is, a variable X_1 is *G-autonomous* to the extent that its own past states help predict its future states over and above predictions based on past states of a set of external variables $X_2 \ldots X_N$. By analogy with G-causality, the G-autonomy of X_1 with respect to X_2 is given by:

$$ga_{X_1|X_2} = log\frac{var(\xi_{1R(11)})}{var(\xi_{1U})},$$

where $\xi_{1R(11)}$ is derived from the model omitting the $A_{11,j}$ (for all j) coefficients in the Granger equations.

G-autonomy amplifies the notion of autonomy as 'self-determination' in contrast to other more abstract notions such as 'organizational closure' (Varela, 1979). It is consistent with the notion that a (behaviorally) autonomous system should not be fully determined by its environment, and that a random system should not have high autonomy (Bertschinger et al., 2008). Put simply, a variable is G-autonomous to the extent that it is dependent on its own history and that these dependencies are not accounted for by external factors. Previously I have shown that G-autonomy behaves as expected for simple model systems and can increase as a result of evolutionary adaptation (Seth, 2007b).

G-emergence

Having defined G-causality and G-autonomy the extension to G-emergence is straightforward. A macro-variable M is *G-emergent* from a set of micro-variables $\mathbf{m}$ if and only if (i) M is G-autonomous with respect to $\mathbf{m}$ and (ii) M is G-caused by $\mathbf{m}$. A simple measure for the G-emergence of M from $\mathbf{m}$ is therefore given by

$$ge_{M|\mathbf{m}} = ga_{M|\mathbf{m}}\left(\frac{1}{N}\sum_{i=1}^{N} gc_{m_i\rightarrow M}\right)$$

This measure captures three basic intuitions about weak emergence (Bedau, 1997): that it is a subset of nominal emergence, that it involves dependence on underlying processes, and that it involves autonomy from underlying processes. Importantly, $ge_{M|\mathbf{m}}$ will be zero either if M is independent of $\mathbf{m}$ or if M is fully predicted by $\mathbf{m}$.

Under what circumstances could G-emergence be high? A macro variable could be G-emergent from a set of micro variables if there are 'hidden' or 'latent' influences, i.e., relevant micro causal factors not represented in the regression. However, even if all micro causal factors are present G-emergence can still arise because of dependence on the prediction algorithm used. It is plausible, and indeed necessary for G-emergence to be useful in practice, that in some cases the macro variable is more epistemically transparent to the prediction algorithm than is the collection of micro variables. This, too, is consistent with Bedau's weak emergence, where 'underivability except by simulation' is replaced by '(un)predictability by Granger causality'.

An obvious criticism of G-emergence as measured using linear modeling is that a macro-variable may appear to be G-emergent in virtue of being a nonlinear function of its micro-level components. Clearly, a satisfying measure of emergence should not rely on the failure of linear methods to detect nonlinear dependencies. Fortunately, it is easy to extend G-causality (and hence G-autonomy and G-emergence) to nonlinear situations, for instance by a Taylor expansion:

$$\begin{aligned} X_1(t) = &\sum_{k=1}^{q}\sum_{j=1}^{p} A_{11,j,k}{X_1}^k(t-j) + \\ &\sum_{k=1}^{q}\sum_{j=1}^{p} A_{12,j,k}{X_2}^k(t-j) + \\ &\sum_{k=1}^{q}\sum_{j=1}^{p} A_{13,j,k}{X_3}^k(t-j) + \xi_1(t) \qquad (1) \\ X_2(t) = &\sum_{k=1}^{q}\sum_{j=1}^{p} A_{21,j,k}{X_1}^k(t-j) + \\ &\sum_{k=1}^{q}\sum_{j=1}^{p} A_{22,j,k}{X_2}^k(t-j) + \end{aligned}$$

$$\sum_{k=1}^{q}\sum_{j=1}^{p} A_{23,j,k} {X_3}^k(t-j) + \xi_2(t) \quad (2)$$

$$X_3(t) = \sum_{k=1}^{q}\sum_{j=1}^{p} A_{31,j,k} {X_1}^k(t-j) + \sum_{k=1}^{q}\sum_{j=1}^{p} A_{32,j,k} {X_2}^k(t-j) + \sum_{k=1}^{q}\sum_{j=1}^{p} A_{33,j,k} {X_3}^k(t-j) + \xi_3(t) \quad (3)$$

where q is the number of polynomial terms to be included in the Taylor expansion; note the set of variables has been expanded to three in order to illustrate extension to the multivariate ($n > 2$) case. In this example:

$$ge_{X_1|X_2,X_3} = \log \frac{var(\xi_{1R(11)})}{var(\xi_{1U})} \times \frac{1}{2}\left(\log \frac{var(\xi_{1R(12)})}{var(\xi_{1U})} + \log \frac{var(\xi_{1R(13)})}{var(\xi_{1U})}\right) \quad (4)$$

where, following the previous convention, $\xi_{1R(ab)}$ is derived from the model omitting the A_{ab} coefficients in (3). A value of linear or nonlinear G-emergence can be considered statistically significant if the corresponding G-autonomy and G-causality measures are themselves statistically significant. This can be assessed by F-tests on the null hypothesis that the coefficients in A_{11} (G-autonomy) and $A_{12} \ldots A_N$ (G-causality) are zero (Granger, 1969; Geweke, 1982).

It is worth noting that the concept of G-emergence does not depend on using a particular method for nonlinear regression. There exist other more sophisticated methods than Taylor expansions which can be less sensitive to noisy observations and which involve fewer parameters. For example, Ancona et al. (2004) have shown that radial basis functions can serve as effective regression kernels for measuring nonlinear Granger causality. However, for present purposes the Taylor method is preferable because (i) it is simple to describe and to implement, (ii) statistical significance can easily be assessed, and (iii) it supplies an explicit formula for G-emergence (4). Finally, note that the value of G-emergence will depend on the set of micro-variables included in $\mathbf{m}$. Therefore, in heterogenous systems it will be possible to identify a *G-emergence set* as that set of micro-variables which maximizes $ge_{M|\mathbf{m}}$.

Example: Flocking

I now show that G-emergence behaves appropriately in a simple computational model of property emergence. As noted above, a canonical example of property emergence is flocking behavior among birds. In a seminal work in artificial life, Reynolds (1987) showed that visually compelling bird flocking can be simulated by combining three simple rules for simulated birds (boids):

- *aggregation*, each boid tends to fly towards the perceived centre-of-mass (CM) of the flock,
- *avoidance*, each boid tends to avoid colliding with other nearby boids,
- *matching*, each boid tends to align its velocity with that of other nearby boids.

Here, a simple boids simulation is used to test whether visually compelling flocking correlates with high G-emergence of the CM of the flock (the macro-variable) with respect to the trajectories of the individual boids (the micro-variables).

$N = 10$ boids were simulated in a toroidal square environment of length 200 (all dimensions and distances are in arbitrary units; speeds are given in units per time-step). Boids were initialized with positions and velocities randomly chosen from the range [0,200] (x, y position), $[0, 2\pi]$ (heading), and [3,9] (speed). At each time-step the heading α_i and speed s_i of each boid i were updated synchronously according to:

$$\alpha_i = \alpha_i + a_1\theta_1 + a_2(\pi + \theta_2) + a_3\theta_3 + r_1,$$
$$s_i = s_i + a_4\overline{ds} + r_2,$$

where θ_1 is the bearing to the perceived CM (i.e., the CM not including boid i), θ_2 is the bearing to the nearest boid, θ_3 is the bearing to the mean heading of all other boids within a 20 unit range, $\overline{ds}$ is the difference between the speed of boid i and the mean speed of all other boids within 20 units, and r_1 and r_2 are random numbers in the range [-0.01,0.01]. The parameter vector a (all $a \in [0, 1]$) determines the relative contribution of each factor. Toroidal distances were calculated in the standard way, according to the minimum distance either across, or not across, the boundary. CM positions were calculated iteratively in order to minimize the toroidal distance to each boid (i.e., not as the average boid position, which leads to boundary artifacts).

Three different conditions were tested. Condition R (random) produced near-random boid behavior ($a_R = [0.01, 0.01, 0.01, 0.01]$). Condition L (low) evoked poor flocking behavior by imposing a strong dependence on velocity matching; boids in this condition tended to move in semi-rigid formations ($a_L = [0.1, 0.1, 0.6, 0.6]$). Condition H (high) evoked compelling flocking behavior; the parameter set ($a_H = [0.1, 0.3, 0.3, 0.3]$) was selected by hand. Examples of boid and CM trajectories from each condition are shown in Figure 2. Although static images do not fully capture the dynamic nature of flocking it is clear that boid trajectories in condition H are more flock-like than those in conditions L and R.

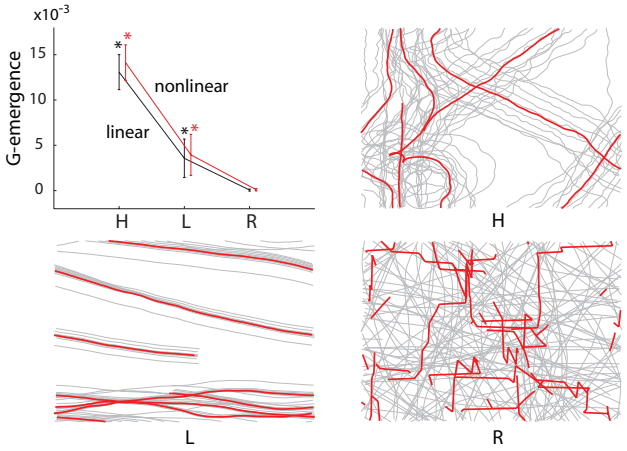


Figure 2: G-emergence of the centre-of-mass (CM) of a boid flock. Top left: Mean and standard deviation linear and nonlinear G-emergence by condition (asterisks show statistical significance). Other panels: Example trajectories (500 time-step segment) of the boids (grey) and CM (red) in condition H (high G-emergence), L (low G-emergence), and R (random).

For each condition the boid simulation was run 25 times with each run lasting 5000 time-steps; for each run the x, y coordinates of each boid and the global CM were recorded. Several preprocessing steps were carried out prior to calculation of G-emergence. In order to reduce the dimensionality of the dataset and to provide further robustness against boundary effects, each x, y coordinate pair was transformed into a single variable reflecting distance from the centre of the environment. The first 500 data points were removed to eliminate initial transients and each resulting time series was transformed into its zero-mean equivalent. Finally, each time series was first-order differenced in order to ensure covariance stationarity (Seth, 2005). Following preprocessing, for each run in each condition both linear and nonlinear G-emergence of the CM were computed using ordinary least squares regression. I chose a model order $p = 5$ and (for the nonlinear analysis) a polynomial order $q = 3$. The model order was selected based on the average Akaike information criterion (Seth, 2007a) across all 75 runs.

Figure 2 shows the mean linear and nonlinear G-emergence of the CM in each condition. Confirming the prediction that high G-emergence tracks compelling flocking, both linear and nonlinear measures show significantly higher values of G-emergence in condition H than in conditions L and R. All values of G-emergence in conditions in H and L were significant ($P < 10^{-5}$ for G-autonomy and G-causality, two-tailed t-test); those in condition R were not.

To test the behavior of G-emergence across different parameter combinations in the boids model, I computed linear and nonlinear G-emergence for each parameter vector in the

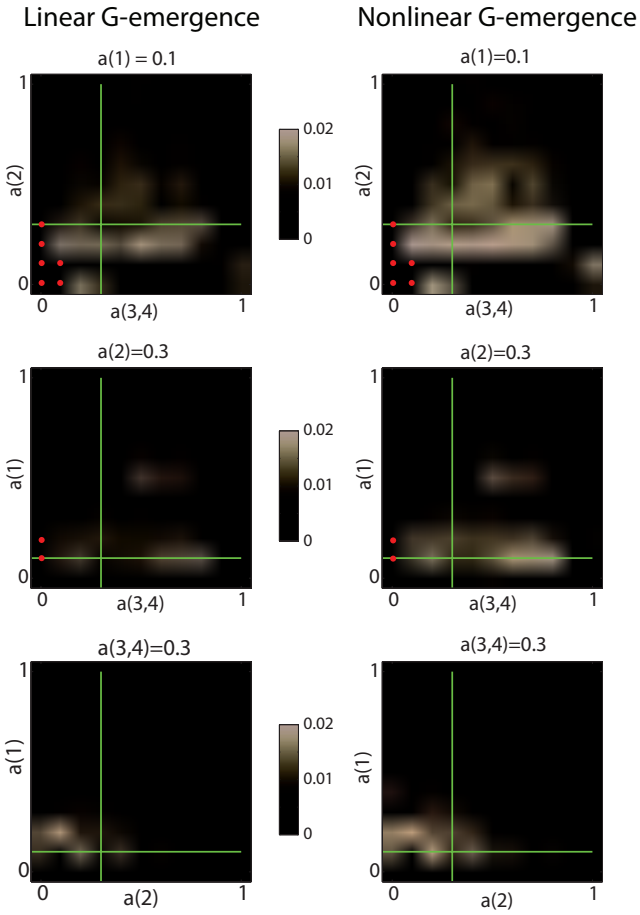


Figure 3: Parameter space of the boids model. The parameter vector a_H is indicated by the intersection of the green lines. Grey scale shows average linear and nonlinear G-emergence of the global CM. Each value is the average of three evaluations of 5000 time-steps each. Red dots indicate parameter combinations which lead to reliably non-stationary time-series.

space $a_{(1,2,3)} \in [0.0, 0.1, \ldots, 1.0]$. Parameters a_3 and a_4 were yoked together because they both influence the same rule (velocity matching) and three evaluations were carried out for each vector, requiring a total of $11 \times 11 \times 11 \times 3 = 3993$ evaluations. Figure 3 shows G-emergence for three orthogonal cross-sections through the three-dimensional parameter space; in each cross-section the vector corresponding to a_H (condition H) is marked by the intersection of the green lines.

Several aspects of the above cross-sections are notable. First, linear and nonlinear G-emergence are strongly correlated suggesting that even linear measures can provide insight into emergent properties in some complex systems. Second, in most regions of parameter space G-emergence changes smoothly, suggesting it is a robust measure. However, some regions show sharp transitions, for example be-

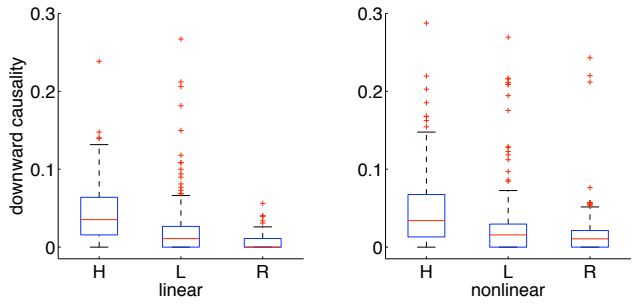


Figure 4: Downward causation is significantly higher in condition H than in conditions L or R. Boxplots show linear and nonlinear Granger causality from the global CM to individual boids, calculated separately for each boid for all 25 runs in each condition (i.e., 250 values per boxplot). Non-significant causalities were set to zero (nominal threshold of 0.01, Bonferroni corrected to 10^{-5}). Resulting distributions are non-normal and differences between conditions were tested using the Wilcoxon rank sum test. For both linear and nonlinear analyses all pairwise comparisons among medians were significantly different ($P < 10^{-3}$). Each boxplot shows lower quartile, median, and upper quartile values; whiskers show range of remaining data and '+' denote outliers.

tween some vectors with $a_1 = 0$ and neighboring vectors. The sensitivity of G-emergence to these transitions indicates that it can usefully identify parameter regions of complex models in which non-trivial weak emergence is present.

Downward causation

A common intuition regarding emergence is that it involves 'downward' causation from macro-levels to micro-levels. For proponents of strong emergence, downward causation is in fact an essential aspect of what it means to be emergent (Kim, 1999). However, physical interpretations of downward causality pose tricky metaphysical problems, for example, how to resolve competing micro- and macro- causes (Bedau, 2003). G-emergence, being statistically defined, presents a metaphysically innocent alternative according to which downward causality is reflected by G-causality from the macro-variable(s) to the micro-variable(s).

Figure 4 shows downward (Granger) causation from the global CM to the individual boid trajectories, for both linear and nonlinear G-causality measures. Averages are taken across all boids and across all 25 runs in each condition. Consistent with an association between emergence and downward causation, both measures of downward causation are significantly higher in condition H than in conditions R or L. Despite this result, it seems possible in principle for weak emergence to occur without downward causation (of course strong emergence requires downward causality by definition). Having separately applicable measures of weak emergence and downward causation makes it possible to explore conditions (if any) in which emergence and downward causation do not occur together, potentially refining and deepening the concept of emergence.

Discussion

In this paper I have introduced a method for detecting the degree of weak emergence in a system by performing physical measurements on it. Because this measure it based on a statistical interpretation of causality it sidesteps conceptual pitfalls such as competition among micro- and macro-causes, and it provides an objective and graded assessment of the non-triviality of micro-to-macro inferential pathways. MATLAB (Mathworks, Natick, MA) code for calculating G-emergence from arbitrary time-series data is provided on the author's website, www.anilseth.com.

Diachronic emergence

Diachronic (temporal) emergence refers to the appearance of new properties over time, as exemplified by evolution and development. A diachronically emergent process is by definition statistically non-stationary and therefore not amenable to direct measurement by G-emergence. Nonetheless, it is plausible that a diachronically emergent process is bracketed by statistically stationary periods with different G-emergence properties. In this way, G-emergence could be used to indirectly to infer diachronic emergence.

Relation to other measures

The intuition that differences in predictive ability may be important in defining macro-level properties is shared by (Shalizi and Moore, 2006). However these authors focus on clarifying the concept of a macro-state and they do not explicitly combine measures of autonomy and causal dependence. Rather, one process is called emergent from another if it has a higher 'predictive efficiency' than the process it derives from. Their measure of predictive efficiency is based on information-theoretic model reconstruction [the epsilon-machine concept, see (Crutchfield, 1994)], which is powerful but less easy to apply in practice than the time series metrics described here. A related approach is taken by Polani (2006) in which an 'emergent description' involves a further step of decomposing systems into independent informational sub-components.

According to the 'contextual emergence' of Atmanspacher (2007), derivation of macro-level properties requires knowledge of micro-level properties and of contingent contextual conditions, the latter defined in terms of stability criteria according to a dynamical systems analysis. This concept diverges from the doctrine of causal fundamentalism (or supervenience) by proposing that micro-level properties offer necessary but not sufficient conditions for deriving macro-level properties, which is suggestive of strong emergence. An explicit measure of strong emergence is offered by Bar-Yam (2004) which is based on measuring the entropy of a system at multiple scales. Oscillations in 'multiscale variety' are suggested to reveal constraints on the values of multiple variables which are not present among subsets of these variables, and the existence of

such constraints is taken to indicate strong emergence. However, since on the present account strong emergence rejects mechanistic explanations altogether, a full analysis of Bar-Yam's measure is beyond the present scope.

Phase transitions

Physicists have recently become interested in the onset of collective behaviors among boid-like self-propelling particles (Vicsek et al., 1995; Gregoire et al., 2003). In such systems, phase transitions can be observed among 'gaseous' phases (each particle moves independently), 'liquid' phases (particles move collectively but still diffuse with respect to each other) and 'solid' phases (particles move collectively and remain fixed with respect to each other). Plausibly, these phases correspond respectively to conditions R, H, and L of the present model and the sharp boundaries noted in figure 3 may correspond to phase transitions. However, phase transition analyses tend to focus on the dynamics of transition and assume emergent behavior is phenomenologically obvious in some phases and is absent in others. In contrast, the present focus is on detecting the *degree* of emergence by making physical measurements on a system.

Strong emergence and consciousness

As already noted, strong emergence differs fundamentally from weak emergence in that (strongly) emergent properties are suggested to be causally irreducible to their micro-level components and to exert downwardly causal influences on these components (Kim, 2006). Strong emergence thus poses a radical challenge to science because it implies that there exist real properties in the world that do not 'bottom out' in known sorts of physical interactions.

David Chalmers has made explicit a recurring idea, which is that there is exactly one clear case of a strongly emergent phenomenon, and that is the phenomenon of consciousness (Chalmers, 2006). It seems that two commonly held intuitions about consciousness drive this suspicion. First, the idea that even complete knowledge of the physical interactions sustained by brains will not provide an understanding what is like to have a conscious experience: this is the infamous 'hard problem' of consciousness. Second, the intuition that conscious states have causal efficacy in the world, as exemplified by the notion of free will but which runs through all aspects of consciousness; after all, why have experiences at all if they don't do anything? These intuitions map cleanly onto the defining features of strong emergence, namely that macro-level properties in principle cannot be identified from micro-level observations, and that macro-level properties have irreducible causal powers.

These intuitions can however be challenged. First, to expect a scientific resolution to the 'hard problem' as it is presently conceived may be to misunderstand the role of science in explaining nature. A scientific theory cannot presume to replicate the experience it describes or explains; a theory of a hurricane is not a hurricane (Seth and Edelman, 2008). If the phenomenal aspect of experience is irreducible, so is the fact that physics has not explained why there is something rather than nothing, and this has not prevented physicists from laying bare many mysteries. Second, consciousness can be functionally efficacious without recourse to downward causation. It is entirely plausible that certain neural mechanisms support useful functions in virtue of the fact that they entail conscious experiences (Seth, 2008b). For example, the neural mechanisms underlying consciousness may serve to integrate large amounts of information over short time periods, leading to functionally effective high-dimensional discriminations among a large repertoire of sensorimotor scenes (Tononi and Edelman, 1998). Such information integration may entail conscious qualia in just the same way that the molecular structure of hemoglobin entails a particular spectroscopic response: it simply could not be otherwise (Edelman, 2003). Moreover, experiences of 'free will' and 'volition' are just experiences like any other, and there is a wealth of experimental evidence showing, unsurprisingly, that awareness of a voluntary action is preceded by recognizable signatures in neural activity (Libet, 1985). Together, these points suggest that the association of consciousness with strong emergence does not rest on solid ground.

In contrast, it is very likely that the connection between neural mechanism and conscious experience involves *weak* emergence in many ways. A striking feature of conscious experience is that it seems more than the sum of its parts (each conscious experience is a unity) and that it has a vivid temporality (William James' 'stream of consciousness'). Models of consciousness that can be analyzed in terms of weak emergence therefore have the potential to *explain* features of phenomenology in terms of dynamical processes at the level of neural mechanism. The development and experimental testing of such 'explanatory correlates' (Seth and Edelman, 2008) is a highly promising avenue towards a scientific description of consciousness. It is exciting to consider that measures of weak emergence may eventually find utility in accounting for apparent free will and in crossing the explanatory gap between neural mechanism and phenomenal experience.

Conclusions

Scientific progress in understanding a phenomenon relies on the ability to measure that phenomenon. 'Emergence' has thus far resisted the development of useful measures, perhaps because of a suspicion that it necessarily involves violation of mechanistic/reductionistic explanations. But this suspicion is only valid for 'strong' emergence and proposed measures of strong emergence are correspondingly difficult to apply and interpret (Bar-Yam, 2004). In this paper I have developed and illustrated a quantitative, intuitive, and practically straightforward measure of weak emergence. G-

emergence is based on the intuition that emergent properties are both *dependent on* and *autonomous from* their components (Bedau, 1997) and is operationalized using linear and nonlinear time series analysis.

In a simulation of bird flocking, visually compelling flocking behavior is accompanied by high G-emergence as compared to random movement or flight in rigid formations. High G-emergence is also accompanied by downward (Granger) causation from the flock to each boid, though this may not be the case for all systems. Finally, G-emergence provides a platform for measuring other sorts of emergence; for instance 'temporal emergence' and/or 'self-organization' could be measured as the change in G-emergence between two different time periods.

References

Ancona, N., Marinazzo, D., and Stramaglia, S. (2004). Radial basis function approaches to nonlinear granger causality of time series. *Physical Review E*, 70:056221.

Atmanspacher, H. (2007). Contextual emergence from physics to cognitive neuroscience. *Journal of consciousness studies*, 14:18–36.

Bar-Yam, Y. (2004). A mathematical theory of strong emergence using multiscale variety. *Complexity*, 9(6):15–24.

Bedau, M. (1997). Weak emergence. *Philosophical Perspectives*, 11:375–399.

Bedau, M. (2003). Downward causation and the autonomy of weak emergence. *Principia*, 6:5–50.

Bertschinger, N., Olbrich, E., Ay, N., and Jost, J. (2008). Autonomy: an information theoretic perspective. *Biosystems*, page in press.

Broad, C., editor (1925). *The mind and its place in nature*. Routledge and Kegan Paul, London.

Chalmers, D. (2006). Strong and weak emergence. In Clayton, P. and Davies, P., editors, *The re-emergence of emergence*. Oxford University Press, Oxford.

Chang, H., editor (2004). *Inventing temperature: Measurement and scientific progress*. Oxford University Press, New York, NY.

Crutchfield, J. (1994). The calculi of emergence: Computation, dynamics, and induction. *Physica D*, 75:11–54.

Ding, M., Chen, Y., and Bressler, S. (2006). Granger causality: Basic theory and application to neuroscience. In Schelter, S., Winterhalder, M., and Timmer, J., editors, *Handbook of Time Series Analysis*, pages 438–460. Wiley, Wienheim.

Edelman, G. (2003). Naturalizing consciousness: A theoretical framework. *Proceedings of the National Academy of Sciences, USA*, 100(9):5520–5524.

Geweke, J. (1982). Measurement of linear dependence and feedback between multiple time series. *Journal of the American Statistical Association*, 77:304–13.

Granger, C. (1969). Investigating causal relations by econometric models and cross-spectral methods. *Econometrica*, 37:424–438.

Gregoire, G., Chate, H., and Tu, Y. (2003). Moving and staying together without a leader. *Physica D*, 181:151–170.

Jackson, F. and Pettit, P. (1992). In defence of explanatory ecumenism. *Economics and Philosophy*, 8:1–21.

Kim, J. (1999). Making sense of emergence. *Philosophical Studies*, 95:3–36.

Kim, J. (2006). Emergence: core ideas and issues. *Synthese*, 151:547–559.

Libet, B. (1985). Unconscious cerebral initiative and the role of conscious will in voluntary action. *Behavioral and brain sciences*, 8:529–566.

Polani, D. (2006). Emergence, intrinsic structure of information, and agenthood. *Interjournal Complex Systems*, page 1973.

Reynolds, C. (1987). Flocks, herds, and schools: A distributed behavioral model. *Computer Graphics*, 21(4):25–34.

Seth, A. (2005). Causal connectivity of evolved neural networks during behavior. *Network: Computation in Neural Systems*, 16:35–54.

Seth, A. (2007a). Granger causality. *Scholarpedia*, 2(7):1667.

Seth, A. (2007b). Measuring autonomy via multivariate autoregressive modelling. In Almeida e Costa, F., editor, *Proceedings of the Ninth European Conference on Artificial Life*, pages 475–485. Springer-Verlag.

Seth, A. (2008a). Causal networks in simulated neural systems. *Cognitive Neurodynamics*, 2:49–64.

Seth, A. (2008b). Functions of consciousness. In Banks, W., editor, *Springer Encyclopedia of Consciousness*, page in press. Springer Verlag.

Seth, A. and Edelman, G. (2008). Consciousness and complexity. In Meyers, R., editor, *Springer Encyclopedia of Complexity and Systems Science*, page in press. Springer Verlag.

Shalizi, C. and Moore, C. (2006). What is a macrostate: Subjective observations and objective dynamics. http://arxiv.org/abs/cond-mat/0303625.

Sperry, R. (1969). A modified concept of consciousness. *Psychological Review*, 76:532–536.

Tononi, G. and Edelman, G. (1998). Consciousness and complexity. *Science*, 282:1846–1851.

van Gulick, R. (2001). Reduction, emergence and other recent options on the mind-body problem. *Journal of Consciousness Studies*, 8:1–34.

Varela, F. (1979). *Principles of biological autonomy*. Elsevier, North Holland, NY.

Vicsek, T., Czirok, A., Ben-Jacob, E., Cohen, I., and Shochet, O. (1995). Novel type of phase transition in a system of self-driven particles. *Physical Review Letters*, 75:1226.

Modeling the Neural Basis of Cognitive Integration and Consciousness

Murray Shanahan and Dustin Connor

Department of Computing, Imperial College London,
180 Queen's Gate, London SW7 2AZ, United Kingdom
m.shanahan@imperial.c.uk

Abstract

This paper presents a number of models whose aim is to establish a computational basis for the hypothesis that conscious information processing in the brain is mediated by a mechanism of global broadcast. A possible role for this putative "global neuronal workspace" in achieving cognitive integration is mooted in the context of modular theories of mind, and an argument is advanced for its likely emergence within the sort of small-world brain network seemingly favoured by evolution. The paper concludes with some speculation on the relationship between life and consciousness as it could be.

Introduction

This article interweaves three strands of thinking in contemporary cognitive science. First, according to Baars (1988; 1997; 2002), the architecture of the mammalian brain comprises a number of parallel specialist processes (or modules) that compete and/or co-operate for access to a *global workspace*, in effect a mechanism for broadcasting information back to the whole cohort of specialists (Dehaene, *et al*., 2006; Shanahan, 2008a). The central claim of Baars's theory is that information processing which is local to the specialists is non-conscious and only broadcast information is consciously processed.

Second, advocates of modular theories of mind, despite the diversity of their views, are largely in agreement that some mechanism for transcending modular boundaries is a prerequisite for the highest levels of cognitive attainment (Fodor, 1983; 2002; Tooby & Cosmides, 1992; Mithen, 1996; Carruthers, 2002; 2006). This facilitates what Mithen (1996) calls *cognitive fluidity*, a capacity to integrate across distinct domains of expertise that promotes innovation and creativity (Wynn & Coolidge, 2004).

Third, it has been shown that cortical wiring in mammals exhibits the properties of a *small-world network* (Sporns & Zwi, 2004; Bassett & Bullmore, 2006). According to Striedter (2005), this is the consequence of evolutionary pressure to maintain communication between anatomically segregated regions in the face of an increasing neuron count, since this cannot go hand-in-hand with a proportional increase in connectivity

Drawing together these three themes, this article proposes that the long-range white matter connections that serve to keep down the average path length in large-scale cortical networks have been structured by evolution so as to develop into a global neuronal workspace (Dehaene & Naccache, 2001; Dehaene, *et al*., 2006; Shanahan, 2008a), which not only provides the integrative facility required to promote cognitive fluidity but is also a candidate for the neural substrate underlying consciousness (Shanahan & Baars, 2005). The argument draws on a variety of computer and robot models, and is capped with a short discussion of the relationship between consciousness as it is found in nature and consciousness as it could be.

Global workspace theory

Global workspace theory (Baars, 1988; 1997) is one of the most influential ideas in the burgeoning field of consciousness studies. Its basic tenets have been endorsed by respected philosophers (Dennett, 2001; Metzinger, 2003) and neuroscientists (Dehaene, *et al*. 1998; Dehaene & Naccache, 2001), and in cognitive psychology it has entered the undergraduate curriculum (Eysenck & Keane, 2005). Of course, the field is young and global workspace theory is open to future amendment or refutation. But it currently enjoys widespread support and a growing body of favourable evidence (Baars, 2002).

Central to the high-level, functional presentation of the theory is a computational architecture whose origins are in the blackboard systems of 1980s AI research. The architecture comprises a number of parallel, specialist processes and a *global workspace* (Fig. 1). The parallel specialists compete (and sometimes co-operate) to influence the global workspace, whose contents are broadcast back to the whole cohort of specialists, influencing them in turn. In operation, the architecture alternates between periods of competition and broadcast.

According to global workspace theory, the human brain instantiates the global workspace architecture, permitting a distinction to be drawn between conscious and non-conscious neural information processing. Information processing that takes place in the parallel specialists is non-conscious, while only information that is broadcast via the global workspace is consciously processed. Using the experimental paradigm of *contrastive analysis*, wherein closely matched conscious and non-conscious conditions are compared, this hypothesis can be tested empirically. Evidence to date using this method has been broadly supportive (Baars, 2002). Crucially, for

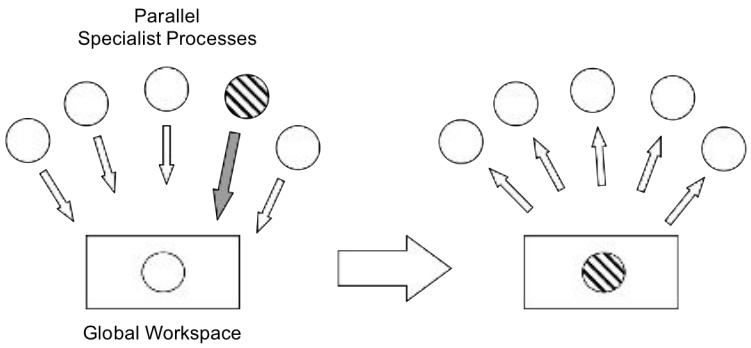


Fig. 1: The global workspace architecture. A set of parallel processes (shown as circles) compete for access to the global workspace (left). The winner (shown with hatched lines) influences the state of the global workspace, which is then broadcast back out to the whole cohort of processes (right). The resulting series of workspace states is the product of the repeated alternation between episodes of competition and broadcast.

contrastive analysis to be possible, both conscious and non-conscious processing must be capable of influencing behaviour. In the human case, introspective verbal report is typically taken as an index of conscious processing, while priming effects that occur in visual masking experiments are a good example of the influence of non-conscious processing (Breitmeyer & Öğmen, 2006).

Further support for global workspace theory can be garnered from its potential to bolster so-called modular theories of mind (Fodor, 1983; 2002; Tooby & Cosmides, 1992; Mithen, 1996; Carruthers, 2002; 2006). Modular theories of mind are challenged by the need for a mechanism that transcends modular boundaries, in order to implement what Fodor (2000) calls *informationally unencapsulated* cognitive processes, such as analogical reasoning, and to realise what Mithen (1996) calls *cognitive fluidity*. According to (Shanahan & Baars, 2005), the global workspace architecture incorporates just such a mechanism. Each parallel specialist process corresponds to a distinct module, and modular boundaries are transcended within the global workspace because the serial procession of states that unfolds there integrates the contributions of many of these parallel, specialists. Moreover, because the responsibility for determining the relevance of a potential contribution is not centralised but distributed among the specialists themselves, the resulting system is not vulnerable to the computational infeasibility arguments made by Fodor (1983; 2000).

One of the most pressing questions left open when global workspace theory is presented in functional terms is how the architecture maps onto the biological brain, and in particular what, in the brain, corresponds to the global workspace itself. A naive reading of the theory might attempt to associate the global workspace with a specific brain region, something reminiscent of the discredited notion of a Cartesian Theatre – "a place in the brain where it all comes together and consciousness happens" (Dennett, 1991). A more sophisticated understanding views the global workspace as an access-controlled, bandwidth-limited communications infrastructure that allows information to be distributed pan-cortically by means of global brain states. According to Dehaene and his colleagues, a *global neuronal workspace* of this sort is realised by the long-range cortico-cortical pathways of the cerebral white matter (Dehaene, *et al*., 1998; Dehaene & Naccache, 2001).

Modeling the global neuronal workspace

In order to realise a pan-cortical communications infrastructure and facilitate cognitive integration in accordance with the hypothesis of (Shanahan & Baars, 2005), the hypothesised global neuronal workspace should conform to the following four desiderata (Dehaene & Naccache, 2001; Shanahan, 2008a). 1) It should sustain reverberating patterns of activation over several tens of milliseconds. 2) It should disseminate (broadcast) patterns of activation throughout cortex, preserving the information inherent in their spatiotemporal structure. 3) It should be sensitive to new patterns of activation, and when overtaken by one only a trace should remain of any previous pattern. 4) Cortical populations should win the right to influence the pattern of activation in the workspace through competitive interaction.

One way to test the neurological plausibility of a global neuronal workspace conforming to these desiderata is to use the methods of computational neuroscience to build models of possible instantiations of the idea. In (Dehaene, *et al*., 2003) and (Dehaene & Changeux, 2005), a computer model is presented that simulates competitive access to a global neuronal workspace, emulating two well-known experimental phenomena, namely the attentional blink and inattentional blindness. But Dehaene's model does not simulate neuronal activity within the global workspace itself. In (Shanahan, 2008a), a complementary computer model is presented that simulates the (putative) global neuronal workspace itself in addition to a small number of cortical populations that compete to influence it (Fig. 2, left).

A schematic of the latter model is shown in Fig. 2 (right). Each box in the diagram represents a heterogenous population of over 1000 spiking neurons with conduction delays, implemented (in Matlab) using Izhikevich's (2003) equations. The global workspace comprises the five workspace nodes labeled W1 to W5, which serve to connect widely distributed regions of cortex. To keep the simulation manageable, only two such regions are included. Area W1 gives cortical

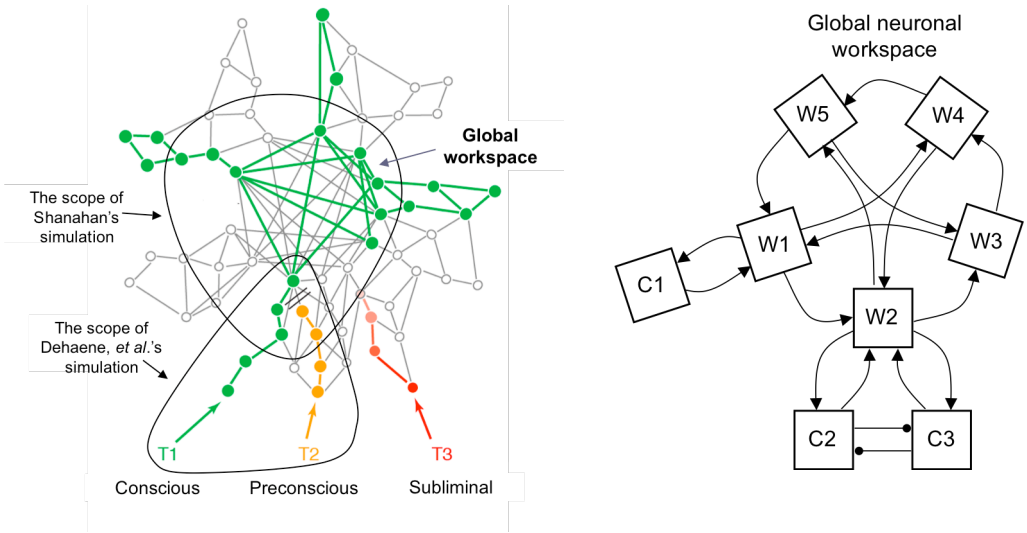


Fig. 2: The global neuronal workspace (left) and its model (right). The brains of cognitively sophisticated animals can be thought of as instantiating the architecture of Fig. 1, with the long-range fibres of the cerebral white matter constituting a global neuronal workspace (left, adapted from Dehaene, *et al.* (2006)). The schematic on the right depicts the computer simulation described in (Shanahan, 2008a).

population C1 access to the workspace, while area W2 gives populations C2 and C3 access to the workspace. C2 and C3 are in a competitive relationship, mediated by local inhibitory connections as shown. All of the excitatory connections shown are focal and topographically organised, ensuring that the spatial structure of an activation pattern is preserved as it spreads out from a cortical population and into the workspace. The inhibitory connections between C2 and C3, on the other hand, are diffuse.

Not shown, but present in the model, are further diffuse inhibitory connections among the workspace nodes. Given the extensive recurrent connections between workspace nodes and the potential for feedback these provide, a suitable balance of excitation and inhibition is required to promote reverberation without preventing new paterns of activation from invading the workspace (Wang, 2001). Transitions from one workspace state to another are achieved thanks to the cortical populations C1 to C3. Each of these is trained, using a form of spike-timing dependent plasticity (STDP), to respond to the appearance of a certain pattern Q in the workspace by taking on an associated pattern R, which may then invade the workspace in turn. Suppose pattern Q is presently in the workspace. If C2 associates Q with R and C3 associates Q with S then a competition will ensue. If C2 wins the competition, the next workspace state will be R. This in turn may stimulate another cortical area to respond (C1 perhaps). Overall, the system alternates periods of broadcast with bursts of competition, and the workspace exhibits a procession of

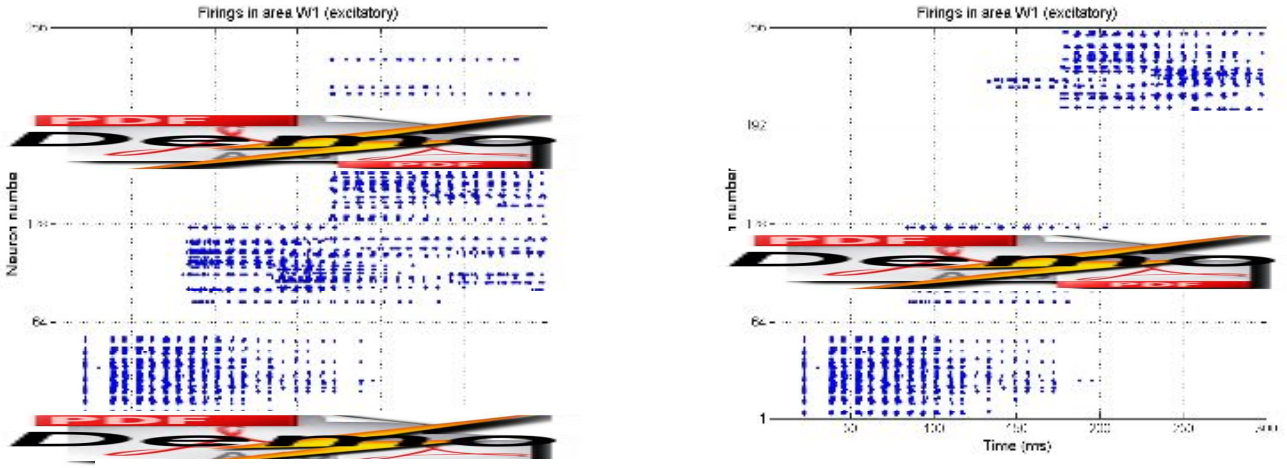


Fig. 3: Raster plots of neuron firings in two representative trials of the model of (Shanahan, 2008a). Both trials use the same network with identical synaptic weights. The difference is due to the competition between cortical popoulations C2 (influencing neurons 129–192) and C3 (influencing neurons 193–256), both of which respond equally strongly to activation in neurons 65 to 128, but with different associations. In the left-hand plot, C2 is the winner of the competition, shutting out its opponent by means of lateral inhibition, while in the right-hand plot the winner is C3.

broadcast states. Each of the components of the schematic in Fig. 2 (right) requires further internal structure to realise this behaviour. For full details the reader is referred to (Shanahan, 2008a) and (Shanahan, 2008b).

Fig. 3 shows raster plots of two representative trials of the simulation. For presentational purposes, the initial stimulus and the responses offered by C1 to C3 each activate a distinct set of contiguously numbered neurons. Firings in the excitatory neurons in workspace area W1 are shown. The other four workspace areas exhibit similar patterns, as we should expect if the workspace is operating effectively as a broadcast mechanism. In each trial, an initial stimulus is injected into the workspace at 20ms, which institutes a pattern of reverberating firing. C1 has an association with this particular pattern, and the pattern of firing it responds with begins to invade the workspace at around 80ms. This causes a surge of inhibition in the workspace thanks to which the original stimulus fades. By around 175ms almost no trace is left of it in either run.

At this point the two trials diverge. Areas C2 and C3 both have associations with the pattern of activation in the workspace, and a competition between them ensues. In the left-hand run C2 wins the competition, causing its response to take over the workspace, while in the right-hand run C3 is the victor. Note that in each case there is an outright winner, which prevents its rival from exercising any influence at all on the workspace. These trials were generated using the same network, with identical synaptic weights resulting from the same training run. The only source of difference between them is a small noise term added to the base current of each neuron. So taken together the two trials show that small differences at the level of individual neuron firings can result in qualitatively different sequences of workspace states at the macroscopic level (cf. Izhikevich & Edelman (2008)). A more complete description of the range of behaviours that can arise over multiple trials with differently trained networks can be found in Shanahan's papers (2008a; 2008b).

A workspace with stochastic wiring

The model of (Shanahan, 2008a) conforms to the desiderata set out earlier. But its neurological plausibility is compromised by the overly regular character of the workspace wiring. To address this shortcoming, work is ongoing to build and study global workspace models in which the long-range recurrent connections that promote reverberation and broadcast are established with a stochastic method that better reflects the statistical character of the evolutionary and developmental processes by which the brain's white matter pathways are formed.

In the new model, the 1280 excitatory workspace neurons, rather than being partitioned into five distinct sets as in the previous simulation, are arranged in a ring, which immediately induces a distance measure between any two neurons (Fig. 4, left). The workspace is then wired up by repeatedly forming circuits of connections. Each neuron in a circuit is selected randomly, subject to the constraint that no two neurons in the same circuit are allowed to be too close to each other, and the circuits are of variable length. Because each neuron is self-exciting via a circular route of connections, reverberating activation is promoted. But because recurrent connections cannot stimulate nearby neurons, the spatial organisation of a pattern of activation is preserved rather than smeared as it spreads throughout the workspace.

As with the previous model, excitatory influences in the workspace must be balanced with inhibitory connections, to ensure that reverberation is not so strong that it prevents new patterns from forming. In the stochastically-wired workspace, this is achieved using a second ring of 320 inhibitory neurons, concentric to the first. Each of these inhibitory neurons is excited locally, enabling it to detect patches of high firing, but has a widespread inhibitory effect on the workspace. The idea is to allow strong patterns of activation to damp rival workspace activity.

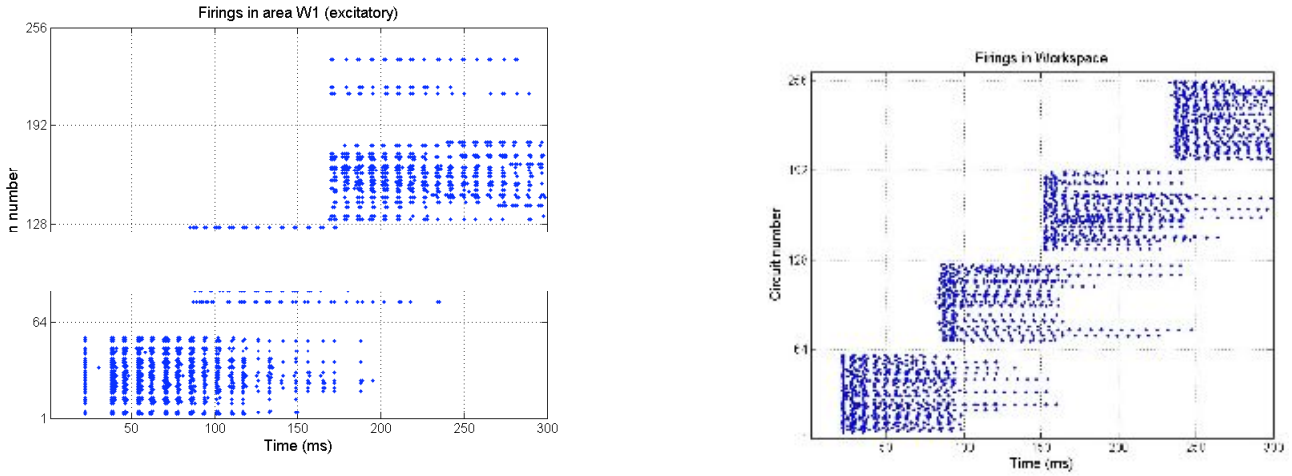


Fig. 4: A workspace with stochastic wiring. The workspace itself is a ring of neurons. Patterns of activation are broadcast (reverberate) around the ring via circuits of excitatory connections like the example shown on the left. Each circuit is wired up stochastically, but no two neurons in the same circuit are permitted to be close to eachother in the ring. Inhibitory neurons are locally excited but have diffuse influence (centre). The representative raster plot on the right shows that the workspace conforms to the desiderata.

Fig. 4 (right) shows a raster plot of a representative run in which a succession of four stimuli is delivered directly to the workspace. (The present model consists of the workspace only, and so far lacks the cortical populations of the previous model.) Each point in the plot represents that at least one neuron in the relevant circuit has fired. As the figure shows, the workspace maintains reverberation over several tens of milliseconds, and is susceptible to new patterns of activation which tend to push out their predecessors. In other words, the workspace conforms to three of the four desiderata proposed earlier, the fourth being inapplicable in the absence of cortical competition. Ongoing work aims systematically to map the range of model parameters and the space of possible network topologies that yield qualitatively equivalent behavioural characteristics.

Embodiment and cognitive architecture

The distinction between conscious and non-conscious processing that is the target of global workspace theory is only amenable to empirical investigation insofar as it impacts on outward behaviour. But the two computer models presented above are disembodied, closed systems. They must be embedded in a complete cognitive architecture if they are to stand as useful investigative tools. In (Shanahan, 2006), a cognitive architecture is presented that shows how a global workspace can be used in combination with an internally closed sensorimotor loop to realise a form of cognitively mediated action selection for a robot.

The central idea is that the internally closed sensorimotor loop permits the robot to rehearse trajectories through its sensorimotor space prior to enacting them (Hesslow, 2002). Rehearsed trajectories are evaluated, and the relative salience of the set of currently executable actions is modulated as a result – those initiating a trajectory whose outcome is associated with reward become more salient while those whose outcome is associated with aversion become less salient. Using a winner-takes-all strategy, the most promising action is selected and executed.

In the architecture of (Shanahan, 2006), the circuitry that makes up the inner sensorimotor loop takes in the global workspace itself (Fig. 5, left), and the series of rehearsed sensorimotor states unfolds within it. Hence these states are made available to the whole cohort of specialist networks that are attached to the workspace, enabling the trajectory of rehearsal to be determined by competition among those networks. Fig. 5 (right) presents the high-level schematic for a rationalised and extended version of the architecture. Internal sensorimotor activity, corresponding to that generated by the internally closed loop in Fig. 5 (left), results from mutual stimulation among motor and sensory areas, mediated by the global workspace. External sensory input causes activity in the sensory areas, which gives rise to activation in the workspace, from where it propagates to motor circuits. This stimulates a competition among motor areas to respond. During rehearsal, the resulting motor activity does not issue in overt behaviour, but instead gives rise to further, internally mediated stimulation of the sensory regions, completing the inner sensorimotor loop.

To date, the emphasis of our modeling work has been competition. Competitive access to the global workspace facilitates search through the sensorimotor space of a robot or animal because, as Fig. 3 shows, in cases where a sensorimotor state has multiple associations its successor in the workspace is non-deterministic. So revisiting a state can precipitate the rehearsal of an unexplored trajectory. But the hypothesis of the present paper is that the potential for co-operation among different networks might be equally

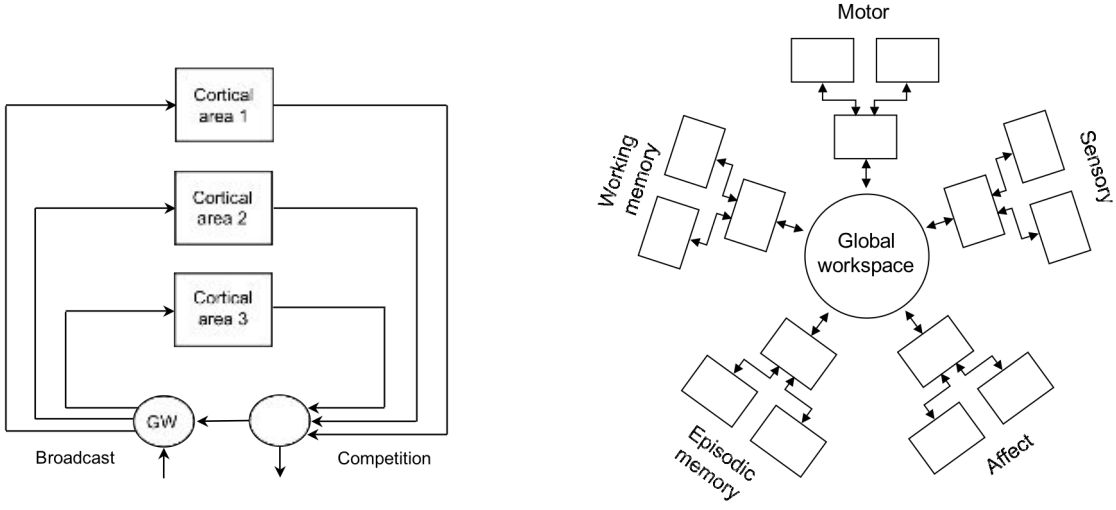


Fig. 5: Combining a global workspace with an inner sensorimotor loop (left), and a proposed rationalisation and extension of the architecture (right, cf. Fig. 1 of (Friston, 2003)). In both diagrams, information fans out from the global workspace into many distributed, parallel networks (broadcast) and funnelling back into it from those networks (competition). In the new architecture (right), five broad categories of functionally distinct networks are shown, each having a hierarchical structure. Co-operation, co-ordination, and competition among these networks is mediated by the global workspace, which best thought of as an access-controlled, bandwidth-limited communications infrastructure, rather than a functional component in its own right.

important. This is because co-operation may permit the adaptation and combination of elements from different parts of a learned repertoire of sensorimotor patterns, possibly enabling rehearsal even in a novel situation, such as that faced by an animal in the classic trap-tube test of causal understanding (Povinelli, 2000). To pass this test, an animal has to select the end from which to push a food item out of a transparent tube. The wrong choice results in the reward falling into a hole in its path which is visible to the animal. Some non-human animals, including chimpanzees and crows, are able to pass variants of this test, although there is no consensus among animal cognition researchers about how they do it (Seed, *et al*., 2006; Penn & Povinelli, 2007). A normal human adult, of course, is not unduly taxed this problem. Indeed, such capacity to innovate in the presence of novelty is often taken as a hallmark of human-level intelligence (Wynn & Coolidge, 2004).

According to the present hypothesis, the integrative facility of the global workspace supports the level of cognitive sophistication required to solve problems such as the trap-tube test. Networks encoding incompatible learned sensorimotor patterns are obliged to *compete* to influence the trajectory of rehearsal as it unfolds in the workspace. But where different networks encode compatible spatiotemporal patterns, they may be able to *co-operate*, allowing their respective influences to be blended together. Each specialist process may be thought of as encapsulating expertise in a particular micro-domain, such as dropping-things-in-holes or pushing-things-with-sticks. In effect, the global workspace promotes cognitive fluidity, permitting expertise in one micro-domain to be combined with expertise in another micro-domain (Shanahan & Baars, 2005). Our future work aims to explore this hypothesis with the aid of a large-scale spiking neuron implementation of the architecture of Fig. 5 (right), deployed to control a dextrous humanoid robot.

The emergence of a (small) world

Recent studies of neural connectivity lend further support to the hypothesis that cognitively proficient brains conform to the global workspace architecture. In particular, there is compelling evidence that human cortex constitutes a *small-world network* (Watts & Strogatz, 1998), which is a sparsely connected graph with a small mean path length and a large clustering coefficient. Consider a graph G comprising a set of nodes and edges. The *path length* between any pair of nodes in G is the number of edges in the shortest path between those nodes, and G's *mean path length* is the path length averaged over every pair of nodes in G. The *clustering coefficient* of a node P in G is the fraction of the set of all possible edges between immediate neighbours of P that are actual edges, and the *clustering coefficient* of the whole graph G is the clustering coefficient averaged over the set of all nodes in G. Many naturally occurring networks have been shown to have small-world properties, but our concern is only with those that are found in the brain.

A typical small-world network comprises numerous densely interconnected local clusters that are connected to each other via a small number of so-called *hub nodes* but are otherwise isolated. If the hub nodes have many edges compared to the cluster nodes then such a network may also be *scale-free*, meaning that the probability of a random node having k edges conforms to a power law – it is proportional to $k^{-\lambda}$ for some λ. However, as we shall use the term, a node does not require a large number of edges to be designated a hub. A hub node may, for example, be the only node in cluster A that is connected to a node in cluster B, thus helping to confer the small-world property on the overall graph. (In graph-theoretic terms, such nodes have low degree but high betweenness centrality.)

Even without formal analysis, it is easy to see that the topology of the model in (Shanahan, 2008) (Fig. 2, right) leads to small-world connectivity. First, thanks to the connections to, from, and between workspace neurons (the hub nodes), the maximum shortest path length between any two cortical neurons is just 6, even with the addition of further cortical areas (2 hops to get to a workspace area, 2 more to traverse the workspace, plus 2 to get out of the workspace). Second, the dense connectivity within cortical areas entails a high clustering coefficient. Finally, although the network is not especially sparse with only C1 to C3 attached to the workspace, its sparseness increases rapidly with the addition of further cortical areas. Similar considerations apply with the stochastically wired workspace.

Using neuroanatomically established connectivity matrices, it has been shown that the cortices of cats and macaques enjoy small-world network properties (Hilgetag, *et al*., 2000; Sporns & Zwi, 2004). Moreover, several recent *in vivo* studies purport to establish similar results for human cortex. Using fMRI, Eguíluz, *et al*. (2005) revealed a network of functional brain connections that conform to the power law characteristic of a scale-free, small-world network. Also using fMRI, Achard, *et al*. (2006) confirmed this result and built a connectivity map of the cortical hub nodes underlying it. At the structural level, He, *et al*. (2007) supply a similar map by correlating measures of cortical thickness in different brain regions obtained by MRI.

The question of why evolution should favour neural networks with small-world properties naturally arises. A number of answers have been suggested (Bassett & Bullmore, 2006). Wiring cost is likely to be one major factor (Striedter, 2005; Wen & Chklovskii, 2006). If connectivity is maintained as brains increase in neuron count, then the quantity of wiring must increase too. Wiring is costly "due to metabolic energy required for maintenance and conduction, guidance mechanisms in development, conduction time delays and attenuation, and wiring volume" (Wen & Chklovskii, 2006, p.0617). But pressure to minimise wiring can lead to a network that is segregated into clusters (or modules). A small-world network compensates for this by allowing effective communication to be maintained between distant regions (Striedter, 2005). At the same time, pressure to minimise conduction delays may also lead to small-world properties, as well as the division of the brain into grey and white matter (Wen & Chklovskii, 2006).

In addition to their favourable wiring cost, small-world networks have been shown to possess information processing characteristics that make them especially well-suited to realising a global neuronal workspace. Specifically, Sporns, *et al* (2000) argue that small-world networks support high dynamical "complexity", according to a formal measure that assesses the co-existence in a network of functional

specialisation and integration (Tononi, *et al*., 1998; Seth, *et al*., 2006). According to this measure, the complexity of a system X comprising n variables x_i is approximated by the function $C(X)$, given by

$$C(X) = H(X) - \sum H(x_i \mid X - x_i)$$

where $H(Y)$ is the entropy of a system Y and $H(y \mid Y)$ is the conditional entropy of y given Y. In essence, if a system has a low level of integration then values for $H(x_i \mid X)$ will be high, while if the system has a low degree of specialisation the value of $H(X)$ will be low. Using an evolutionary algorithm, Sporns, *et al*. searched a space of possible network topologies, selecting for networks with high $C(X)$. A typical network obtained after 2000 generations with this method had a mean path length comparable to that of an equivalent random graph, but a significantly higher clustering coefficient.

Intuitively, this result makes perfect sense. At a local level, the densely interconnected clusters of a small-world network are functionally segregated, while at a global level the connections between hub nodes ensure that the network's overall activity has widespread local influence. Moreover, it should be clear that a capacity to support high dynamical complexity in the sense quantified by $C(X)$ is a prerequisite for any neural network instantiation of the global workspace architecture, and that a network with small-world properties supplies the means to fulfil this prerequisite. The local specialists of the global workspace architecture can be realised by the highly interconnected, functionally segregated clusters of a small-world network, ensuring a high value for $H(X)$, while the global workspace itself is realisable by a web of hub-node-to-hub-node connections, promoting low values for $H(x_i \mid X)$.

Additional organisation over and above small-world topology is required for a network to conform to the desiderata set out earlier and realise the function of a global neuronal workspace. But only a relatively conservative set of modifications to the hub node connections of a sufficiently large small-world network may be needed for their integrative potential to be recruited to this role. Of course, once these modifications have been selected for, their cognitive advantages will ensure their perpetuation. But it is an intriguing thought that consciousness might initially have arisen only as a side-effect of the evolutionary pressure to keep wiring cost down, a constraint that applies across the phylogenetic scale from *C.elegans* upwards, but which ensures that the necessary infrastructure to support the distinction between conscious and non-conscious processing is already in place as neuron count goes up.

Consciousness as it could be

Artificial life, according to one of the field's founders, "can contribute to theoretical biology by locating *life-as-we-know-it* within the larger picture of *life-as-it-could-be*" (Langton, 1989, p.1). In a similar vein, the use of computer and robot models might aspire to contribute to cognitive science by situating consciousness as we know it within the larger picture of consciousness as it could be. No less interesting is the challenge of situating consciousness as it could be in relation to life as it could be. Indeed several authors argue for the deep continuity of life and mind: "life and mind share a set of basic organizational properties, and the organizational properties distinctive of mind are an enriched version of those fundamental to life" (Thompson, 2007, p. 128).

The argument for this position is roughly as follows. An organism perpetually constitutes its own identity through metabolic exchange of matter and energy with the environment so as to maintain the boundary between self and non-self. At the same time this "autopoietic" process brings forth a domain of concern, wherein features of the environment acquire significance according to their relevance to that organism's wellbeing and perpetuation. Moreover, an organism's need constantly to change in order simply to maintain its identity opens up what phenomenologists call a temporal and spatial "horizon" for that organism. For phenomenologists, such a "horizon of transcendence" is also a necessary feature of lived experience, motivating the conclusion that "certain existential structures of human life are an enriched version of those constitutive of all life" (Thompson, 2007, p.157).

Let's review the principles of organisation claimed in this paper to be fundamental to consciousness, and consider the extent to which they resonate with the thesis of deep continuity of life and mind. The global workspace architecture harnesses the power of massively parallel computation. The global workspace itself exhibits a *serial* procession of states, yet each state-to-state transition is the result of filtering and integrating the contributions of huge numbers of *parallel* computations. In essence, the architecture thereby distils *unity* out of *multiplicity*. This unity is achieved within the global workspace itself, which is both the source and sink of information in the fan and funnel model (Fig. 5, left). But it is also a locus of control, and the informatic singularity of the global workspace is inherently bound to the spatially localised body whose control is in question, the point of convergence of perception and action (Legrand, 2006). The *remit* of all the processes that are brought into unity by the global workspace is duly inherited from the body to which it is bound (Shanahan, 2005). Everything they do pertains to, or is *indexical* to, that body and its point of view.

In the natural world this remit in large part subserves metabolism, and is plausibly cast in terms of autopoiesis. But in the realm of the possible, of consciousness as it could be, metabolism is not a prerequisite for being a centre of concern, for possessing self-related purpose within a spatial and temporal "horizon of transcendence". In a properly embodied instantiation of the global workspace architecture, the identity of the conscious subject is underwritten by the common remit of a set of processes that pertain to the past, present, and future of the spatially localised body to which they are all indexically oriented (Fig. 5, right). In conclusion, however formidable the practical obstacles might be to creating a conscious artefact, the absence of metabolism presents no obvious theoretical obstacle. Perhaps the appeal of the deep continuity thesis is attenuated by this caveat.

References

Achard, S., Salvador, R., Whitcher, B., Suckling, J. & Bullmore, E. (2006). A Resilient, Low-Frequency, Small-World, Human Brain Functional Network with Highly-Connected Association Cortical Hubs. *Journal of Neuroscience* 26 (1), 63–72.

Baars, B.J. (1988). *A Cognitive Theory of Consciousness*. Cambridge University Press.

Baars, B.J. (1997). *In the Theater of Consciousness: The Workspace of the Mind*. Oxford University Press.

Baars, B.J. (2002). The Conscious Access Hypothesis: Origins and Recent Evidence. *Trends in Cognitive Sciences* 6 (1), 47–52.

Bassett, D.S. & Bullmore, E. (2006). Small-World Brain Networks. *The Neuroscientist* 12 (6), 512–523.

Breitmeyer, B. & Öğmen, H. (2006). *Visual Masking: Time Slices Through Conscious and Unconscious Vision*. Oxford University Press.

Carruthers, P. (2002). The Cognitive Functions of Language. *Behavioral and Brain Sciences* 25 (6), 657–674.

Carruthers, P. (2006). *The Architecture of the Mind*. Oxford University Press.

Dehaene, S. & Naccache, L. (2001). Towards a Cognitive Neuroscience of Consciousness: Basic Evidence and a Workspace Framework. *Cognition* 79, 1–37.

Dehaene, S., Changeux, J.-P., Naccache, L., Sackur, J. & Sergant, C. (2006). Conscious, Preconscious, and Subliminal Processing: A Testable Taxonomoy. *Trends in Cognitive Sciences* 10 (5), 204–211.

Dehaene, S., & Changeux, J.-P. (2005). Ongoing Spontaneous Activity Controls Access to Consciousness: a Neuronal Model for Inattentional Blindness. *PLoS Biology* 3 (5), e141.

Dehaene, S., Kerszberg, M. & Changeux, J.-P. (1998). A Neuronal Model of a Global Workspace in Effortful Cognitive Tasks. *Proceedings of the National Academy of Science* 95, 14529–14534.

Dehaene, S., Sergent, C., & Changeux, J.-P. (2003). A Neuronal Network Model Linking Subjective Reports and Objective Physiological Data During Conscious Perception. *Proc. National Academy of Sciences* 100 (14), 8520–8525.

Dennett, D. (1991). *Consciousness Explained*. Penguin.

Dennett, D. (2001). Are We Explaining Consciousness Yet? *Cognition* 79, 221–237.

Eguíluz, V.M., Chialvo, D.R., Cecchi, G.A., Baliki, M. & Apkarian, A.V. (2005). Scale-Free Brain Functional Networks. *Physical Review Letters* 94, 018102.

Eysenck, M.W. & Keane, M.T. (2005). *Cognitive Psychology; A Student's Handbook*. Fifth edition. Psychology Press.

Fodor, J.A. (1983). *The Modularity of Mind*. MIT Press.

Fodor, J.A. (2000). *The Mind Doesn't Work That Way*. MIT Press.

Friston, K. (2003). Learning and Inference in the Brain. *Neural Networks* 16, 1325–1352.

He, Y., Chen, Z.J. & Evans, A.C. (2007). Small-World Anatomical Networks in the Human Brain Revealed by Cortical Thickness from MRI. *Cerebral Cortex* 17, 2407–2419.

Hesslow, G. (2002). Conscious Thought as Simulation of Behaviour and Perception. *Trends in Cognitive Sciences* 6 (6), 242–247.

Hilgetag, C.-C., Burns, G.A.P.C, O'Neill, M.A., Scannell, J.W. & Young, M.P. (2000). Anatomical Connectivity Defines the Organization of Clusters of Cortical Areas in the Macaque Monkey and the Cat. *Philosophical Transactions of the Royal Society B* 355, 91–110.

Izhikevich, E. M. (2003). Simple Model of Spiking Neurons. *IEEE Transactions on Neural Networks* 14, 1569–1572.

Izhikevich, E.M. & Edelman, G.M. (2008). Large-Scale Model of Mammalian Thalamocortical Systems. *Proc. National Academy of Sciences* 105 (9), 3593–3598.

Langton, C. (1989). Artificial Life. In C.Langton (Ed.), *Artificial Life*, Addison Wesley, pp. 1–47.

Legrand, D. (2006). The Bodily Self: The Sensori-motor Roots of Pre-reflexive Self-Consciousness. *Phenomenology and the Cognitive Sciences* 5, 89–118.

Metzinger, T. (2003). *Being No One: The Self-Model Theory of Subjectivity*. MIT Press.

Mithen, S. (1996). *The Prehistory of the Mind*. Thames & Hudson.

Penn, D.C. & Povinelli, D.J. (2007). Causal Cognition in Human and Nonhuman Animals: A Comparative, Critical Review. *Annual Review of Psychology* 58, 97–118.

Povinelli, D.J. (2000). *Folk Physics for Apes*. Oxford University Press.

Seed, A.M., Tebbich, S., Emery, N.J. & Clayton, N.S. (2006). Investigating Physical Cognition in Rooks, *Corvus frugilegus*. *Current Biology* 16, 697–701.

Seth, A.K., Izhikevich, E., Reeke, G.N. & Edelman, G.M. (2006). Theories and Measures of Consciousness: An Extended Framework. *Proc. National Academy of Science* 103 (28), 10799–10844.

Shanahan, M.P. (2005). Global Access, Embodiment, and the Conscious Subject. *Journal of Consciousness Studies* 12 (12), 46–66.

Shanahan, M. P. (2006). A Cognitive Architecture that Combines Internal Simulation with a Global Workspace. *Consciousness and Cognition* 15, 433–449.

Shanahan, M.P. (2008a). A Spiking Neuron Model of Cortical Broadcast and Competition. *Consciousness and Cognition* 17, 288–303.

Shanahan, M.P. (2008b). Supplementary Note on "A Spiking Neuron Model of Cortical Broadcast and Competition". *Consciousness and Cognition* 17, 304–306.

Shanahan, M.P. & Baars, B. (2005). Applying Global Workspace Theory to the Frame Problem. *Cognition* 98 (2), 157–176.

Sporns, O., Chialvo, D.R., Kaiser, M. & Hilgetag, C.-C. (2004). Organization, Development and Function of Complex Brain Networks. *Trends in Cognitive Sciences* 8 (9), 418–425.

Sporns, O., Tononi, G. & Edelman, G.M. (2000). Theoretical Neuroanatomy: Relating Anatomical and Functional Connectivity in Graphs and Cortical Connection Matrices. *Cerebral Cortex* 10, 127–141.

Sporns, O. & Zwi, J.D. (2004). The Small World of the Cerebral Cortex. *Neuroinformatics* 2 (2), 145–162.

Striedter, G.F. (2005). *Principles of Brain Evolution*. Sinaur Associates, Inc.

Thompson, E. (2007). *Mind in Life: Biology, Phenomenology, and the Sciences of Mind*. Harvard University Press.

Tononi, G., Edelman, G.M. & Sporns, O. (1998). Complexity and Coherency: Integrating Information in the Brain. *Trends in Cognitive Sciences* 2 (12), 474–484.

Tooby, J., & Cosmides, L. (1992). The Psychological Foundations of Culture. In J. H. Barkow, L. Cosmides, & J.Tooby (Eds.), *The Adapted Mind*, Oxford University Press, pp. 19–136.

Watts, D.J. & Strogatz, S.H. (1998). Collective Dynamic of 'Small-world' Networks. *Nature* 393, 440–442.

Wang, X.-J. (2001). Synaptic Reverberation Underlying Mnemonic Persistent Activity. *Trends in Neuroscience* 24 (8), 455–463.

Wen, Q. & Chklovskii, D.B. (2006). Segregation of the Brain into Gray and White Matter: A Design Minimizing Conduction Delays. *PLoS Computational Biology* 1 (7): e78.

Wynn, T. & Coolidge, F.L. (2004). The Expert Neandertal Mind. *Journal of Human Evolution* 46, 467–487.

Evolving an Agent-Based Model to Probe Behavioral Rules in Flocks of Cowbirds

V. Anne Smith[1]

[1]School of Biology, University of St Andrews, St Andrews, KY16 9TH, United Kingdom
anne.smith@st-andrews.ac.uk

Abstract

Flocks of brown-headed cowbirds, *Molothrus ater*, self-organize social environments, which have strong impacts on social learning and behavior. To understand the rules underlying self-organization of the social environment, I develop an agent-based model of cowbird social association and evolve it to match observed patterns of association measured from real birds. The behavioral rules evolved in the model provide insight into the type of rules real birds use to organize their social environment. The evolved models successfully predicted both association patterns and additional related movement variables measured from a new flock of birds.

Introduction

Animal behavior often occurs embedded within the complex system of a group of interacting animals; however, traditional methods for elucidating behavioral rules include reductionist techniques that can destroy the very social environment necessary for the behavior of interest (Schank, 2001). Within the last decade, studying animal behavior within its natural, complex context has been enabled through use of agent-based models (Schank and Alberts, 1997; Powell et al., 1999; Schank and Alberts, 2000; Jackson et al., 2004; Bryson et al., 2007; Sellers et al., 2007). These models typically implement hypotheses about how individual animals move, make decisions, etc., and are then evaluated for ability to match emergent properties of situations they model, such as breeding productivity (Powell et al., 1999) or group decision making (Sellers et al., 2007). But this human-designed model building leaves open the possibility that other, non-considered scenarios could also match the observed behavior (Bryson et al., 2007). Schank and Alberts (2000) pioneered a method to address this issue: allowing heuristic search to optimize parameters of an agent-based model, thus reducing bias from pre-conceived notions. Here, I take this approach, developing an agent-based model, and evolving it with a genetic algorithm (GA), to elucidate general principles underlying behavioral rules used in social assortment of brown-headed cowbirds, *Molothrus ater*.

The brown-headed cowbird is an obligate brood parasite: females lay their eggs in the nests of other species, leaving the host species to raise their young. Because of this behavior, cowbirds were long thought to be exemplars of instinctual control of all aspects of social behavior, and in particular, mating behavior (Mayr, 1974). However, modern research revealed that cowbirds rely heavily on social learning for social interactions, including mating preferences and appropriate courtship behavior (King and West, 1989; Freeberg et al., 1995). This learning occurs when adult and juvenile cowbirds gather in large over-winter flocks (Friedmann, 1929; King and West, 1988).

These over-winter flocks have recently been shown to have strong self-organized patterns of social association based on age and sex (Smith et al., 2002). Furthermore, the make-up of the social environment surrounding a juvenile male within this self-organized pattern correlates with his singing behavior and courtship success (Smith, 2001; Smith et al., 2002), and experimentally induced differences in social environment in a flock can radically adjust many aspects of birds' future social and mating behavior (King et al., 2002; West et al., 2002; White et al., 2002a,b,c). Thus, the self-organized social environment provides the scaffold surrounding social learning in this brood-parasitic species.

However, the mechanism behind this self-organization is unclear: it must be assumed some preferential approach or avoidance occurs, but features such as specificity of preference and if all birds, or only some ages/sexes, drive the patterns, is unknown. To investigate mechanisms underlying such self-organization, I develop an agent-based model of cowbird social association, using a modified classifier system where movement is controlled by a set of interpretable rules. I evolve this model to match the self-organized association patterns seen in one group of birds, then use the evolved models to predict association patterns as well as other movement variables from a new group of birds.[1]

Agent-Based Model

Because the aim of this work is to gain insight into rules birds use to self-organize their social environment, I chose to model individual cowbirds as modified classifier systems

[1]Model and GA created in C++ and available upon request.

(Holland et al., 1986; Booker et al., 1989). The if-then statements in the classifier are easily interpretable as choices made by a bird about its future behavior based on its current environment. The traditional classifier system was modified in two major ways for this agent-based model: (1) choice of classifiers was performed probabilistically based on strength to simulate the stochastic nature of animal behavior and (2) learning did not occur through reinforcement nor evolution of an individual agent's classifiers; instead, all parameters of the model controlling bird behavior, including classifiers, were evolved with a GA based on association patterns resulting from the interaction of multiple agents.

A model was characterized by the number of bird-agents of each of four possible age and sex classes (AM: adult male, JM: juvenile male, AF: adult female, JF: juvenile female) and model parameters controlling the behavior of each class.

Model Parameters

Two activity-state probabilities controlled probability of a bird becoming (1) active if inactive and (2) inactive if active. Only active birds moved in relation to their environment. This distinction allows modeling of situations where birds may be unresponsive to social environment, for example sheltering from weather or predators (Smith, 2001).

A list of classifiers governed each birds' behavior: when the environment matched conditions described by all five bits of an if-statement, the five bits of its then-statement directed a potential behavior (Table 1). The if-statement relates to the social environment: a bird was aware of neighboring birds if they were within 15 units; neighbors were near if within 5 units and far otherwise; neighbors' age and sex was noted; and if this neighbor had the same relationship in the last time step (old) or not (new). Eighteen distinct environmental conditions are thus represented; wild-cards in the if-statement enable a classifier to apply to multiple conditions. The near and aware distances mimic distances relevant to cowbirds: cowbird song degrades rapidly beyond 0.3 m (King et al., 1981), making it a socially relevant "near" distance (Smith et al., 2002); birds engage in social interactions from as far as 0.9 m, and social companions within this distance influence social learning (Smith, 2001).

The then-statement relates to birds' movement choices: whether to move or be still; if moving, to move in a directed manner related to a neighboring bird or randomly; and if moving directed, to move towards or away from the other bird. Additionally, two bits adjusted the overall activity state of a bird, making it active or inactive.

Finally, each classifier had an integer strength S which influenced the probability of its being chosen to perform.

These model parameters are what is evolved by the GA based on assortment patterns arising from individual agent-birds executing repeated classifier system cycles.

Bit	**Interpretation of Value**	
If	**1 (on)**	**0 (off)**
neighbor	aware of bird	no birds
distance	near	far
age	adult	juvenile
sex	female	male
time	new	old
Then	**1 (on)**	**0 (off)**
move	move	do not move
directed	move in relation to bird	move randomly
to	move toward	move away
inactive	become inactive	no change
active	become active	no change

Table 1: Interpretation of possible values in classifier statements. The if-statement could also include wildcards, which would match conditions corresponding to either value.

Classifier System Cycle

The agent-based model runs by each individual bird going through a classifier system cycle, consisting of: detect, match, select, and effect. Birds are modeled in an 90x90 unit artificial world; each bird is identified with its age (adult or juvenile) and sex (male or female), and a unique number. Before each cycle, each bird determines its activity-state based on its activity-state probabilities. All birds go through the cycle whether or not they are active.

Detect. Each bird populates a message board with messages in the same format as the if-statement of classifiers. A message is produced for every bird within its awareness distance, noting distance, age, and sex. The message is flagged with the neighbor's identifying number and compared to a list of messages from the previous time step: if identical conditions for this neighbor exist, the message is set as old; otherwise, new. If no other birds are in the individual's awareness distance, a single "no bird" message is produced, with wildcards in the distance, age, and sex bits, and the time bit reflecting whether lack of neighbors is new or old.

Match. Every message is compared to the if-statement of the bird's classifiers, and added to a matched set if all non-wildcard bits are identical. Classifiers in the matched set are flagged with the identity of neighboring birds; a classifier may be added to the matched set multiple times for different neighboring birds. If no classifiers match the message(s) on the message board, the matched set is empty.

Select. One classifier from the matched set is selected to perform; a classifier's probability of being chosen is equal to its strength's proportion of the total strength extant in the matched set. The classifiers in the matched set are ordered arbitrarily and contiguously assigned spans of integers equal

in size to their strengths; i.e., classifier c with strength $S_c>0$ is assigned integers $\{A_{c1}, \ldots, A_{cS_c}\}$, where:

$$A_{c1} = 1 + \sum_{n=1}^{c-1} S_n$$

$$A_{cS_c} = S_c + \sum_{n=1}^{c-1} S_n$$

If $S_c=0$, the classifier is assigned no integers. One classifier is selected when a random integer, chosen between 1 and the sum of all strengths in the matched set, falls within its assigned span. If the matched set is empty, a null classifier is passed to the next stage.

Effect. From the then-statement of the chosen classifier, a bird first determines any changes to its activity-state due to the active- and inactive-bits. The inactive-bit is processed first, such that the active-bit can "mask" the inactive-bit.

Birds that are inactive, have a null classifier, or an off movement-bit remain still. Active birds with an on movement-bit analyze the remainder of the classifier: if the directed-bit is off, they move 4 units in a random direction; if it is on, they move 4 units toward (on to-bit) or away (off to-bit) from the neighboring bird flagged on that classifier. If a "no bird" classifier has a flagged directed-bit, movement is in a random direction. If movement would take a bird out of the 90x90 world, they are stopped on the outside boundary.

Running the Agent-Based Model

A run of the agent-based model begins by creating the appropriate number of bird-agents of each of the four age and sex classes; each bird receives parameters specific to its age and sex. Birds are placed randomly in the 90x90 world, and 50 classifier system cycles are run to allow the birds to develop assortment. A further 300 classifier system cycles are run while the GA collects the data for its fitness function.

Genetic Algorithm

The GA evolves populations of agent-based models, storing parameters necessary to define a model in five chromosomes: one contains activity-state probabilities for all four age and sex classes; the remaining four contain the list of classifiers specific to each age and sex. Each classifier is stored as: 5 if-bits, 5 wild-bits which when on, make the corresponding if-bit wild, 5 then-bits, then a strength.

Fitness Evaluation

Model fitness is estimated as an average calculated from 200 runs (repetitions determined by power analysis). Fitness is based on match of association patterns to an ideal pattern, defined by proportions of near neighbor points (PNN) for each age and sex class with all others. PNN_{IJ} of class I with class J is calculated as in behavioral experiments (Smith et al., 2002; White et al., 2002b) using points per bird (NN_{iJ}) as counts of near neighbor observations of bird i with bird j (NN_{ij}) normalized to size of class J (subtracting self when normalizing with own class):

$$\mathrm{NN}_{iJ} = \frac{\sum_{j \in J} \mathrm{NN}_{ij}}{|J \backslash \{i\}|}$$

$$\mathrm{PNN}_{IJ} = \frac{\sum_{i \in I} (\mathrm{NN}_{iJ} / \sum_K \mathrm{NN}_{iK})}{|I|}, K \in \{\mathrm{AM}, \mathrm{JM}, \mathrm{AF}, \mathrm{JF}\}$$

The NN_{ij} values are collected from the last 300 classifier system cycles of each model run: NN_{ij} is increased by 1 after every cycle for which bird j is near (within 5 units of) bird i. Fitness F of a model run is then calculated as:

$$F = 16 - \sum_I \sum_J \begin{cases} \left(\frac{\mathrm{PNN}_{IJ}^{\mathrm{ideal}} - \mathrm{PNN}_{IJ}^{\mathrm{measured}}}{\mathrm{PNN}_{IJ}^{\mathrm{ideal}}} \right)^2, & \mathrm{PNN}_{IJ}^{\mathrm{ideal}} > 0 \\ \left(\mathrm{PNN}_{IJ}^{\mathrm{measured}} \right)^2, & \mathrm{PNN}_{IJ}^{\mathrm{ideal}} = 0 \end{cases}$$

Evolution Process

A generation starts with 10 seed models, representing the 10 models with the highest fitness last generation, and 5 new random models (to maintain variation in the population). In the initial generation, the 10 seed models are also randomly generated. For random generation, values are chosen between: 0-1 for the 8 activity-state probabilities; 1-6 classifiers for each age and sex class; on/off for the 15 bits in each classifier; and integers 0-50 for each strength.

Each generation the 10 seed models mate to produce 35 offspring (see modification later). A mating produces one offspring from two models, each randomly assigned to be mother or father. Crossover occurs for each chromosome with a probability of 0.5 (otherwise, offspring receive mothers' chromosomes). In probability chromosomes, a crossover probability is chosen: the offspring receives the father's chromosome up to and including that probability and the mother's thereafter. In classifier chromosomes, a crossover classifier from both the mother and father and a crossover point within the classifier is chosen: the offspring receives the father's list of classifiers up to and including the crossover point in the chosen father's classifier and the mother's list after that point in her chosen classifier and thereafter. The offspring's list is truncated to 6 classifiers, to prevent overgrowth of classifier lists which was otherwise rampant. After crossover, point mutations occurr at each chromosome element with probability 0.01: probabilities add or subtract a value 0-0.2; classifier bits flip state; classifier strengths add or subtract an integer 0-3.

The 10 highest fitness models are chosen to seed the next generation. The GA is run for 40-50 generations.

Evolving the Agent-Based Model

Initial verification

Before evolving models to match movement patterns of real birds, models were evolved to a test situation of total own-

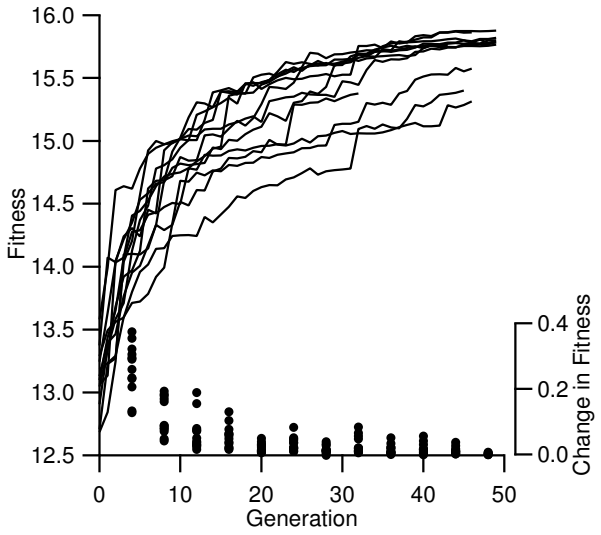


Figure 1: Fitness of best model for 12 runs of GA at each generation (lines) and change in fitness averaged every four generations (dots).

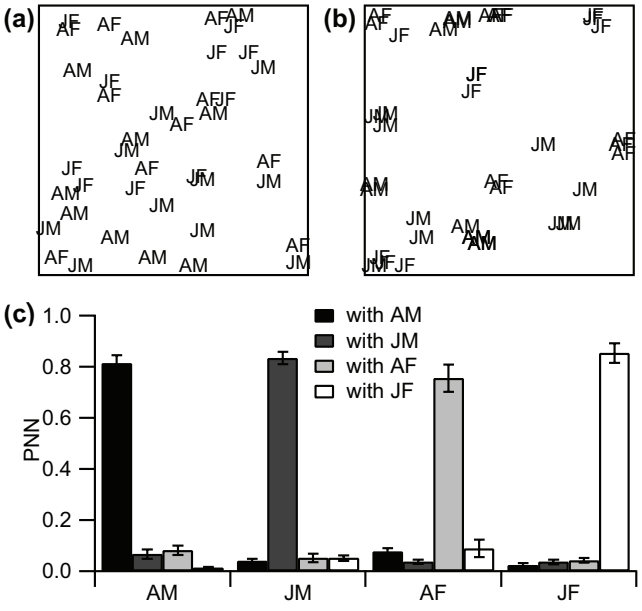


Figure 2: Evolution for total assortment. **(a)** Initial random placement of birds in world; **(b)** assortment at end of run following evolved rules. **(c)** PNN averaged over all 12 evolved models. Error bars represent standard error of the mean.

class association, i.e., $\mathrm{PNN}_{IJ} = 1$ for $I = J$ and 0 otherwise, in order to investigate behavior of the GA and model. Models were set up with 10 members of each age and sex class, and the GA was run 12 times to produce 12 models.

Genetic algorithm performance. The GA succeeded in increasing fitness of the models from their initial random starting points (Fig.1), reaching fitness mean=15.7±SE=0.06 of maximum 16. The average change in fitness each generation was significantly positive every four generations through generation 45 ($t_{9\text{-}11} \geq 3.4$, $P \leq 0.007$), and not thereafter (t_5=1.9, P=0.12).

Evolved model performance. Evolved models showed desired assortment patterns, with birds gathering in small clumps of same age and sex class (Fig.2a,b). Perfect total assortment was not achieved, but evolved models reached about 0.8 PNN for own class and near 0 for others (Fig.2c).

Analysis of evolved parameters. To examine how evolution affected the models, evolved parameters were evaluated for their deviation from randomness.

There was no evidence of either directional selection on activity probabilities P ($|t_{47}| \leq 0.2$, $P \geq 0.8$, one sample t-test H_0=$E(P)$=0.5) or stabilizing selection for values of 0.5 ($|t_{47}| \leq 0.8$, $P \geq 0.5$, one sample t-test H_0=$E(|P-0.5|)$=0.25). However, the active-bit in the classifier was significantly more often on (χ^2_1=23.2, P<0.0001). When not masked by an on active-bit, the inactive-bit was more often off (χ^2_1=31.8, P<0.0001); when masked, there was no difference (χ^2_1=1.5, P=0.2).

All bits in the remainder of the then-statement showed evidence of selection: the move-bit and directed-bit more often on ($\chi^2_1 \geq 33.6$, P<0.0001), and movements more often toward the other bird (χ^2_1=7.5, P=0.006).

In the if-statement, all bits were more often wild than not ($\chi^2_1 \geq 9.7$, $P \leq 0.002$). When not masked by a wild, the time-bit was more often set to old (χ^2_1=10.4, P=0.001); the age-bit to juveniles (χ^2_1=4.1, P=0.04); and there was no difference in the rest ($\chi^2_1 \leq 3.4$, $P \geq 0.07$).

In order to determine if different strategies had evolved for remaining near one's class, classifier lists were divided into two groups: those which contain classifiers directing a move toward a neighbor of the same class in (1) more than half or (2) less than half of possible environmental conditions containing such a bird. Classifier lists in the second group more often simply had no classifiers at all that could respond to the same class (Mann-Whitney $U_{N=19,29}$=76, P<0.0001), suggesting that the different behaviors were due to evolutionary constraints rather than different strategies.

Evolution to match real bird behavior

The agent-based models were evolved to match association patterns measured from real birds in Smith et al. (2002) during a subset of the Spring sample (22 Mar - 14 Apr); this consists of PNN for birds within 0.3 m and shows the typical stair-step pattern of increasing association with birds similar first in sex then age (Fig.3). The models were set up with 19 AM, 14 JM, 21 AF, and 16 JF to match age and sex distri-

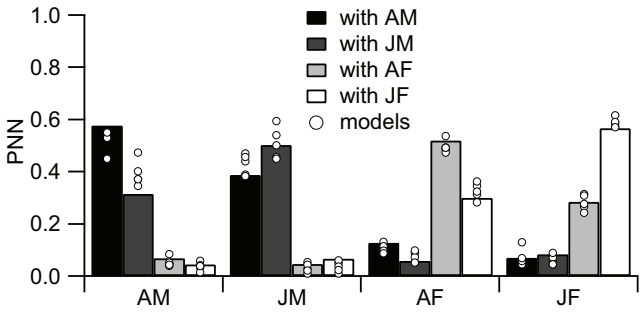


Figure 3: Association pattern from real birds used to evolve models (bars); association patterns from each of five evolved models, averaged over 100 runs (circles).

bution of the study. The basic GA was modified in response to initial tests' indication that constraints may be limiting evolution. The 5 new random models each generation may be too poor to contribute useful variation; thus the random models were mated to the 10 seed models to produce 10 of the 35 offspring, providing a mixture of proven and new elements. Five models were evolved to 49-52 generations. They successfully matched the desired PNN pattern (Fig.3) and averaged fitness 15.6±0.02, comparable to initial tests.

Interpretation of evolved models. As with the initial tests, the models showed neither directional nor stabilizing selection of activity parameters ($|t_{19}| \leq 0.2$, $P \geq 0.8$), but did show selection for on active-bits (χ^2_1=6.5, P=0.01). There was less evidence of selection in the remainder of the classifiers, with only neighbor, sex, and time-bits more often wild in the if-statement ($\chi^2_1 \geq 4.3$, $P \leq 0.04$), and move-bits more often on in the then-statement (χ^2_1=30, P<0.0001). This makes sense, as the behavior evolved for is not so directional nor easily defined as in the initial test. Rather than looking at individual bits across all classifiers, it is more instructive to examine the overall behavior produced.

In general, the evolved rules designated behaviors that increased chances of NN with birds of both ages of the same sex: such behaviors were moving towards or remaining still in response to the other bird (27% of all classifiers; 94% of those applicable to such situations, χ^2_1=24, P<0.0001). Behaviors which decreased chances of NN with opposite sex birds (moving away or randomly) showed a non-significant trend to be more common than behaviors increasing chances (20% all; 67% applicable, χ^2_1=3.6, P=0.06). Behaviors increasing or decreasing chances of NN with both same age and opposite age birds were present in approximately equal numbers ($\chi^2_1 \leq 0.2$, $P > 0.6$). Classifiers which responded to any bird, regardless of age and sex, were rare; most classifiers were specific to at least age or sex (87% of all classifiers, χ^2_1=59, P<0.0001), but rarely to both (27%, χ^2_1=23, P<0.0001). No model ever evolved a class with specific rules in response each age and sex: all classifier sets included generalities. With one exception (JM in Run 1), all models evolved age and sex-based behaviors for all classes. All models evolved some classifiers which went against the overall pattern, for example, in one model juvenile males remained still in response to adults and moved randomly in response to juveniles. Thus, the rules evolved by the classifiers were characterized by a combination of partially specific responses; these tended to increase proximity to the same sex, decrease proximity to the opposite sex, and had mixed responses to different ages.

Evaluation of Evolved Models on New Data

The five evolved models were evaluated for their ability to predict association patterns plus additional behaviors not used for fitness evaluation – approach, response, and flight – in a new group of birds. These birds are the same as the JA condition in White et al. (2002b); I refer the reader to this publication for details of bird capture, housing, etc.

Behavioral data collection

Association patterns. I used near neighbor data collected by Dr Andrew King and Dr David White as part of White et al. (2002b); I calculated PNN for birds within 0.3 m for data collected between 13 Mar - 10 Apr, matching as close as possible the period for which the models were evolved.

Behavior samples. During the same time period, I collected behavioral focal samples: morning (0815-1145 hr) and afternoon (1215-1600 hr) 10 min focal samples on each bird every week for the four weeks, totaling 33.3 hrs observation. Observations were carried out in groups 3-5 times a week, and order of observation balanced across weeks.

During a focal sample, the following behaviors were recorded: approach (moving from a further distance to 0.3 m of another bird), response to approach (leave: moving further than 0.3 m from approaching bird; stay: remaining within 0.3 m), flight (take to the air by flapping wings), and landing environment (near: within 0.3 m of another bird; aware: within 0.9 m of another bird; alone: more than 0.9 m from any bird). Identity of other birds involved in these behaviors was noted. Note that these are not mutually exclusive behaviors: e.g., flights can also be approaches.

Comparison to models.

Data from models. Each evolved model was set up with 5 AM, 8 JM, 7 AF, and 5 JF, to match the make-up of the new group of birds. The same data as above was collected from 100 runs of each model, with the following conversions for the simulated world: near and aware distances were 5 and 15 units respectively; when a movement resulted in being within 5 units of another bird which had also moved that time step, it was only scored as approach if the movement was orientated <90° either side of the other bird's original

location (i.e., an approach cannot be scored if moving away); and all movements were regarded as flights.

Comparison values. From each data set (real and models), 60 comparison values were calculated: 16 PNN among all age and sex classes; 16 proportion approach (PAP) values, calculated for all age and sex classes in the same manner as PNN, except using counts of approach (AP_{ij}) of bird i to j rather than near neighbor (NN_{ij}) observations; 16 proportion leave (PLE) values, calculated from counts of bird i responding with a leave (LE_{ij}) to bird j's approach (AP_{ji}):

$$\mathrm{LE}_{iJ} = \frac{\sum_{j \in J} \mathrm{LE}_{ij}}{\sum_{j \in J} \mathrm{AP}_{ji}}$$

$$\mathrm{PLE}_{IJ} = \frac{\sum_{i \in I} \mathrm{LE}_{iJ}}{|I|}$$

and 12 proportion flight landing (PFL) values calculated from counts of bird i's flights (F_i) and landing environments E (L_{iE}):

$$\mathrm{PFL}_{IE} = \frac{\sum_{i \in I} (\mathrm{L}_{iE} / \mathrm{F}_i)}{|I|}$$

Evaluation of model fit. Comparison values V_R from the real birds were compared to the distribution of 100 values V_M from each evolved model, to determine the fit of the model: if a two-tailed probability of the real value being drawn from that distribution (calculated directly as twice the proportion of V_M as extreme or more so than V_R) was less than α=0.05, the model was considered to not be a good approximation for that value.

The models showed themselves to be good fits to PNN, PAP, and PLE values, but not PFL values (Fig.4). For the first three value types, the model distributions were generally centered around corresponding V_R values, while for PFL the V_R values were in the tail of the distributions (Fig.4a,b). Correspondingly, over half of real PFL values were significantly different from the model, while very few of the other value types were (Fig.4c).

Discussion

An agent-based model of bird behavior was created and successfully evolved using a genetic algorithm to produce desired emergent properties of the system: association patterns of a group of birds. An initial test verified performance of the GA, which was subsequently used to evolve models to match association patterns measured from real birds. The evolved models matched not only association patterns from the situation for which they were evolved, but association patterns collected from a new set of birds. Additionally, they matched patterns of approach and response of the new birds, behaviors for which they were not directly evolved.

The initial test evolved for total assortment showed that the classifier system-based model was an appropriate choice

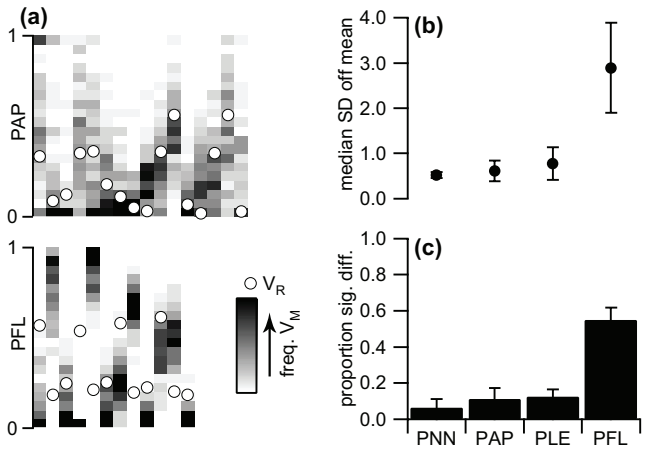


Figure 4: Fit of models to real data. **(a)** Example match of one evolved model's V_M distributions to real values for well matching (PAP) and poor matching (PFL) values. **(b)** Location of real values relative to model distribution shown by median number of standard deviations between V_R and mean V_M. **(c)** Proportion of V_R significantly different from model distribution. For **b**,**c**: mean across 5 evolved models shown; error bars represent standard error of the mean.

for modeling behavioral rules. The evolved rules were straight-forward and easy to interpret. The activity probability chromosomes appeared to have little impact, with the classifiers themselves controlling activity state; overall, it was more beneficial to be active, and thus able to respond to the environment. The classifiers evolved for directed movement towards other birds. It is possible that moving toward was favored over moving away as the movement distance was somewhat less than the near distance: moving away may not avoid an NN point next time step, whereas moving towards always maintained one. Classifiers made strong use of wildcards in the if-statement, allowing behavioral rules to apply to multiple conditions. When not wild, old conditions were more often coded for: this is sensible, as a condition still there from the previous time step may be more likely to remain, and thus be more useful to respond to. Any reason for the bias towards responding to juveniles is unclear; as this comparison was considerably less strong than all others, it is likely that it represents random variation. It is easy to see how rules coding for moving in response to the environment, particularly reoccurring conditions, and in a directed manner could create a strong pattern of association with similar birds. Also, when evolving for total association with only one class, use of wildcards enables birds to more efficiently respond to multiple conditions similarly.

There was no evidence of evolution of multiple strategies for achieving total assortment; instead, it appears evolutionary constraints were responsible for different responses to a bird's own class, in particular, never evolving applicable

classifiers. Thus, when evolving to match patterns of real birds, more variation was introduced by mating randomly generated models with the fittest ones.

The models evolved to real bird behavior successfully matched the association patterns for which they were evolved; this indicates these patterns do not require individual-based responses and can be produced using only age and sex-based rules. The models also matched association patterns of an entirely different group of birds, having a different age and sex distribution; this indicates the evolved association pattern is not tied to age and sex distribution, but is due entirely to the behavioral rules. Thus, these rules are generalizable beyond a particular model to other groups of birds, and so can provide insight into general principles underlying production of self-organized association.

By examining the five evolved models, we can postulate the following about behavioral rules of real birds. First, being able to become active in relation to the environment is key to production of self-organized behavior. Second, instead of requiring specific responses to all age and sex classes, general rules applicable across broad classes of birds (e.g., avoid males, be attracted to juveniles) can lead to assortment. Third, it is likely that all age and sex classes are actively involved in assortment: this is not only supported by repeated emergence in evolved models, but also by match of the models with other movements of real birds. Finally, existence of rules counter to the general pattern indicates that such rules can co-exist with assortment. This is particularly encouraging, as it is known that interaction of juvenile males with adult females–complete opposite classes–is highly important to social development of the juveniles (Smith et al., 2000; Smith, 2001; Smith et al., 2002). The models show that self-organized patterns of association can be maintained even with attraction between different classes of birds.

The above inferences could be tested by future behavioral or computer experiments. For example, behavioral choice tests could characterize attraction/avoidance to age and sex classes; observation could track birds' activity-states; simulations could be designed to match measured activity proportions but disallow response to environment, or only allow 1-2 classes to behave preferentially, and these simulations' effectiveness compared to the current model.

The evolved models were not only successful at predicting association patterns of a new group of birds; they also predicted other behaviors not included in the fitness function. In particular, the models were good fits for observed patterns of approaching other birds and the response to such approaches. These behaviors are key contributors to the self-organized association pattern; attraction and repulsion between birds is what drives with whom they associate. Thus, in evolving models to match a particular association pattern, I succeeded in creating models which match the mechanism producing these patterns. This success indicates the appropriateness of the formulation of the behavioral model and thus provides confidence for the generalizations to real birds.

Another type of behavior not included in the fitness function, proportion of flights to different proximities of other birds, was not well matched by the models. However, these flight behaviors are not contributors to association patterns: birds can land near other birds often or rarely, and maintain the same association pattern, as long as they approach particular classes in the same proportion. Thus, this behavior was independent of the criteria used for evolving the models. Additionally, the formulation of the model did not well reflect the range of potential flight behaviors: the model birds always moved a fixed distance, while real birds make flights of varying distances. This difference did not strongly impact association patterns: a flight of long distance when approaching or leaving could be approximated by several successive flights. However, this difference made certain types of flights, particularly to alone, less likely in the model.

That the models were successful at predicting association patterns for which they were evolved, as well as contributing behaviors, but not behaviors which could vary independently of association pattern, highlights both the limitations and strengths of this modeling approach. The models were designed with the intent of understanding rules underlying self-organized assortment. As such, they included parameters relating intuitively to such rules: behavior in response to neighboring birds. They were not designed to model patterns of flight behavior, and included no parameters for adjusting flight distance. When evolved to match association patterns, there was no reason for the independently varying flight behavior to have an impact, and the models did not accurately reflect these behaviors. However, the models were highly successful at the task for which they were designed: matching association patterns, predicting the behaviors which create these patterns, and thus providing insight into the mechanism behind self-organized social association.

The model is clearly a simplification of bird behavior, even in behaviors related to association patterns. Real birds exhibit behaviors related not only to age and sex of other birds, but to individuals; for example, birds exhibit consistent individual differences in their association behavior (Smith, 2001; Smith et al., 2002), and even "friendships" (spending large amounts of time with a specific individual). This model does not capture such individualized behaviors. Modeling individual differences, and allowing individual-specific behaviors, is an area of future investigation. It would be interesting to see if consistent patterns of individual difference emerged across evolved models, and thus perhaps be relevant to self-organization of association patterns.

Another potential area of future investigation corresponds well with the modeling framework used, classifier systems. Implementing learning of classifiers on an individual-agent basis could allow investigation of learning trajectories. For example, female feedback on singing behavior of males could be modeled, either on top of or distinct from the cur-

rent assortment model. Such a model could address whether the known JM-AF interactions are driven by males' learning or also require female attraction.

This work has shown that it is possible to use agent-based models as a tool to help understand animal behavior. With a question, what rules do birds use to produce their self-organized association pattern, and a model designed to provide intuitive answers, I was able to make inferences about the behavior of real birds. The use of a heuristic search method, the GA, moved the modeling exercise from one of validating a single hypothesized set of rules to one where I probed general principles of behavior. The five evolved models all had a different rules, and all successfully matched bird behavior, showing that there are many possibilities for the rule set. However, while it is not possible to say precisely what behavioral rules the birds use, the five evolved models provide independently generated hypotheses about such rules; thus, similarities across them point towards general principles that are likely to be true of any set of behaviors able to produce the desired association patterns, including those used by real birds. In sum, combining evolutionary algorithms with agent-based modeling can become a valuable tool, enabling study of animal behavior within the complex environments in which it naturally occurs.

Acknowledgements

I thank Drs. King and White for use of their data. This work was supported by an HHMI Predoctoral Fellowship to VAS. All birds were collected under permit No. PRT 767881; all protocols were approved by the Indiana University Care and Use Committee (96-179).

References

Booker, L., Goldberg, D., and Holland, J. (1989). Classifier systems and genetic algorithms. *Artif Intell*, 40:235–282.

Bryson, J. J., Ando, Y., and Lehmann, H. (2007). Agent-based modelling as scientific method: a case study analysing primate social behaviour. *Philos Trans R Soc Lond B Biol Sci*, 362:1685–1698.

Freeberg, T. M., King, A. P., and West, M. J. (1995). Social malleability in cowbirds (*Molothrus ater artemisiae*): species and mate recognition in the first 2 years of life. *J Comp Psy*, 109:357–367.

Friedmann, H. (1929). *The Cowbirds*. CC Thomas, Springfield, IL.

Holland, J. H., Holyoak, K. J., Nisbett, R. E., and Thagard, P. R. (1986). *Induction: Processes of Inference, Learning, and Discovery*. MIT Press, Cambridge, MA.

Jackson, D., Holcombe, M., and Ratnieks, F. (2004). Coupled computational simulation and empirical research into the foraging system of pharaoh's ant (*Monomorium pharaonis*). *Biosystems*, 76:101–112.

King, A. P. and West, M. J. (1988). Searching for the functional origins of cowbird song in eastern brown-headed cowbirds (*Molothrus ater ater*). *Anim Beh*, 36:1575–1588.

King, A. P. and West, M. J. (1989). Presense of female cowbirds (*Molothrus ater ater*) affects vocal imitation and improvisation in males. *J Comp Psy*, 103:39–44.

King, A. P., West, M. J., Eastzer, D. H., and Staddon, J. E. R. (1981). An experimental investigation of the bioacoustics of cowbird song. *Beh Ec Sociobio*, 9:211–217.

King, A. P., West, M. J., and White, D. J. (2002). The presumption of sociality: social learning in diverse contexts in brown-headed cowbirds (*Molothrus ater*). *J Comp Psy*, 116:173–181.

Mayr, E. (1974). Behavior programs and evolutionary strategies. *Am Sci*, 62:650–659.

Powell, L. A., Conroy, M. J., Krementz, D. G., and Lang, J. D. (1999). A model to predict breeding-season productivity for multibrooded songbirds. *Auk*, 116:1001–1008.

Schank, J. C. (2001). Beyond reductionism: Refocusing on the individual with individual-based modeling. *Complexity*, 6:33–40.

Schank, J. C. and Alberts, J. R. (1997). Self-organized huddles of rat pups modeled by simple rules of individual behavior. *J Theor Biol*, 189:11–25.

Schank, J. C. and Alberts, J. R. (2000). The developmental emergence of coupled activity as cooperative aggregation in rat pups. *Proc R Soc Lond B*, 267:2307–2315.

Sellers, W. I., Hill, R. A., and Logan, B. S. (2007). An agent-based model of group decision making in baboons. *Philos Trans R Soc Lond B Biol Sci*, 362:1699–1710.

Smith, V. A., , King, A. P., and West, M. J. (2002). The context of social learning: association patterns in a captive flock of brown-headed cowbirds. *Anim Beh*, 68:23–35.

Smith, V. A. (2001). *Social behavior in brown-headed cowbirds: emergence in a complex system*. PhD thesis, Indiana University.

Smith, V. A., King, A. P., and West, M. J. (2000). A role of her own: female cowbirds, *Molothrus ater*, influence the development and outcome of song learning. *Anim Beh*, 60:599–609.

West, M. J., White, D. J., and King, A. P. (2002). Female brown-headed cowbirds', *Molothrus ater*, organization and behaviour reflects male social dynamics. *Anim Beh*, 64:377–385.

White, D. J., King, A. P., Cole, A., and West, M. J. (2002a). Opening the social gateway: early vocal and social sensitivities in brown-headed cowbirds (*Molothrus ater*). *Ethology*, 108:23–37.

White, D. J., King, A. P., and West, M. J. (2002b). Facultative development of courtship and communication in juvenile male cowbirds (*Molothrus ater*). *Behavioral Ecology*, 13:487–496.

White, D. J., King, A. P., and West, M. J. (2002c). Plasticity in adult development: experience with young males enhances mating competence in adult male cowbirds, *Molothrus ater*. *Behaviour*, 139:713–728.

Evolutionary Advantages of Neuromodulated Plasticity in Dynamic, Reward-based Scenarios

Andrea Soltoggio[1], John A. Bullinaria[2], Claudio Mattiussi[3], Peter Dürr[4], and Dario Floreano[5]

[1,2]School of Computer Science, University of Birmingham, Birmingham B15 2TT, UK
{a.soltoggio,j.a.bullinaria}@cs.bham.ac.uk
[3,4,5] Laboratory of Intelligent Systems, EPFL, Lausanne, CH
{claudio.mattiussi,peter.duerr,dario.floreano}@epfl.ch

Abstract

Neuromodulation is considered a key factor for learning and memory in biological neural networks. Similarly, artificial neural networks could benefit from modulatory dynamics when facing certain types of learning problem. Here we test this hypothesis by introducing modulatory neurons to enhance or dampen neural plasticity at target neural nodes. Simulated evolution is employed to design neural control networks for T-maze learning problems, using both standard and modulatory neurons. The results show that experiments where modulatory neurons are enabled achieve better learning in comparison to those where modulatory neurons are disabled. We conclude that modulatory neurons evolve autonomously in the proposed learning tasks, allowing for increased learning and memory capabilities.

Introduction

The importance of modulatory dynamics in neural substrates has been increasingly recognised in recent years. The notion that neural information processing was fundamentally driven by the electrical synapse has been replaced by the more accurate view that modulatory chemicals play a relevant computational role in neural functions (Abbott and Regehr, 2004). Experimental studies on both invertebrates and vertebrates (Burrell and Sahley, 2001; Birmingham and Tauck, 2003) suggest that neuromodulators such as Acetylcholine (ACh), Norepinephrine (NE), Serotonin (5-HT) and Dopamine (DA) closely affect synaptic plasticity, neural wiring and the mechanisms of Long Term Potentiation (LTP) and Long Term Depression (LTD). These phenomena are deemed to affect both short and long term configuration of brain structures, and therefore have been linked to the formation of memory, brain functionalities and considered fundamental in learning and adaptation (Jay, 2003).

The realisation that the Hebb's synapse (Cooper, 2005) does not account entirely for experimental evidence on synaptic modification has brought growing focus on modulatory dynamics. Associative learning as classical and operant conditioning, and various forms of long-term wiring and synaptic changes seem to be based on additional mechanisms besides the Hebbian synapse. Studies on mollusks like the *Aplysia californica* (Roberts and Glanzman, 2003) have shown neuromodulation to regulate classical conditioning (Carew et al., 1981; Sun and Schacher, 1998), operant conditioning (Brembs et al., 2002) and wiring in developmental processes (Marcus and Carew, 1998).

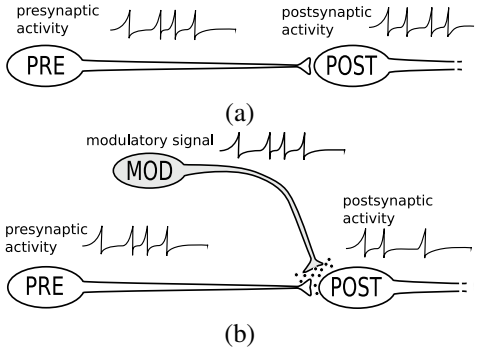


Figure 1: (a) Hebbian plasticity: the connection strength is updated as function of pre- and postsynaptic activity only. (b) Heterosynaptic mechanism, or neuromodulation: the connection growth is mediated by neuromodulation, i.e. the amount of modulatory signal determines the response to Hebbian plasticity. The dots surrounding the synapse represent the concentration of modulatory chemicals released by the modulatory neuron.

Classical Hebbian plasticity refers to synapse modification based on pre- and postsynaptic activities. A presynaptic and a postsynaptic neuron are involved in the process. Neuromodulation, on the other hand, involves a third modulatory neuron that diffuses chemicals at target synapses as illustrated in Figure 1. A unique working mechanism for neuromodulation has not been identified due the large variety of modulatory dynamics involving different chemicals, stimuli, brain areas and functions. However, Bailey et al. (2000a) suggest that heterosynaptic modulation is essential for stabilising Hebbian plasticity and memory. That review paper outlines the nonlinear effect of modulatory signals; when neuromodulation is coupled with presynaptic stimuli, it results in the activation of transcription factors and pro-

tein synthesis during synaptic growth. This in turn leads to durable and more stable synaptic configuration (Bailey et al., 2000b). The underlying idea is that the synaptic growth that occurs in the presence of modulatory chemicals is long lasting, i.e. has a substantially longer decay time than the same growth in absence of modulation.

At a system level, the release of modulatory chemicals has been linked to learning. In (Schultz et al., 1993), dopamine activation patterns recorded in monkeys' brains followed a measure of prediction-error during learning tasks in classical conditioning. Following studies have linked modulatory activity with learning, reward and motivation (Schultz et al., 1997). How cellular mechanisms of synaptic growth and global patterns of neural activation relate has not been unveiled yet, however, growing evidence indicates a direct link between cellular and system level.

Advances in biology have resulted in the formulation of computational models (Fellous and Linster, 1998; Doya, 2002), which try to capture the computational role and significance of neuromodulation. Artificial Neural Networks (ANNs) have also been extended to include forms of neuromodulation. Short term memory by means of neuromodulation was investigated in (Ziemke and Thieme, 2002) where a robot navigated in a T-maze and remembered turning directions according to visual clues in the maze. Improved evolvability in neural controller was shown with the use of GasNet (Smith et al., 2002), although these networks have modulated output functions rather than synaptic plasticity. Learning and adaptivity were shown in navigation tasks in (Sporns and Alexander, 2002) where a neural architecture was manually designed to update weights according to reinforcement signals. Improved performance and adaptation by means of neuromodulation were shown on a real robot in (Kondo, 2007).

Because synaptic plasticity is often considered a way to achieve adaptation and learning, many benchmark problems for neuromodulation are based on uncertain environments. A single modulatory neuron was used to evolve learning behaviour for a simulated foraging task in uncertain environments (Niv et al., 2002). In that study, a simulated flying bee was capable of choosing the higher rewarding flower in a flower-field with changing reward conditions. This experimental setting was chosen also by Soltoggio et al. (2007) to show that modulatory architectures could freely develop throughout evolution to achieve higher performance than in (Niv et al., 2002). These previous two studies (Niv et al., 2002; Soltoggio et al., 2007) support the idea that neuromodulation plays a central role in regulating plasticity when variable environmental conditions require a change in policies of control. However, despite the recent computational models, studies on the precise computational advantages of neuromodulation are very limited. In addition, there are few working models of learning in networks.

This work addresses the issue by analysing the spontaneous evolution of neuromodulation in T-maze navigation tasks, and assesses the advantage of modulatory over traditional networks when dealing with learning problems. In these environments, an agent navigates a T-maze to discover the location of a reward. The location of the reward is not kept fixed, but changes during the agent's lifetime, fostering the development of adaptive and learning behaviour. A comparison of modulatory and non-modulatory networks is presented, where the results suggest an evolutionary advantage in the use of neuromodulation.

The next section describes the computational model of modulatory neurons. Following, the T-maze learning problems are presented before illustrating the evolutionary algorithms by means of which networks are evolved. The Results section presents experimental results and discussion. The paper ends with final remarks in the Conclusion.

Modulatory Neurons

In Artificial Neural Networks (ANNs) with only one type of neuron, each node exerts the same type of action on all the other nodes to which it is connected. Typically this consists in the propagation of activation values throughout the network. However, given the variety of neurons and chemicals in the brain, it is conceivable to extend ANNs by devising different types of neurons. Here, we introduce a special kind of neuron that we define *modulatory neurons*: accordingly, nodes in the network can be either *modulatory* or *standard* neurons (Soltoggio et al., 2007). In doing so, the rules of interactions among neurons of different kinds need to be devised. Assuming that each neuron can receive inputs from neurons of both types, each node in the network will store the intensity of inputs deriving from each sub-system, i.e. from the sets of neurons belonging to different kinds. This principle is comparable to the presence of many kinds of receptors in biological neurons.

Because two types of neurons are considered here, *standard* and *modulatory*, each neuron i regardless of its type has an internal value for a *standard activation* a_i and a value for a *modulatory activation* m_i. The two activations are computed by summing the inputs from the two subsets of neurons in the network

$$a_i = \sum_{j \in Std} w_{ji} \cdot o_j \quad , \tag{1}$$

$$m_i = \sum_{j \in Mod} w_{ji} \cdot o_j \quad , \tag{2}$$

where w_{ji} is the connection strength from neuron j to i, o_j is the output of a presynatic neuron computed as function of the standard activation $o_j(a_j) = tanh(a_j/2)$.

The novel aspect in the model is the modulatory activation that determines the level of plasticity for the incoming connections from standard neurons. Given a neuron i, the

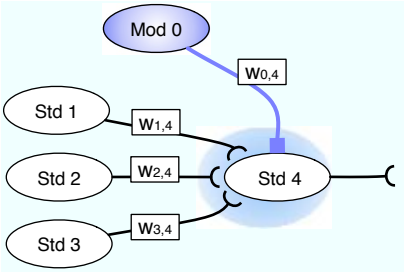


Figure 2: Ovals represent standard and modulatory neurons labeled with *Std* and *Mod*. A modulatory neuron transmits a modulatory signal – represented as a coloured shade – that diffuses around the incoming synapses of the target neuron. Modulation affects the learning rate for synaptic plasticity on the weights $w_{1,4}$, $w_{2,4}$ and $w_{3,4}$ that connect to the neuron being modulated.

incoming connections w_{ji}, with $j \in Std$, undergo synaptic plasticity according to the equation

$$\Delta w_{ji} = tanh(m_i/2) \cdot \delta_{ji} \quad (3)$$

where δ_{ji} is a *plasticity term*. A graphical interpretation is shown in Figure 2. The idea in Equation 3 is to model neuromodulation with a multiplication factor on the plasticity δ of individual neurons being targeted by modulatory neurons. A modulation of zero will result in no weight update, maintaining the weights at the current value; higher levels of modulation will result in a weight change proportional to the modulatory activity times the plasticity term.

In this work, the synaptic plasticity is described by the rule

$$\delta_{ji} = \eta \cdot [Ao_j o_i + Bo_j + Co_i + D] \quad (4)$$

where o_j and o_i are the pre- and postsynaptic neuron outputs, η is the learning rate, and A, B, C, and D are tunable parameters. Equation 4 has been used in previous studies of neuromodulation (Niv et al., 2002; Soltoggio et al., 2007). Its generality is given by the presence of a correlation term A, a presynaptic term B, a postsynaptic term C and a constant D. D allows for strict heterosynaptic update, meaning synaptic update in absence of pre- or postsynaptic activity. The use and tuning of one or more of these terms allow for the implementation of a large variety of learning rules. The modulatory operation of Equation 3 can be applied to any kind of plasticity rule δ and neural model, e.g. Hebbian correlation rules with discrete time dynamics, spiking neural networks, or other. From this view, the idea of modulating, or gating, plasticity is independent of the specific neural model chosen for implementation: its role consists in the activation of local plasticity upon transmission of modulatory signals to specific neurons.

When applied to a suitable neural architecture, this form of gated plasticity can selectively activate learning in specific parts of the network and at the onset of specific events.

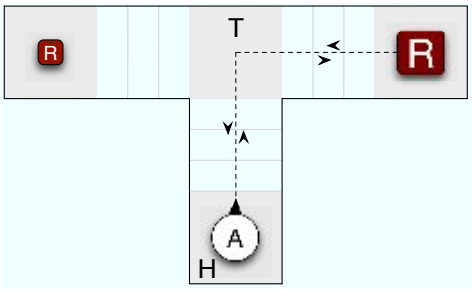


Figure 3: T-maze with homing. The agent navigates the maze returning home (H) after collecting the reward. The amount of reward is proportional to the size of the token.

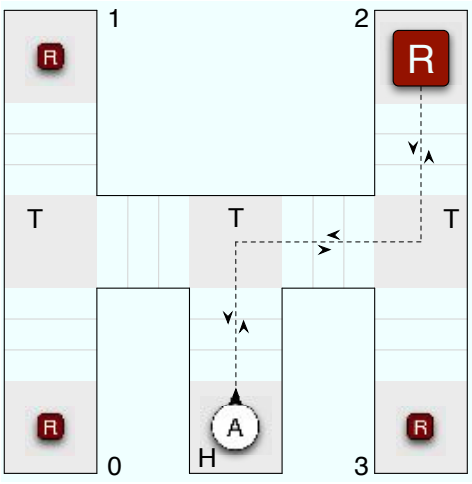


Figure 4: Double T-maze with homing.

This may prevent catastrophic forgetting that often results from continuously updating networks and lead to more efficient learning.

Learning in the T-maze

T-mazes are often used to observe operant conditioning (Britannica, 2007) in animals that are required to learn and remember – for instance – whether a reward in the form of food is located either on the right or on the left of a T-maze. This makes an ideal scenario for testing the effect of neuromodulation.

We simulated two T-mazes represented in Figures 3 and 4. In the first case (Figure 3), an agent is located at the bottom of a T-maze. At the end of two arms (left and right) there is either a high or a low reward. The task of the agent is to navigate the corridors, turn when it is required, collect the reward and return home. This is repeated many times during a lifetime: we call each trip to a maze-end a *trial*. A measure of quality in the agent's strategy is based on the total amount of reward collected. To maximise this measure, the agent needs to learn where the high reward is located.

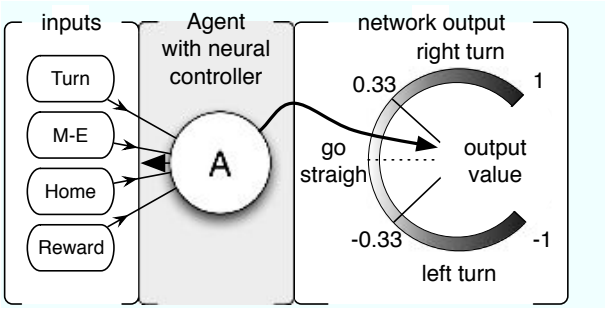


Figure 5: Inputs and output the neural network. The *Turn* input is 1 when a turning point is encountered. M-E is Maze-End: it goes to 1 at the end of the maze. *Home* becomes 1 at the home location. The *Reward* input returns the amount of reward collected at the maze-end, it remains 0 during navigation. One output determines the actions of turning left (if less than -1/3), right (if greater than 1/3) or straight navigation otherwise. Inputs and internal neural transmission are affected by 1% noise.

The difficulty of the problem lies in the fact that the position of the reward changes across trials. When this happens, the agent has to forget the position of the reward that was learnt previously and explore the maze again. In our experiments, the position of the high reward is changed at least once during lifetime, resulting in an uncertain foraging environment where the pairing of actions and reward is not fixed: turning left might result in a high reward at a certain time but in a lower reward later on. The intent is to foster the emergence of learning behaviour.

The complexity of the problem can be increased, as shown in Figure 4, by enlarging the maze to include two sequential turning points and four possible endings. In this problem an optimal strategy is achieved when the agent explores sequentially the four possible maze-ends until the high reward is found. At this point, the sequence of turning actions that leads there should be learnt and memorised together with the return sequence to the home location.

An agent is exposed to 100 trials in the experiments with the single T-maze and to 200 trials in the double T-maze. Each trial consists of a number of steps during which the neural network is updated and the agent moved accordingly (Figure 5). The large reward is randomly positioned and relocated after 50 trials on average, with a random variability of ± 15. The high reward value is 1.0 whereas the low reward is 0.2. The agent that fails to return to the home position (within a trial) will be relocated automatically to the home position and will suffer a penalty of 0.3, which is subtracted from the total amount of reward collected. The agent is required to maintain a forward direction in corridors and perform a right or left turn at the turning points: failure to do so results in the agent crashing, a penalty of 0.4 and being relocated to the home position. Each corridor and turning point stretches for three steps of the agent. Higher or variable numbers of steps have been tested providing similar results.

The control systems of the agents are evolved using the agents' performance as a measure of fitness.

Evolutionary Search

An Evolution Strategy (ES) (Bäck et al., 1997) was used to search for network topologies. The genome was encoded as a matrix of real-valued weights that represent the strengths of the initial connections w_{ij}. The 5 parameters for the plasticity rule A, B, C, D and η of Equation 4 were separately encoded and evolved in the range [-1,1] for A-D, and [-100,100] for η. A set of special genetic operators was devised to perform the topology search: insertion, duplication and deletion of neurons were introduced respectively to insert a new neuron in the network (a new line and row are added to the weight matrix) with probability 0.04, to duplicate an existing neuron (a line and a row are duplicated in the weight matrix) with probability 0.02, and delete a neuron (a line and a row are deleted in the weight matrix) with probability 0.06. Inserted neurons have the same probability (0.5) of being standard or modulatory.

All real values in the genome (GeV_i) are in the range [-1,1], and the phenotypical values PhV_i (with the exception of η), are mapped as $PhV_i = R_i \cdot (GeV_i)^3$, where R is the range (10 for weights, 1 for A..D). The mapping with a cubic function was introduced to favor small weights and parameter initially, and allow for the evolutionary growth of larger values by selection pressure when those are needed. Weights below 0.1 were set to 0.

Mutation is applied to all individuals (except the best) at each generation by adding to each gene a positive or negative perturbation $d = W \cdot exp(-P \cdot u)$, where u is a random number drawn from a uniform distribution [0,1] and P is a precision parameter here set to 180. This probability distribution favours local search with occasional large jumps, see (Rowe and Hidovic, 2004) for details. The function, although differently shaped than the traditional Gaussian, does not introduce a conceptual difference in the evolutionary algorithm. One point crossover on the weight matrix was applied with probability 0.1. A selection mechanism to enhance diversity in the population was devised. All individuals were positioned sequentially on an array. At each generation, the array was divided into consecutive segments of size 5 (with random segmentation offset at each generation), and the best individual of each segment was copied over the neighboring four. In this way, a successful individual spreads its genes only linearly with the generations. A population size of 300 for the single T-maze and 1000 for the double T-maze were used with termination criterion of 600 and 1000 generations. Generation zero was initialised with networks with one neuron per type and random connections.

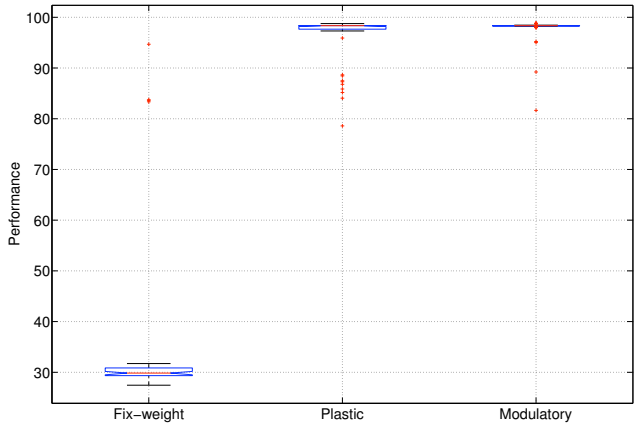


Figure 6: Box plots with performances of 50 runs on the single T-maze. The boxes are delimited by the first and third quartile, the line inside the boxes is the median value while the whiskers are the most extreme data samples from the box not exceeding 1.5 times the interquartile interval. Values outside this range are outliers and are marked with a cross. Boxes with non overlapping notches have significantly different median (95% confidence) (Matlab, 2007)

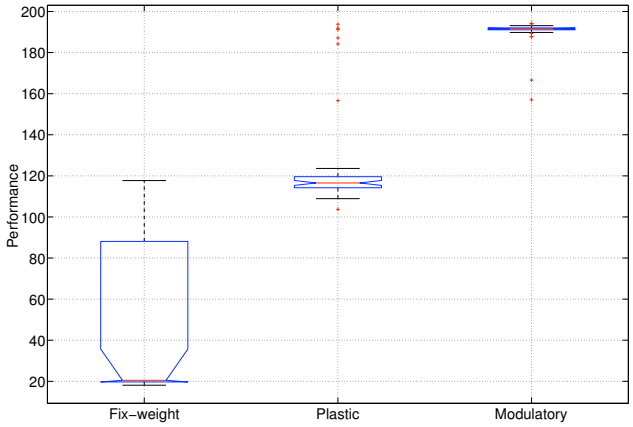


Figure 7: Box plots with performances of runs on the double T-maze.

Experimental Results

We conducted three types of evolutionary experiments, each characterised by different constraints on the properties of the neural networks: 1) fixed-weight, 2) plastic, and 3) plastic with neuromodulation. The fixed-weight networks were implemented imposing a value of zero on the modulatory activity, which resulted in a null update of weights (Equation 3). Plastic networks had a fixed modulatory activity of 1 so that all synapses are continuously updated (Equation 3 becomes $\Delta w = 0.462 \cdot \delta$). Finally, neuromodulatory plastic networks could take advantage of the full model described in Equations 1-4.

Fifty independent runs were executed for each of the three conditions. For each run, the individual that performed best at the last generation was tested 100 lifetimes with different initial conditions. The average reward collected over the 100 tests is the numerical value of the performance. The procedure was repeated for all the 50 independent runs. The distribution of performance is summarized by box plots in Figure 6 for the single T-maze, and in Figure 7 for the double T-maze.

For the single T-maze, the theoretical and measured maximum amount of reward that can be collected on average is 98.8, and not 100 due to the minimum amount of exploration that the agent needs to perform at the beginning of its lifetime and when the reward changes position. For the double T-maze, the theoretical and measured maximum amount of reward that can be collected is 195.2 when averaged on many experiments.

The experimental results indicate that plastic networks achieve far better performance than the fixed-weight networks. Fixed-weight networks could potentially display levels of learning-like behaviour by exploiting recurrent connections and storing state-values in the activation of neurons (Blynel and Floreano, 2002). However, our experiments show that such solutions are more difficult to evolve.

Among plastic networks, those that could exploit modulation displayed only a small advantage in the single T-maze. However, when memory and learning requirements increase for the double T-maze, modulated plasticity displayed a considerable advantage. Figure 7 shows that modulatory networks achieved nearly optimal performance in the double T-maze experiment. Simplified versions of the single and double T-maze can be obtained by removing the requirement for homing. Experiments not reported here on T-mazes without homing confirmed the results showing an advantage for modulatory networks.

It is important to note that the exact performance reported in Figures 6 and 7 depend on the specific design and settings of the evolutionary search. Higher or lower population numbers, available generations, different selection mechanisms and mutation rates affect the final fitness achieved in all cases of fix-weight, plastic and modulatory networks. However, a set of preliminary runs performed by varying the above settings confirmed that the differential in performance between modulatory networks and plastic or fix-weight networks is consistent, although not always the same in magnitude.

Analysis and Discussion

The agents achieving optimal fitness in the tests display an optimal control policy of actions. This consists in adopting an exploratory behaviour initially – until the location of the high reward is identified – followed by an exploitative

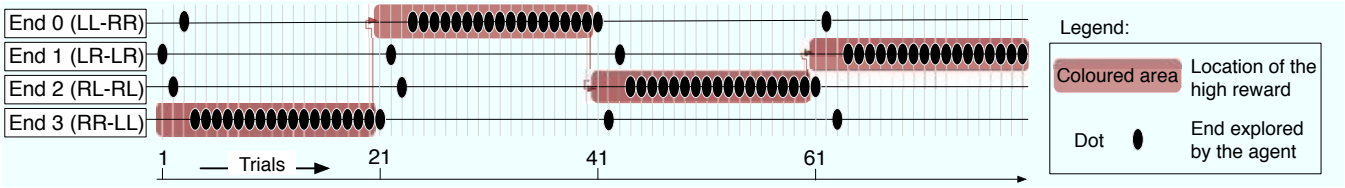


Figure 8: Behaviour of an agent exploring the double T-maze of Figure 4. A test of 80 trials is performed. The four horizontal lines track the events at each of the four maze-ends. The position of the reward is changed every 20 trials. The coloured area indicates where the high reward is located. The black dots show the maze-end explored by the agent at each trial. The agent adopts an explorative behaviour when it does not find the high reward, and settles on an exploitative behaviour after the high reward is found.

behaviour of returning continuously to the location of the high reward. Figure 8 shows an evolved behaviour, which is analogous to operant conditioning in animal learning. This policy involves the exploration of the 4 maze-ends. When the high reward is discovered, the sequence of turning actions that lead there, and the correspondent homing turning actions, are memorised. That sequence is repeated as long as the reward remains in the same location, but abandoned when its position changes. At this point the exploratory behaviour is resumed. This alternation of exploration and exploitation driven by search and discovery of the reward continues indefinitely across trials.

Although this strategy is a mandatory choice to maximise the total reward, from the performance indices presented in the previous section (Figures 6 and 7) we deduce that this behaviour can be more easily evolved when modulatory neurons are allowed into networks.

Functional Role of Neuromodulation

The experimental data on performance showed a clear advantage for networks with modulatory neurons. Yet, the link between performance and characteristics of networks is not easy to find due to the large variety of topologies and learning rules that evolved from independent runs. Figure 9 shows an example of a network that solves the double T-maze. The neural topology, number of neurons and learning rule may vary considerably across evolved networks that perform equally well.

Nonetheless, it is possible to check if the better performance in the double T-maze agents evolved with neuromodulated plasticity is correlated with a differential expression of modulatory and standard neurons. The architecture and composition of the network are modified by genetic operators that insert, duplicate and delete neurons. We measured the average number of the two types of neurons in evolving networks for the condition where plasticity is not affected by modulation (Figure 10, top left graph) and for the condition where plasticity is affected by modulatory inputs (Figure 10, bottom left graph). In both conditions, the number of modulatory neurons is higher than the number of standard neurons. However, the presence of modulatory neurons when those are not active (top left graph) depends only on insertion, duplication and deletion rates, whereas in the case when they are enabled (bottom left graph) their presence might be linked to a functional role. This fact is suggested by the higher value of the mean fitness.

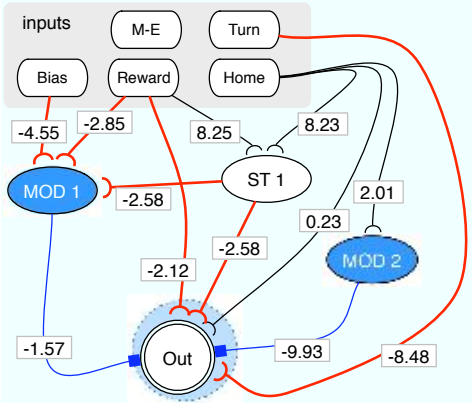


Figure 9: Example of an evolved network that solves the double T-maze. This network has two modulatory neurons and one standard neuron beside the output neuron. Arcs represent synaptic connections. The inputs (Bias, Turn, Home, M-E, Reward) and standard neurons (ST 1 and OUT) send standard excitatory/inhibitory signals to other neurons. Modulatory neurons (MOD 1 and MOD 2) send modulatory signals which affects only plasticity of postsynaptic neurons, but not their activation level. The evolved plasticity rule is $A = 0, B = 0, C = -0.38, D = 0, \eta = -94.6$.

In a second phase, we continued the evolutionary experiments for additional thousand generations, but we set to zero the probability of inserting and duplicating neurons, while the probability of deleting neurons was left unchanged. In both conditions all types of neurons slightly decreased in number. However, modulatory neurons completely disappeared in the condition where the modulatory input had no effect on plasticity (Figure 10, top right graph) while on average two modulatory neurons were observed in the condition where modulation could affect plasticity. This repre-

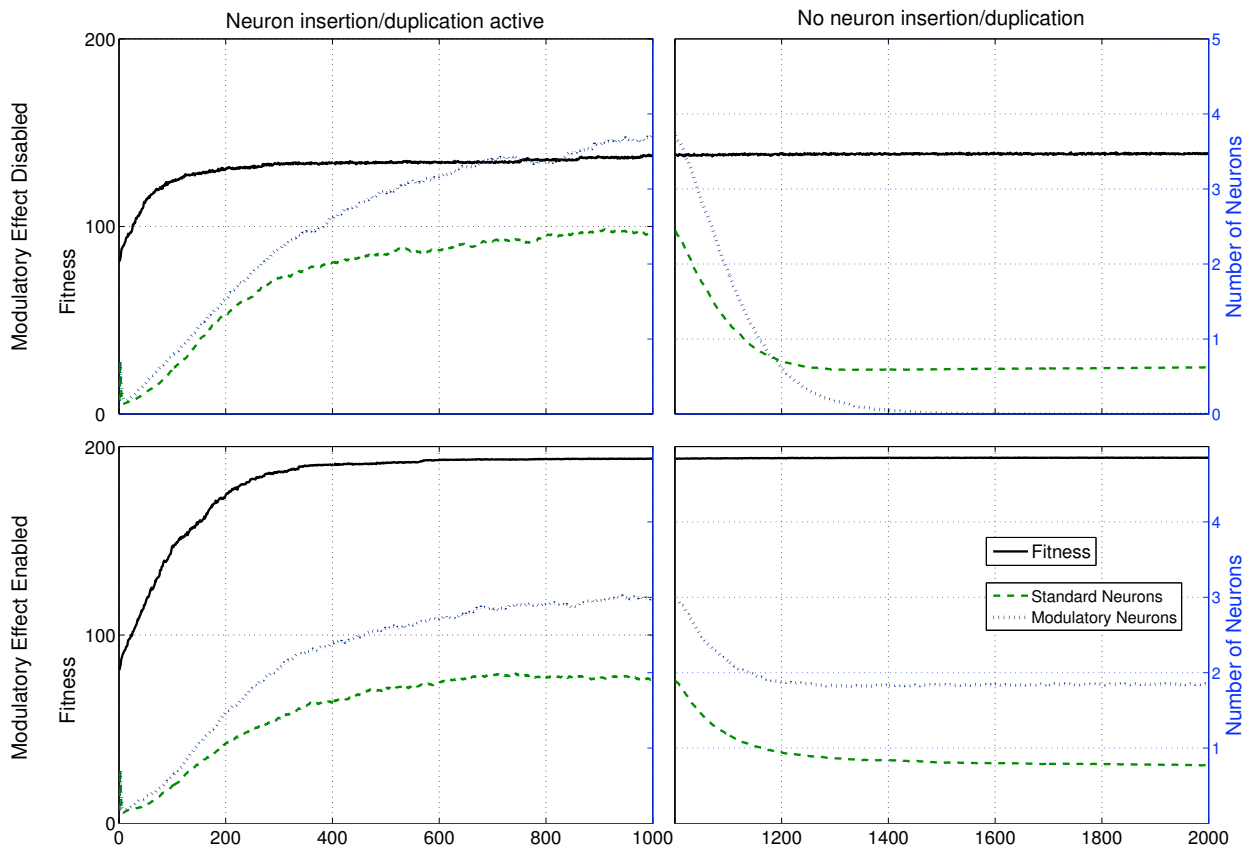


Figure 10: Fitness (continuous line) and number of neurons (dashed lines for standard and dotted lines for modulatory) in networks during evolution (average values of 50 independent runs).

sents a further indication that neuromodulation of synaptic plasticity is responsible for the higher performance of the agents in the double T-maze problem and that they play a functional role in guiding reward-based learning.

A further test was conducted on the evolved modulatory networks when the evolutionary process was completed. Networks with high fitness that evolved modulatory neurons were tested with modulation disabled. The test revealed that modulatory networks, once deprived of modulatory neurons, were still capable of navigation by turning at the required points and maintaining straight navigation along corridors. The low level navigation was preserved and the number of crashes did not increase. However, most of networks seemed capable of turning only in one direction (i.e. always right, or always left), therefore failing to perform homing behaviour. None of the networks appeared to be capable of learning the location of the high reward. Generally, networks that were evolved with modulation and that were downgraded to plastic networks (by disabling modulatory neurons) performed worse than evolved plastic networks. Hence, we can assume that modulatory neurons are not employed to implement a higher level of functionality, otherwise not achievable with simple plasticity. Rather, modulatory neurons are employed to design a completely different neural dynamics that, according to our experiments, are easier to evolve, and on average resulted in better performance at the end of the simulated evolution.

Conclusion

The model of neuromodulation described here applies a multiplicative effect on synaptic plasticity at target neurons, effectively enabling, disabling or modulating plasticity at specific locations and times in the network. The evolution of network architectures and the comparison with networks unable to exploit modulatory effects allowed us to show the advantages brought in by neuromodulation in environments characterised by distant reward and uncertainties. We did not observe an obvious correspondence between performance and architectural motifs: we assume that the unconstrained topology search combined with different evolved plasticity rules allow for a large variety of well performing structures. In this respect, the search space was explicitly unconstrained in order to assess modulatory advantages independently of particular or hand-designed neural structures. In this condition, the philogenetic analysis of evolving networks supports the hypothesis that modulated plasticity

is employed to increase performance in environments where sparse learning events demand memorisation of selected and timed signals.

Future work includes the analysis of working architectures to understand the relation between the requirements of the problems and the type and size of networks that solve them. This study does not address in detail the neural dynamics that allowed for the improved learning and memory capabilities. Further analysis could possibly clarify the properties of the internal neural pattern of activations and weight changes. The relation between reward signals and modulatory activations could unveil important properties of the neural dynamics and explain how information from global reinforcement signals is transferred to the synaptic level, and consequently modify behavioural responses.

Acknowledgements

This work was partly supported by the Swiss National Science Foundation.

References

Abbott, L. F. and Regehr, W. G. (2004). Synaptic computation. *Nature*, 431:796–803.

Bäck, T., Fogel, D. B., and Michalevicz, Z., editors (1997). *Handbook of Evolutionary Computation*. Oxford University Press, Oxford.

Bailey, C. H., Giustetto, M., Huang, Y.-Y., Hawkins, R. D., and Kandel, E. R. (2000a). Is heterosynaptic modulation essential for stabilizing hebbian plasticity and memory? *Nature Reviews Neuroscience*, 1(1):11–20.

Bailey, C. H., Giustetto, M., Zhu, H., Chen, M., and Kandel, E. R. (2000b). A novel function for serotonin-mediated short-term facilitation in aplysia: Conversion of a transient, cell-wide homosynaptic hebbian plasticity into a persistent, protein synthesis-independent synapse-specific enhancement. *PNAS*, 97(21):11581–11586.

Birmingham, J. T. and Tauck, D. L. (2003). Neuromodulation in invertebrate sensory systems: from biophysics to behavior. *The Journal of Experimental Biology*, 20:3541–3546.

Blynel, J. and Floreano, D. (2002). Levels of dynamics and adaptive behavior in evolutionary neural controllers. In *Proceedings of the seventh international conference on simulation of adaptive behavior on From animals to animats*, pages 272–281. MIT Press Cambridge, MA, USA.

Brembs, B., Lorenzetti, F. D., Reyes, F. D., Baxter, D. A., and Byrne, J. H. (2002). Operant Reward Learning in Aplysia: Neuronal Correlates and Mechanisms. *Science*, 296(5573):1706–1709.

Britannica (2007). Animal learning. Encyclopedia Britannica 2007 Ultimate Reference Suite.

Burrell, B. D. and Sahley, C. L. (2001). Learning in simple systems. *Current Opinion in Neurobiology*, 11:757–764.

Carew, T. J., Walters, E. T., and Kandel, E. R. (1981). Classical conditioning in a simple withdrawal reflex in aplysia californica. *The Journal of Neuroscience*, 1(12):1426–1437.

Cooper, S. J. (2005). Donald O. Hebb's synapse and learning rule: a history and commentary. *Neuroscience and Biobehavioral Reviews*, 28(8):851–874.

Doya, K. (2002). Metalearning and neuromodulation. *Neural Networks*, 15(4-6):495–506.

Fellous, J.-M. and Linster, C. (1998). Computational models of neuromodulation. *Neural Computation*, 10:771–805.

Jay, M. T. (2003). Dopamine: a potential substrate for synaptic plasticity and memory mechanisms. *Progress in Neurobiology*, 69(6):375–390.

Kondo, T. (2007). Evolutionary design and behaviour analysis of neuromodulatory neural networks for mobile robots control. *Applied Soft Computing*, 7(1):189–202.

Marcus, E. A. and Carew, T. J. (1998). Developmental emergence of different forms of neuromodulation in Aplysia sensory neurons. *PNAS, Neurobiology*, 95:4726–4731.

Matlab (2007). *Box plots*. The Mathworks Documentation.

Niv, Y., Joel, D., Meilijson, I., and Ruppin, E. (2002). Evolution of Reinforcement Learning in Uncertain Environments: A Simple Explanation for Complex Foraging Behaviours. *Adaptive Behaviour*, 10(1):5–24.

Roberts, A. C. and Glanzman, D. L. (2003). Learning in aplysia: looking at synaptic plasticity from both sides. *Trends in Neuroscience*, 26(12):662–670.

Rowe, J. E. and Hidovic, D. (2004). An evolution strategy using a continuous version of the gray-code neighbourhood distribution. In *GECCO (1)*, pages 725–736.

Schultz, W., Apicella, P., and Ljungberg, T. (1993). Responses of Monkey Dopamine Neurons to Reward and Conditioned Stimuli during Successive Steps of Learning a Delayed Response Task. *The Journal of Neuroscience*, 13:900–913.

Schultz, W., Dayan, P., and Montague, P. R. (1997). A Neural Substrate for Prediction and Reward. *Science*, 275:1593–1598.

Smith, T., Husbands, P., Philippides, A., and O'Shea, M. (2002). Neuronal Plasticity and Temporal Adaptivity: GasNet Robot Control Networks. *Adaptive Behaviour*, 10:161–183.

Soltoggio, A., Dürr, P., Mattiussi, C., and Floreano, D. (2007). Evolving Neuromodulatory Topologies for Reinforcement Learning-like Problems. In *Proceedings of the IEEE Congress on Evolutionary Computation, CEC 2007*.

Sporns, O. and Alexander, W. H. (2002). Neuromodulation and plasticity in an autonomous robot. *Neural Networks*, 15:761–774.

Sun, Z.-Y. and Schacher, S. (1998). Binding of Serotonin to Receptors at Multiple Sites Is Required for Structural Plasticity Accompanying Long-Term Facilitation of Aplysia Sensorimotor Synapses. *The Journal of Neuroscience*, 18(11):3991–4000.

Ziemke, T. and Thieme, M. (2002). Neuromodulation of Reactive Sensorimotor Mappings as Short-Term Memory Mechanism in Delayed Response Tasks. *Adaptive Behavior*, 10:185–199.

Can Body Language Shape Body Image?

Luc Steels[1,2], Michael Spranger[2]

[1]University of Brussels (VUB AI LAB)
[2]Sony Computer Science Laboratory Paris
steels@arti.vub.ac.be

Abstract

One of the central themes in autonomous robot research concerns the question how visual images of body movements by others can be interpreted and related to one's own body movements and to language describing these body movements. The discovery of mirror neurons has shown that there are brain circuits which become active both in the perception and the re-enactment of bodily gestures, although it is so far unclear how these circuits can form, i.e. how neurons become mirror neurons. We report here further progress with our robot experiments in which a group of autonomous robots play language games in order to coordinate their visual, motor and cognitive body image. We have shown that the right kind of semiotic dynamics can lead to the self-organisation of a successful communication system with which robots can ask each other to perform certain actions. The main contribution of this paper is to show that if the robot has the capacity to 'imagine' the behavior of his own body through self-simulation, he is better able to guess what action corresponds to a visual image produced by another robot and thus guess the meaning of an unknown word. This leads to a significant speed-up in the way individual agents are able to coordinate visual categories, motor behaviors and language.

Introduction

Starting with work published in an artificial life context more than a decade ago (Steels, 1995, 2003), we now have quite solid mechanisms that show how a vocabulary may self-organise in a population of embodied agents that lexicalises perceptually grounded categories, such as colors (Steels and Belpaeme, 2005) or spatial relations (Steels and Loetzsch, 2008). In these experiments, the formation of a language is not just an after-thought, something that takes place after solid concepts are formed. Rather, the formation of categories takes place in intimate co-evolution with the formation of language and both mutually influence each other. The question we address here is whether this approach is relevant to understand the formation of a body-image as well. Body image refers to a collection of representations that embodied agents must maintain in order to move about in the world, plan and execute action, perceive and interpret the behaviors of others, build and use an episodic memory, and understand or produce language about action, for example commands. The body image is not static but dynamically changing over time in concordance with own body movements or body movements of others. The subject of body image has received wide attention, particularly in the neurological literature because of puzzling phenomena such as phantom limbs, mirror box experiments, unusual pain, out of body experiences, etc. (Ramachandran and Hirstein, 1998; Rosenfield, 1988; Blanke and Castillo, 2007) and in the neurobiological literature because of the discovery of mirror neurons (Rizzolatti et al., 1996; Rizzolatti and Arbib, 1998). These disorders, experiments and neurobiological observations make it quite clear that body image is not a simple, fixed innately given internal representation but a dense network of representations that forms in development and continues to change and be adapted throughout life. The relation between visual representations and recognition of bodily action on the one hand and the bodily action itself has also been intensely studied in robotics research, particularly in research on imitation (Billard, 2002; Demiris and Johnson, 2003; Mataric, 2002). It is moreover a key topic in 'embodied artificial intelligence' (Pfeifer et al., 2007; Nabeshima et al., 2006) which emphasises the grounding of cognition in bodily activities.

In a recent paper (Steels and Spranger, 2008) we have already reported experiments with QRIO humanoid robots (Gutmann et al., 2005) set up to understand the relation between the visual and motor body image through a particular language game which we call the Action Game. Two agents are randomly chosen from a population and take on the roles of speaker and hearer. They are downloaded in a robot body in order to play a situated embodied language game. The speaker asks the hearer to do a physical action and the game is a success if the hearer indeed performs the requested action as judged by the speaker. If the game fails, the speaker repairs the communication by performing the action himself. Obviously this game can only be played successfully when the agents are able to categorise bodily gestures performed by others based on visual input and relate them to their own motor behaviors that would produce these same gestures. In

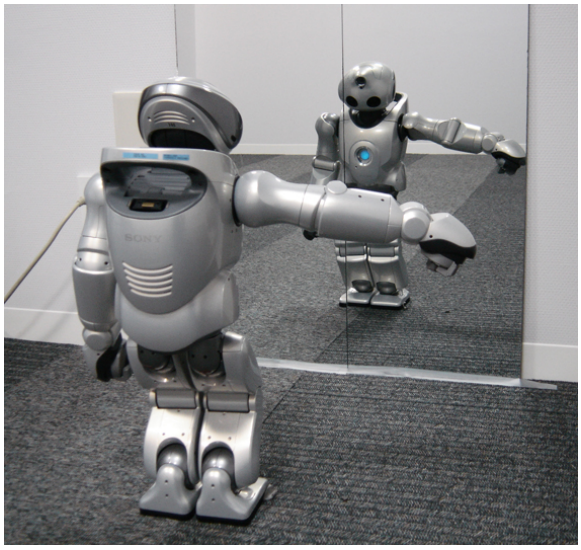


Figure 1: A humanoid robot stands before a mirror and performs various motor behaviors thus observing what visual body-images these behaviors generate.

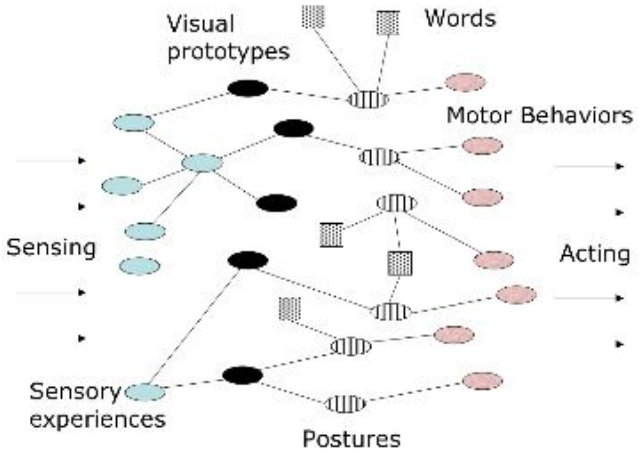


Figure 2: Network linking sensory experiences, image schemata of postures, nodes for postures acting as mirror neurons, and nodes triggering the motor behavior that achieves the posture. Nodes for words (shown as squares) are associated with the posture nodes.

these experiments, agents start without any prior set of image schemata nor words for bodily actions, and they do not know the mapping from the visual domain to the motor domain. If we observe an increase in communicative success, starting from scratch, then this means that agents have not only self-organised a lexicon for naming possible actions and the image schemata they generate, but that they have learned a bidirectional mapping between the visual and motor domain as well. We will see that this is indeed possible.

In the first experiment reported earlier, called the mirror experiment (Steels and Spranger, 2008), robots learn the bi-directional mapping between visual body-image and motor behavior by standing before a mirror, executing actions, and observing the visual body-images that they generate. Once all agents in the group have each learned this mapping, they play language games settling on names for these actions. In the second experiment (called the body language experiment), robots do not learn the bi-directional mapping between image schemata and motor body-image through a mirror but through the language game itself. Both experiments are briefly summarised in the next sections of the paper. Then we focus on the role of imagination through self-simulation. As advocated by several researchers, self-simulation can be used to enhance understanding of the movements of others (Rizzolatti and Arbib, 1998; Jeannerod, 2001), and thus play a role in language understanding and learning (Feldman, 2006; Feldman and Narayanan, 2004). In the experiment to be discussed, robots maintain a motor body image of themselves which they update through proprioception. Additionally we equipped the robots with a kinematic model used for simulating the execution of their own actions. By adding a component that is able to inspect this simulated body image, it becomes possible for the robot to imagine to some extent what a particular action looks like and this helps to guess the meaning of unknown words and thus speed up the coordination between visual body image, motor behaviors and language.

The Mirror Experiment

In the mirror experiment, each robot stands before a mirror in order to acquire the relation between his own motor body-image and (a mirror image) of his own visual body-image (see figure 1). Our experiments have so far focused on static gestures (postures) which require motor behaviors that each involve typically about 20 motor commands with associated proprioceptive feedback. Because all robots have exactly the same body, a robot can use a visual body image of himself in order to categorise the body image of another robot, after perspective reversal (Steels and Loetzsch, 2008). And so once each robot has learned the relation between visual body-image and motor body-image for himself, they are quickly able to settle on a shared vocabulary by playing an Action Game (see figure 7).

We approach the problem of body image here as a coordination problem. Agents maintain a 'semiotic' network linking nodes for image schemata for postures with the motor behaviors that they generate, mediated by a 'posture' node which is functioning as a mirror neuron (see figure 2). Similar to research by Triesch et al. (2007) we assume that there is nothing special about mirror neurons but that neurons become mirror neurons because they take on a particular role in networks. The vision system of the robot performs foreground/background segmentation and feature extraction in



Figure 3: Aspects of visual processing. From left to right we see the source image, the foreground/background distinction, the result of object segmentation (focusing on the upper torso), and the feature signature for successive frames of this posture based on centralised moments.

terms of centralised moments (Mukundan and Ramakrishnan, 1998, see figure 3). Values for each of these features are combined into a feature vector that constitutes the sensory experience of a perceived body image at a particular moment in time. The specific visual feature vectors are then classified using image schemata. An image schema is a feature vector with the same dimensions as the sensory experience and with a typical point as well as maximum accepted deviations from this point. The best matching prototype is found by distance computation and then the prototype is adjusted to better assimilate the new experience. When no prototype is matching or a new action is selected and executed, a new one is created, with the sensory experience being the first seed. Each image schema is linked to a 'posture node' which is also linked to a motor behavior which can achieve that particular posture when executed. When one robot is watching another robot, he performs a perspective reversal operation on the visual image, in the sense that the robot computes the position of the other robot and then perform a geometric transformation, so that the position of left and right arm now matches with his own. For example, if one robot stands in front of another one, this means a 180 degree transformation.

The inventory of image schemata and their relation to sensory features is acquired using a prototype based approach and the links between posture nodes and motor behaviors through kinesthetic teaching. When a robot stands before the mirror, he selects a posture and activates the corresponding motor behavior. This motor behavior generates a sensory image which is categorised with a particular image schema. And so through standard Hebbian learning, which enforces the connection between nodes that are simultaneously active (Hebb, 1949; Bishop, 1995), the link between the image schema and the posture gets established and progressively enforced. Once this network is established, the development of a shared lexicon in a population of agent is straightforward, based on the well known lateral inhibition dynamics of the Naming Game (Steels, 1995).

Results of this experiment for a population of 10 agents are shown in figure 4. The graphs show the global behavior of the population, after each individual has coordinated motor behavior and visual body-image through the mirror.

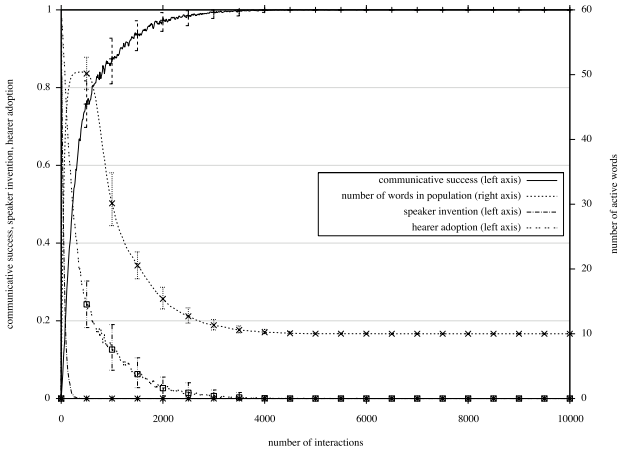


Figure 4: Results of the Action Game played by a population of 10 agents for 10 postures. Robots have first coordinated their visual body-images and motor behaviors by standing before a mirror and observing their visual appearance in relation to certain motor behaviors. The x-axis plots the number of language games. The running average of communicative success and average lexicon size are shown as well as invention and adoption frequency.

100 % success is reached easily after about 2000 games. Already after 900 games there is more than 90 % success. The graph shows the typical overshoot of the lexicon in the early stage as new words are invented in a distributed fashion followed by a phase of alignment as the agents converge on an optimal lexicon. Figures 5 and 6 show snapshots of the semiotic dynamics for the first 500 games. Figures 5 shows the communicative success, lexicon-size, and invention and adoption frequency. Figure 6 we show the average score in the population for different words competing for the same meaning. We see clearly that a winner-take-all situation arises after about 300 games, with one word dominating for naming this particular posture.

Coordination without mirrors

We have seen that once coherent links exist between the image schema for a posture and the motor behavior that generates this posture (here mediated by the posture nodes acting as mirror neurons), it is straightforward for a group of agents to self-organise a lexicon of names which can be used both for describing a particular posture or as commands for asking another robot to achieve it, and thus for playing the Action Game. We now report a second earlier experiment where robots coordinate visual body-image and motor behaviors through language without using a mirror first and without using posture nodes (see Steels and Spranger, 2008 for more details). The game is similar to the mirror experiment except that now the two different robots stand immediately in front of each other (see figure 7), without any prior

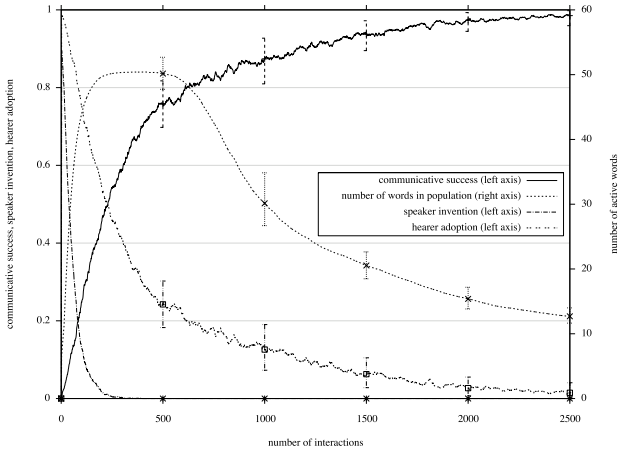


Figure 5: The same graphs as shown in the previous figure, but now zooming in on the first 500 language games. The agents quickly communicate successfully every second game. The population starts to communicate 100% successfully as soon as the number of word drops to the optimal number, by eliminating unsuccessful hypothesis.

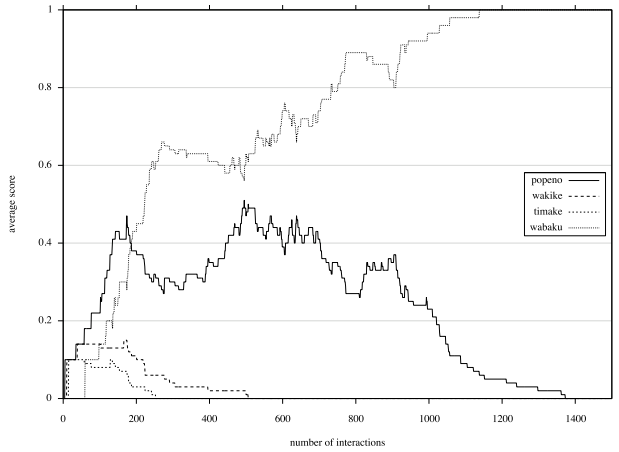


Figure 6: The graph shows the average score for all the words competing for the same meaning (the raise both arms action in this case). For every agent of the population the score of all words (scoring higher than 0) is averaged. The data stems from an experiment equal to the experiments depicted in figures 4 and 5. The winner-take-all dynamics that coordinates the lexicon among the agents is clearly visible. The agents create words for the action, these words become coordinated, with some of dying out and one surviving the competition.

exposure to mirrors. The speaker asks the hearer to perform an action and there is communicative success if the speaker agrees that the right action has been performed. Moreover the speaker also performs the motor-behavior that is linked to the word.

More precisely, the interaction pattern is as follows:

1. The speaker randomly chooses an image-schema from the known schemas as the topic of the conversation.
2. He looks up a word associated with the schema. If there is none he invents one.
3. He looks up the motor-behavior associated with the word. If there is no associated motor-behavior he picks one randomly.
4. The speaker utters the word and performs the motor-behavior picked.
5. The hearer parses the uttered word and looks up the image-schema and motor-behavior associated with the word.
6. If there are no image-schemas associated with the word, he classifies the image-schema resulting from the motor-behavior performed by the speaker and associates it with the word.
7. The hearer performs the motor-behavior associated with the word. If there is none he picks one of the known motor-behaviors randomly.
8. Both agents determine the success of the interaction. The speaker determines if the hearer performed the intended behavior. The hearer determines whether the image-schema associated with the uttered word corresponds to the motor-behavior performed by the speaker. That is he compares the visual features created by the action of the speaker with his expected image-schema.

Upon determination of success the lexicon of the agents taking part in the interaction is updated. Successful links between network nodes are enforced, unsuccessful links or links to nodes which were not activated are punished (lateral inhibition).

Results of this second experiment for a population of 5 agents and 5 postures are shown in figure 8. The graphs show the global behavior of the population focusing on the first 5000 language games. 100 % success is reached after about 5000 games and stays stable (unless new postures get introduced and then the networks of all agents expand to cope with this). Already after 3000 games there is more than 90 % communicative success. The graph shows the typical overshoot of the lexicon in the early stage as new words are invented and a phase of alignment as the agents converge on an optimal lexicon size, which is 5 words to name each of the postures. The frequency of invention and adoption is also shown and dies down as the lexicon stabilises. So clearly agents are able to self-organise a lexicon of commands even if no prior lexicon exists and even if they are not pre-programmed nor learned mappings from motor behaviors to visual image schemata before the lexical process

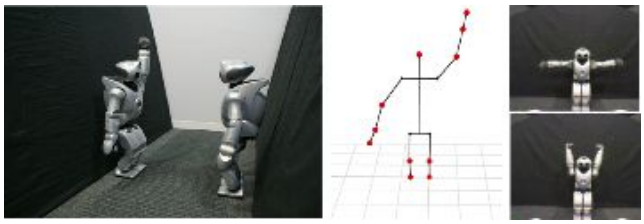


Figure 7: Two humanoid robots face each other to play an Action Game. They ask each other to achieve a certain posture, like raise the left arm, or stretch out both arms. The middle image shows the internal motor-body-image obtained by combining proprioceptive streams with the body model. The right image shows two example postures as seen through the camera of another robot.

starts.

Communicative success does not necessarily mean that agents relate all visual prototypes 'correctly' to the corresponding motor behaviors. However figure 9 shows that the 'right' correspondences progressively emerge. The figure plots the aggregated strength (brightness, black: strongly linked, white no link) of the relation between visual prototypes (y-axis) and motor behaviors (x-axis) in this population over time. Which links are established by agents and which links are consistently established by all agents is shown with two diagrams. The first one (bottom row) shows the evolution of agreement among the agents (computed by taking the product of the strength over all different agents and all experiments) and the second one (bottom row) shows whether agents make any link at all (computed by taking the average strength of all relations between visual prototypes and motor behaviors for all agents for a series of experiments) If there is a single column between a particular visual prototype p_i and its corresponding motor behavior m_i with strength 1.0, then this means that all agents have this particular mapping (bottom row) and agree on it (top row).

Using Imagining through Self-Simulation

We now address the question whether it is possible to improve the efficiency of the overall system. The first step is to realise that agents are dealing with a search problem. When the speaker does not know which motor behavior corresponds to a posture he would like to see achieved, he has to make a guess (step 2) and when the hearer encounters an unknown word, he has to choose a motor behavior that could be associated with the visual image he is perceiving (step 7). In the interaction pattern used in the previous experiment, these choices are made randomly, which implies that the choice is correct in only $\frac{1}{A}$ of the cases, with A equal to the number of possible actions, explaining why we see a worsening of performance as the number of actions increases. We now show how this choice can be improved dramatically by us-

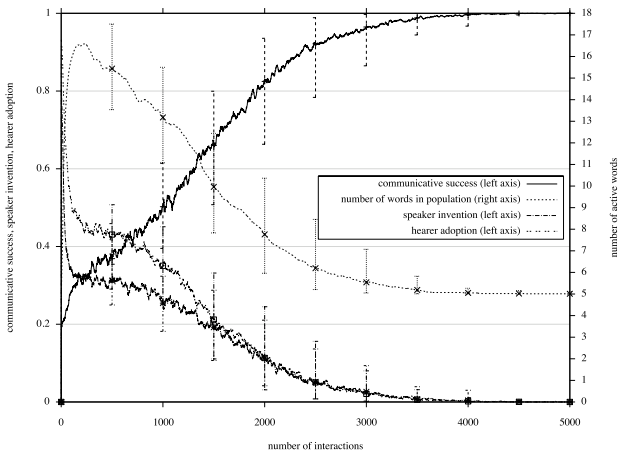


Figure 8: Results of the Action Game played by a population of 5 agents for 5 postures (the results of 100 experiments are averaged). Robots have NOT coordinated their visual body-images and motor behaviors by using a mirror but only through language. The x-axis plots the number of language games. The running average of communicative success and average lexicon size as well as invention and adoption frequency are shown on the y-axis.

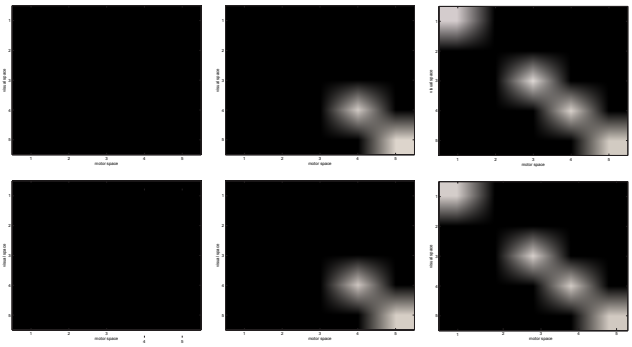


Figure 9: Relation between visual prototypes and motor behaviors for a population of five agents negotiating names for 5 postures. The diagram shows the aggregated strength of the relations over time. Top row: aggregation by product thus showing where they completely agree. Bottom row: aggregation by sum showing which relations have been made. Left column: after 500 interactions. Middle column: after 1500 interactions. Right column: after 2500 interactions. The diagram shows that agents progressively achieve the correct mappings and agree on them.

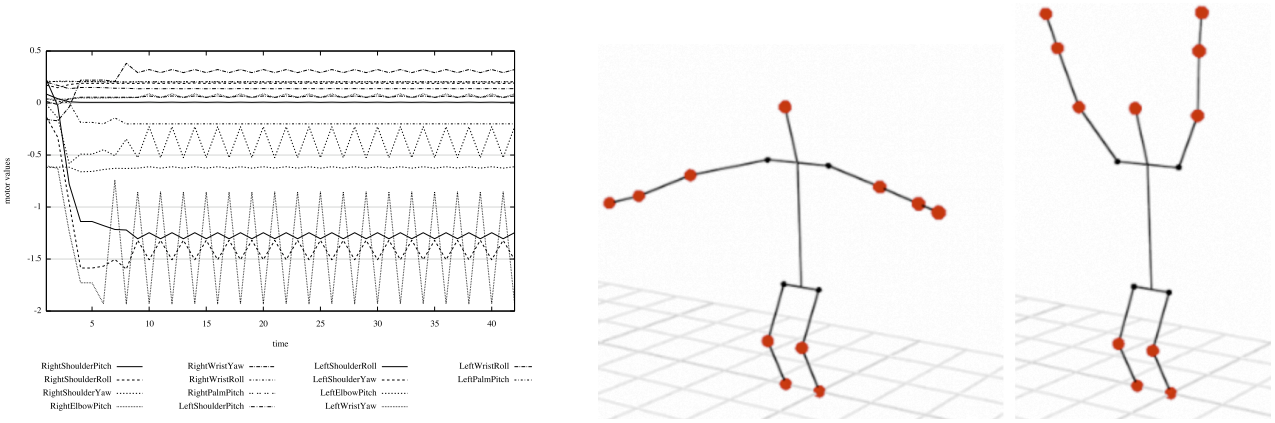


Figure 11: Left: The image shows a motor control data stream for creating a waving motion. The information can be used to create an 'expected motor body image' by combining the information the robot has about its body, limb lengths and dimensions, as well as the position of the motors. Given even more information stemming from proprioceptive sensors, a similar representation called the 'experienced motor body image' can be constructed. Given the knowledge about the position of the sensors as well as the aforementioned configuration of the body, representations like those to the right can be computed. The two images on the right show two stages in the simulation of the 'raise both arms' action creating an 'expected motor body image', as well as an 'experienced motor body image'.

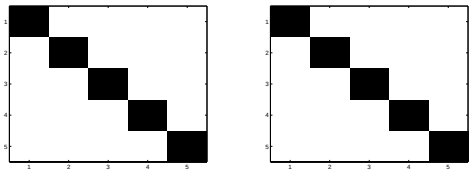


Figure 10: Visuo-motor correspondence results from 100 experiments (5 agents, 5 actions), after 10000 interactions. Left: aggregation by sum showing which associations have been made overall. Right: aggregation by product showing which associations agents agree upon. In all these experiments agents find always the right mappings and establish the correct visuo-motor connections.

ing a simulation-based approach (Feldman, 2006; Feldman and Narayanan, 2004).

The basic idea is straightforward. Each robot maintains an *experienced motor body image* (see figure 7, middle image) which is based on his own body model (containing information about body parts, their size and shape, how they are attached to each other) and recorded proprioceptive motor streams. In addition to the experienced motor body image, robots can also construct an *expected motor body image* for a particular action using his own body model, the initial conditions before the action starts, a model of the body kinematics (see figure 11 right) and recorded motor control streams, such as the one shown in figure 11 left, with quantities like RightShoulderPitch, RightShoulderRoll, RightShoulderYaw, RightElbowPitch, etc. This predictive model is already employed in the adaptive control systems of the robot and is now put to a new usage.

First of all, the expected motor body image can be decoupled entirely from ongoing behavior so that the robot can simulate a complete action. Of course the simulation will deviate from reality depending on the complexity of the action and the amount of interaction with the environment that is needed. Second, a simulated visual body image can be generated from the simulated motor body image. For example, the motor body image contains information about the angles of the different joints which can be used to internally visualise the position of the joints with respect to the body. Note that this visual 'imagining' is from the robot's own frame of reference. In order to create an expectation of what the same behavior performed by another robot would look like, a third step is needed: The robot has to perform a perspective reversal by performing a geometric transform on this visual image, based on knowing the position of the other robot (Steels and Loetzsch, 2008). Finally, similar visual processing and categorisation can be carried out on this simulated visual body image as on the 'real' experienced visual body image of the other robot, specifically centralised moments can be computed again to extract the features needed for gesture categorisation.

Given this competence in visual imagination, robots can now improve drastically the quality of their guesses in steps 2 and 7 of the interaction pattern. Speakers can use the simulated visual body image of the other robot to make a better guess of the correct motor-behavior given a visual prototype of the posture they want to see adopted by the other robot. This is done by a hill-climbing process that starts from a random choice of action from the action repertoire, simulates

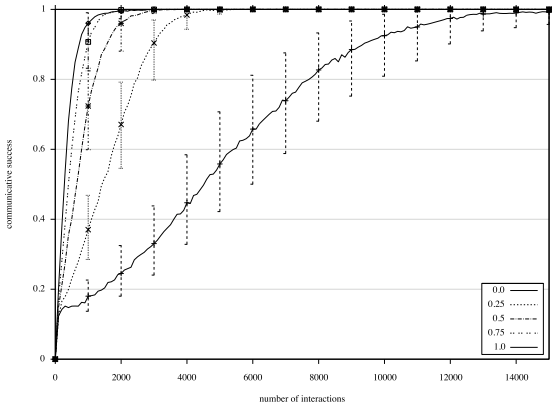


Figure 12: Influence of simulation on the performance of the agents in the Action Game (5 agents, 10 actions). The figure shows the communicative success averaged over 100 different experimental runs, each for a single parameter. 0.0 is the case were the speaker guesses the correct motor-behavior with the base line probability $\frac{1}{10}$. 1.0 means that speaker and hearer "guess" correctly every time they have to choose a motor-command to match with a desired visual prototype (as speaker) or interpret an unknown word (as hearer).

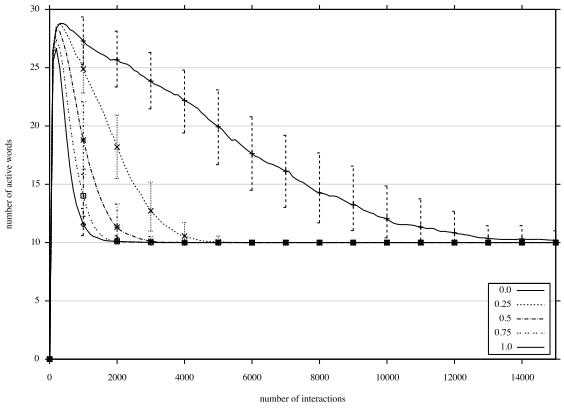


Figure 13: Study of the influence of simulation on the performance of the agents in the game (5 agents, 10 actions). The graph depicts the number of words (average lexicon size across all agents) averaged over 100 experiments for a given parameter. 0.0 is the case where the speaker guesses the correct motor-behavior with the base line probability $\frac{1}{10}$. 1.0 means that the speaker "guesses" correct every time he guesses a motor-command.

that choice to construct a visual body image of the other robot, and compares this to the desired posture. In case of a failure, a new action is chosen until a reasonable match is found. The hearer goes through a similar procedure if he has to guess the meaning of an unknown word. As mentioned earlier, simulated behavior will always deviate from actual behavior and so this process does not give an absolutely certain guess.

Figure 12 shows the important impact on performance of this simulation-based approach. The base-line case (marked 0.0) is the one used in earlier experiments, i.e. where the agent has a $\frac{1}{A}$ chance to choose the correct action. It shows the slowest rise towards communicative success. The best case (marked 1.0) is one where the agents manage to always guess correctly both as speaker and as hearer what motor behavior achieves a visual prototype of a posture based on self-simulation, imagination, perspective reversal, and visual categorisation. This case approaches the performance in the mirror experiment where agents first learned the mapping between visual body image and motor body image by standing in front of a mirror. In the intermediary cases we see that that the higher the probably of correct guessing the faster the population reaches full communicative success.

Figure 13 shows the semiotic dynamics for the vocabulary of the agents for the same series of experiments. There is always a phase of invention and spreading with a typical overshoot in the size of the vocabulary, which then settles down on an optimal vocabulary as agents align their word meanings. We see that the overshoot of words is much smaller when the quality of guessing improves, which means that time to convergence is also much shorter.

Conclusions

This paper examined the role of body language in the formation of body image, specifically whether Action Games in which agents ask each other to perform actions, can lead to a coordination of image schemata and the motor behaviors that generate them. We have seen that this is indeed the case, even if there is no prior coordination of image schemata and motor behaviors (through a mirror for example) or even if no prior lexicon is given to the agents but they have to self-organise one from scratch. This paper focused on whether overall performance could be improved by allowing agents to 'imagine' what their own bodily movements by simulating. Even if self-simulation is not perfect and only partial information can be gleaned from the motor body image, we have seen that agents have now a better way to guess the meaning of unknown words and hence they can faster zoom in on a system that coordinates different body images of the self and those of others.

Acknowledgements

This research has been carried out at the Sony Computer Science Laboratory in Paris with partial support from the ECAgents project, funded by the EU Future and Emerging Technologies program (IST-FET) as IST-1940, and the ALEAR project, funded by the EU Cognitive Systems program. We are extremely grateful to Masahiro Fujita, Hideki Shimomura and their team at the Sony Intelligent Systems Research Lab in Tokyo for giving the opportunity and support

to perform the QRIO experiments, and to Martin Loetzsch for his help in setting up the experiments. The information provided here is the sole responsibility of the authors and does not reflect the EU Commission's opinion. The Commission is not responsible for any use that may be made of data appearing in this publication.

References

Billard, A. (2002). Imitation: a means to enhance learning of a synthetic proto-language in an autonomous robot. In Dautenhahn, K. and Nehaniv, C. L., editors, *Imitation in Animals and Artifacts*, pages 281–311. MIT Press.

Bishop, C. (1995). *Neural Networks for Pattern Recognition*. Oxford University Press, USA.

Blanke, O. and Castillo, V. (2007). Clinical neuroimaging in epileptic patients with autoscopic hallucinations and out-of-body experiences.case report and review of the literature. *Epileptologie*, 24:90–96.

Demiris, Y. and Johnson, M. (2003). Distributed, predictive perception of actions: a biologically inspired robotics architecture for imitation and learning. *Connection Science Journal*, 15(4):231–243.

Feldman, J. (2006). *From Molecule to Metaphor: A Neural Theory of Language*. MIT Press.

Feldman, J. and Narayanan, S. (2004). Embodied meaning in a neural theory of language. *Brain and Language*, 89(2):385–392.

Gutmann, J., Fukuchi, M., and Fujita, M. (2005). Real-Time Path Planning for Humanoid Robot Navigation. In *Int. Joint Conference on Artificial Intelligence (IJCAI), Edinburgh, Scotland*.

Hebb, D. (1949). *The Organization of Behavior: A Neuropsychological Theory*. John Wiley & Sons.

Jeannerod, M. (2001). Neural Simulation of Action: A Unifying Mechanism for Motor Cognition. *Neuroimage*, 14(1):103–109.

Mataric, M. (2002). Sensory-Motor Primitives as a Basis for Imitation; Linking Perception to Action and Biology to Robotics. In Dautenhahn, K. and Nehaniv, C. L., editors, *Imitation in Animals and Artifacts*, pages 392–422. MIT Press.

Mukundan, R. and Ramakrishnan, K. (1998). *Moment Functions in Image Analysis: Theory and Applications*. World Scientific.

Nabeshima, C., Kuniyoshi, Y., and Lungarella, M. (2006). Adaptive body schema for robotic tool-use. *Advanced Robotics*, 20(10):1105–1126.

Pfeifer, R., Bongard, J., and Grand, S. (2007). *How the body shapes the way we think: a new view of intelligence*. Cambridge, Mass.: MIT Press.

Ramachandran, V. and Hirstein, W. (1998). The Perception of Phantom Limbs. *Brain*, 121(9):1603–1630.

Rizzolatti, G. and Arbib, M. (1998). Language within our Grasp. *Trends in Neurosciences*, 21(5):188–194.

Rizzolatti, G., Fadiga, L., Gallese, V., and Fogassi, L. (1996). Premotor Cortex and the Recognition of Motor Actions. *Cognitive Brain Research*, 3(2):131–141.

Rosenfield, I. (1988). *The invention of memory: a new view of the brain*. Basic Books, New York.

Steels, L. (1995). A Self-Organizing Spatial Vocabulary. *Artificial Life*, 2(3):319–332.

Steels, L. (2003). Evolving grounded communication for robots. *Trends in Cognitive Sciences*, 7(7):308–312.

Steels, L. and Belpaeme, T. (2005). Coordinating perceptually grounded categories through language: A case study for colour. *Behavioral and Brain Sciences*, 28(4):469–489.

Steels, L. and Loetzsch, M. (2008). Perspective alignment in spatial language. In Coventry, K. R., Tenbrink, T., and Bateman, J. A., editors, *Spatial Language and Dialogue*. Oxford University Press. To appear.

Steels, L. and Spranger, M. (2008). The robot in the mirror. *Connection Science*, 20(4).

Triesch, J., Jasso, H., and Deak, G. (2007). Emergence of Mirror Neurons in a Model of Gaze Following. *Adaptive Behavior*, 15(2):149–165.

Multiple Functionalities of Biochemical Reaction Networks

M.N. Steijaert, A.M.L. Liekens, H.M.M. Ten Eikelder, P.A.J. Hilbers

Department of Biomedical Engineering, Eindhoven University of Technology,
P.O.Box 513, 5600 MB Eindhoven, The Netherlands
M.N.Steijaert@tue.nl

Abstract

We consider a biological cell as a highly interconnected network of chemical reactions, which is constituted of a large number of semi-autonomous functional modules. Depending on the global state of the network, the separate functional modules may display qualitatively different behavior. As an example, we study a conceptual network of phosphorylation cycles, for which the steady-state concentration of an output compound depends on the concentrations of two input enzymes. We show that the input-output relation depends on the expression of the proteins in the network. Hence changes in protein expression, due to changes in the global regulatory network of the cell, can change the functionality of the module. In this specific example, changed expression of two proteins is sufficient to switch between the functionalities of various logical gates.

Introduction

The human body consists of over 200 distinct cell types, which display a large variety in both morphology and physiology. Often differences between cell types manifest themselves in the response to extracellular stimuli. The same type of molecule may even have a totally different function in different cell types. An example for this is the response of cells to the neurotransmitter acetylcholine (see Alberts et al. (2002)). The extracellular presence of this compound yields contraction in skeletal muscle cells but a decrease of contraction in heart muscle cells and even a multitude of different effects in many other cells.

On the other hand, as all cells in an organism descend from the same zygote, they share the same potential reaction network, i.e., they are capable of producing the same set of macromolecules, the interactions among which are based on the same rules (e.g., rate and diffusion constants). Therefore, the key to differences in behavior must lie in different configurations of the network. In a systems theory sense, these cells can be looked upon as similar systems that due to a different state show very diverse responses to identical stimuli. The state variables of these systems are the concentrations of chemical species, including species with an important regulatory function such as transcription factors. During differentiation, a chain of extracellular stimuli and cell-cell interactions pushes the network for each cell in a certain region of its state space. Each of those regions is characterized by its pattern of present transcription factors (and other regulatory molecules) and consequently a corresponding pattern of protein expression. Not only through development, but also in adult individuals, subtle changes in configuration occur. Examples are synaptic plasticity in neurons and adaptations to a changed environment, but also many diseases coincide with a changed configuration of the reaction network of individual cells: in tumor cells the cell cycle control is disrupted (Hanahan and Weinberg (2000)) and in metabolic syndrome the cellular respiratory system is affected (Kitano et al. (2004)).

In order to deal with the vast number of interactions in an intracellular reaction network, it is common practice to observe semi-autonomous parts of the system as separate modules (Hartwell et al. (1999)). Such a separation is not necessarily artificial, as many biological systems have a tendency towards modularity. Interestingly, modularity has been reported by Variano et al. (2004) to emerge when evolving artificial networks described by linear differential equations with a fitness function that rewards network stability . Depending on the exact scale of observation and field of interest, many definitions for modules and modularity exist (see for instance Polani et al. (2005), Bongard (2002)). In this paper, we use the word *module* to refer to a group of proteins (and metabolites) that interact with each other on the level of protein-protein (and protein-metabolite) reactions, but not with proteins or metabolites outside the module. We do, however, allow the transcriptional and translational regulatory systems of the cell to alter the concentrations of proteins inside the module. Hence, as the tuning of transcription and translation differs between cell types, this may affect protein concentrations and, because of that, the dynamics of the module of interest. This applies to all intracellular reaction networks that depend on protein concentrations, among which metabolic pathways and signaling networks.

Here, we focus on the influence of protein expression on the functionality of a module. There are a number of ways

in which proteins expression can influence this functionality. Obviously, when none of the involved proteins are expressed, the module of interest would be switched off. As a result, many processes are present in only a few types of cells. For instance, only a small number of cells synthesize certain neurotransmitters, although all cells have the blueprints for the necessary enzymes in their DNA. Apart from that, the changes in protein expression can have many more effects on the input-output relations of functional modules. We can roughly distinguish two classes of effects. Firstly, the expression of enzymes involved in a module can have quantitative effects on the input-output relations of the module. For instance, muscle cells are set to take up larger amounts of glucose than less energy consuming cell types such as skin cells. Secondly, there is the possibility that the observed functionality itself is changed in a qualitative fashion. An example is the MAP kinase signaling network of Bhalla et al. (2002), in which the expression of MAP kinase phosphatase determines whether the network displays a gradual or a bistable response to extracellular stimuli.

In this paper we exemplify with a conceptual model, that the steady-state input-output relations of a signal transduction network consisting of phosphorylation cycles may change dramatically by changing only a few concentrations. Moreover, this model shows that even if we know the topology of a network and the sign (i.e., positive or negative influence) and strength of its interactions, the behavior depends heavily on the actual concentrations of proteins. We show that with the same topology and rate constants, this network, which consists of only 5 phosphorylation cycles, can have at least 8 different input-output relations, depending on the chosen protein concentrations. Note that, from a mathematical point of view, there is only one function that is calculated by the network. However, we consider some of the concentrations as parameters that are determined outside the model. The input-output relation for each parameter set is thus considered as a separate function.

The implementation of Boolean logic in biochemical reactions is not new. For instance, Strack et al. (2008) have recently build wet lab implementations of AND, OR, XOR and 'B AND NOT A' functions using enzymatic reactions. Boolean logic is also a subject of research in the field of genetic regulatory networks (see for instance Schilstra and Nehaniv (2008)). Here, we describe one single network that with suitable parameters can compute 8 different Boolean functions.

Model description

Phosphorylation cycles

Phosphorylation cycles are common building blocks of intracellular signaling networks (Cohen (2000)). The generic phosphorylation cycle involves a single type of protein which can be in two states: a phosphorylated and a dephosphorylated state. In a phosphorylation reaction, a phosphate group is transferred from a donor molecule (commonly ATP) to a specific site on the dephosphorylated protein. The enzymes that catalyze this 'forward' reaction are called kinases. The 'backward' reaction, i.e., the dephosphorylation reaction, is catalyzed by a phosphatase enzyme. A schematic representation of such a cycle is shown in Figure 1 a.

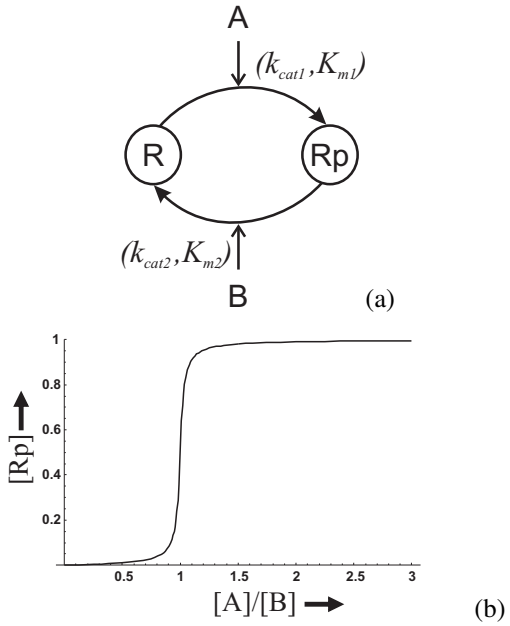


Figure 1: (a) Phosphorylation cycle with one kinase A and one phosphatase B, for which the dynamics can be described by two Michaelis-Menten reactions with constants k_{cat1} and K_{m1}, and k_{cat2} and K_{m2}, respectively. (b) The steady-state concentration of Rp as a function of the ratio of the concentrations of A and B, for the case $k_{\mathrm{cat1}} = k_{\mathrm{cat2}} = 1$ and $K_{\mathrm{m1}} = K_{\mathrm{m2}} = 0.01$.

In this paper we restrict our description of the phosphorylation cycle to a highly idealized cycle consisting of two Michaelis-Menten type reactions (Fersht (1999)). Furthermore, we assume a fixed concentration of phosphate donors. In this way, we can describe the rates v_{phos} and v_{dephos} of the reactions

$$\begin{aligned} \mathrm{A} + \mathrm{R} &\rightarrow \mathrm{A} + \mathrm{Rp} \\ \mathrm{B} + \mathrm{Rp} &\rightarrow \mathrm{B} + \mathrm{R}, \end{aligned}$$

with Michaelis constants K_{m1} and K_{m2} and catalytic constants k_{cat1} and k_{cat2} by

$$v_{phos} = \frac{k_{\mathrm{cat1}}[\mathrm{A}][\mathrm{R}]}{K_{\mathrm{m1}} + [\mathrm{R}]},$$

$$v_{dephos} \frac{k_{\mathrm{cat2}}[\mathrm{B}][\mathrm{Rp}]}{K_{\mathrm{m2}} + [\mathrm{Rp}]}.$$

Note that square brackets indicate the concentration of the corresponding compound. The steady-state of such a cycle

is given by Goldbeter and Koshland (1981) and can be written as (Tyson et al. (2003)):

$$[\mathrm{Rp}] = \frac{2\,v_1 J_2 R_{\mathrm{tot}}}{v_2 - v_1 + v_2 J_1 + v_1 J_2 + D}\,,$$

where

$$D = \sqrt{(v_2 - v_1 + v_2 J_1 + v_1 J_2)^2 - 4(v_2 - v_1) v_1 J_2}$$

and $R_{\mathrm{tot}} = [\mathrm{R}] + [\mathrm{Rp}]$, $v_1 = [\mathrm{A}]\,k_{\mathrm{cat1}}$, $v_2 = [\mathrm{B}]\,k_{\mathrm{cat2}}$, $J_1 = K_{m1}/R_{\mathrm{tot}}$, $J_2 = K_{m2}/R_{\mathrm{tot}}$.

Typically, the steady-state response of such a cycle is sigmoidal as function of $[\mathrm{A}]/[\mathrm{B}]$(see Figure 1 b). As both the phosphorylated and dephosphorylated proteins can act as enzymes themselves, phosphorylation cycles can be coupled in a cascade, in which the substrate of one cycle is a kinase or phosphatase in another reaction. In the next section we introduce a network of such cycles, in which we assume that only the phosphorylated form is active as an enzyme and can have either kinase or phosphatase activity but not both. We do not take into account formation of protein-protein complexes other than the kinase-substrate and phosphatase-substrate complexes taken care of in the model of Goldbeter and Koshland.

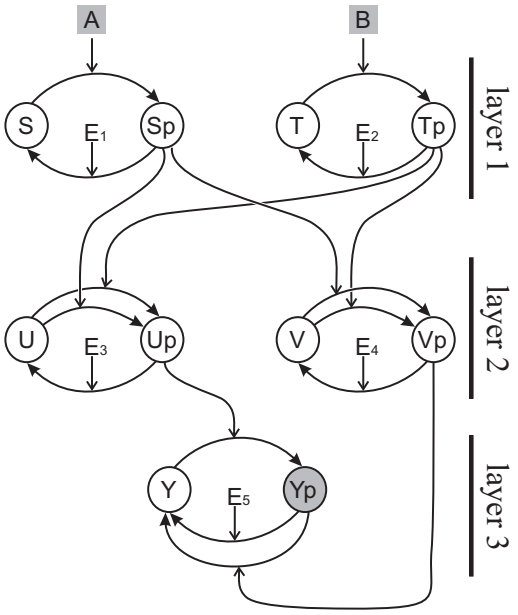


Figure 2: Reaction scheme of the network. System parameters $[\mathrm{E}_1]...[\mathrm{E}_5]$, $S_{\mathrm{tot}}, T_{\mathrm{tot}}, U_{\mathrm{tot}}, V_{\mathrm{tot}}$ determine the response of $[\mathrm{Yp}]$ to input concentrations $[\mathrm{A}]$ and $[\mathrm{B}]$.

Network topology

Consider the network of phosphorylation cycles as shown in Figure 2. Depending on the external concentrations $[\mathrm{A}]$ and $[\mathrm{B}]$ (which are further referred to as 'input concentrations'), the state of the network will change. We focus on the steady-state concentration of Yp for given concentrations $[\mathrm{A}]$ and $[\mathrm{B}]$. Clearly, this concentration depends on the concentrations of uncoupled enzymes ($[\mathrm{E}_1]...[\mathrm{E}_5]$) and the total amount of mass in each phosphorylation cycle (i.e., $S_{\mathrm{tot}} = [\mathrm{S}] + [\mathrm{Sp}]$, $T_{\mathrm{tot}} = [\mathrm{T}] + [\mathrm{Tp}]$, $U_{\mathrm{tot}} = [\mathrm{U}] + [\mathrm{Up}]$, $V_{\mathrm{tot}} = [\mathrm{V}] + [\mathrm{Vp}]$ and $Y_{\mathrm{tot}} = [\mathrm{Y}] + [\mathrm{Yp}]$). The regulation of the amount of protein expression (i.e., transcription, translation and protein degradation) is part of the chemical network of the entire cell, but not of the module of interest. Therefore, the total concentrations of involved proteins are ways for the global system of the cell to tune the functionality of the described module. From here on, we refer to these concentrations as system parameters. Clearly, the rate constants of all the reactions are parameters of the module as well. However, it seems unlikely that the global network, apart from the expression of inhibitors or activators, can change the values of these parameters. Therefore, we keep the rate constants of all reactions at 1.

As there is no feedback, each phosphorylation cycle in our model has one single stable solution irrespective of its initial state. As there are only feed-forward connections, there is only one steady-state solution for the entire network as well for given system parameters and inputs ($[\mathrm{A}]$ and $[\mathrm{B}]$). When two or more enzymes catalyze the same reaction, we consider the sum of their concentrations as the concentration of the catalyst.

Parameter values

In the previous section we have defined system parameters as variables of the global cell network that determine the behavior of functional modules. The presented conceptual model of such a module has 10 system parameters ($[\mathrm{E}_1]...[\mathrm{E}_5]$, S_{tot}, T_{tot}, U_{tot}, V_{tot} and Y_{tot}), which can be used to configure the network. In order to obtain more insight in the possible behaviors of the network, we focus on its possibilities as a logic gate. For the input we define values below 0.5 as False and above 0.5 as True. We show that only two system parameters (S_{tot} and T_{tot}) have to be adjusted to obtain 8 different logical functions with 2 inputs. For all reactions we choose the Michaelis constant $K_{\mathrm{m}} = 0.01$ and the catalytic constant $k_{\mathrm{cat}} = 1$.

The first layer of the network functions as a thresholding device. That is, as we do not demand inputs $[\mathrm{A}]$ and $[\mathrm{B}]$ to be exactly 0 or 1, we want $[\mathrm{Sp}]$ be near 0 if $[\mathrm{A}] < [\mathrm{E}_1]$ and near S_{tot} if $[\mathrm{A}] > [\mathrm{E}_1]$. The same holds for the T-Tp cycle. We choose thresholds $[\mathrm{E}_1] = [\mathrm{E}_2] = 0.5$. As both phosphorylation cycles in the second layer receive input from both $\mathrm{S_p}$ and $\mathrm{T_p}$, we consider the sum of their concentrations (i.e. $[\mathrm{Sp}] + [\mathrm{Tp}]$) as the output of the first layer. The input-output relation of layer 1 is given by Table 1. The output of the first layer $[\mathrm{Sp}] + [\mathrm{Tp}]$ is approximately one of the four numbers 0, T_{tot}, S_{tot} or $S_{\mathrm{tot}} + T_{\mathrm{tot}}$. Note that, as S_{tot} and T_{tot} are the only adjustable system parameters, these four numbers are not yet fixed.

Layers 2 and 3 take $[\mathrm{Sp}] + [\mathrm{Tp}]$ as input, and have $[\mathrm{Yp}]$ as output. If we also take $U_{\mathrm{tot}} > 0$, $[\mathrm{E_5}] \approx 0.5\,U_{\mathrm{tot}}$, $V_{\mathrm{tot}} \gg U_{\mathrm{tot}}$, and $Y_{\mathrm{tot}} = 1$ and choose any combination of $[\mathrm{E_3}]$ and $[\mathrm{E_4}]$ for which $[\mathrm{E_3}] < [\mathrm{E_4}]$, we obtain the input-output relation for layers 2 and 3 that is shown in Figure 3. Now, the only parameters that are not yet fixed are S_{tot} and T_{tot}.

[A]	0	0	1	1
[B]	0	1	0	1
[Sp] + [Tp]	0	T_{tot}	S_{tot}	$S_{\mathrm{tot}} + T_{\mathrm{tot}}$

Table 1: Input-output relation of layer 1

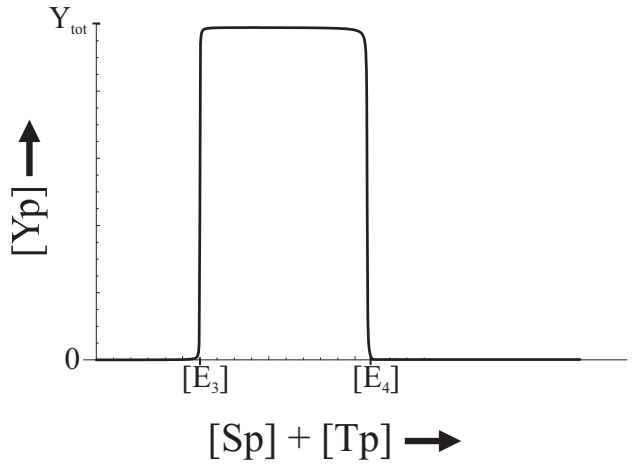


Figure 3: Response of layer 3, when layer 2 receives a total input $[\mathrm{Sp}] + [\mathrm{Tp}]$.

Network behavior

Different combinations of S_{tot} and T_{tot} can yield different computable functions, as shown in Figure 4. Because the response of the network is built up of sigmoidal functions, the borders between the shown regions do not indicate discrete changes in functionality but rather narrow continuous transitions.

For instance, to compute the XOR function, we choose S_{tot} and T_{tot} such that $[\mathrm{E_3}] < S_{\mathrm{tot}} < [\mathrm{E_4}]$, $[\mathrm{E_3}] < T_{\mathrm{tot}} < [\mathrm{E_4}]$ and $[\mathrm{E_4}] < S_{\mathrm{tot}} + T_{\mathrm{tot}}$. Similarly, to compute the OR function we require $[\mathrm{E_3}] < S_{\mathrm{tot}} < [\mathrm{E_4}]$, $[\mathrm{E_3}] < T_{\mathrm{tot}} < [\mathrm{E_4}]$ and $[\mathrm{E_3}] < S_{\mathrm{tot}} + T_{\mathrm{tot}} < [\mathrm{E_4}]$. It is easily verified from Table 2 and Figure 3 that indeed the XOR respectively OR function are computed. Note that the conditions on S_{tot} and T_{tot} for the OR function require that $[\mathrm{E_3}] < [\mathrm{E_4}]/2$. As can be seen from Figure 4, all 8 Boolean functions that yield 0 for input $[\mathrm{A}] = [\mathrm{B}] = 0$ can be computed if $[\mathrm{E_3}] < [\mathrm{E_4}]/2$.

If we consider both concentrations $[\mathrm{Y}]$ and $[\mathrm{Yp}]$ as outputs, our network is able to calculate all 16 possible logical functions for two inputs. The reason is that, because $[\mathrm{Y}] = 1 - [\mathrm{Yp}]$, we can consider $[\mathrm{Y}]$ as the negation of $[\mathrm{Yp}]$.

The possible outputs of the network are shown in Figure 5. Also, without considering $[\mathrm{Y}]$, other logical functions may be calculated, as well as functions with a more gradual response to different inputs. To this end, alternative choices for system parameters, other than S_{tot} and T_{tot} should be used. Even more possibilities may appear by adding extra enzymes (kinases or phosphatases) to some of the phospho-

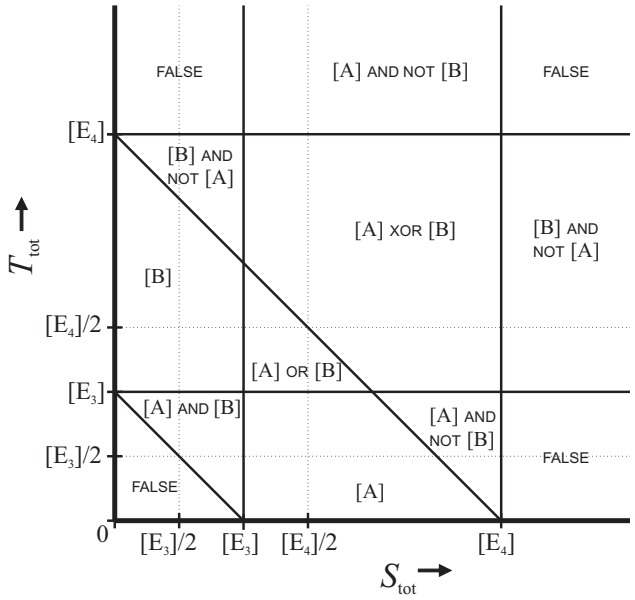


Figure 4: Network output for combinations of S_{tot} and T_{tot}. Only relative concentrations are shown, see Figure 5 for examples of actual parameter values.

XOR

[A]	0	1	0	1
[B]	0	0	1	1
[Sp]+[Tp]	0	$> [\mathrm{E_3}]$, $< [\mathrm{E_4}]$	$> [\mathrm{E_3}]$, $< [\mathrm{E_4}]$	$> [\mathrm{E_4}]$
[Yp]	0	1	1	0

OR

[A]	0	1	0	1
[B]	0	0	1	1
[Sp]+[Tp]	0	$> [\mathrm{E_3}]$, $< [\mathrm{E_4}]$	$> [\mathrm{E_3}]$, $< [\mathrm{E_4}]$	$> [\mathrm{E_3}]$, $< [\mathrm{E_4}]$
[Yp]	0	1	1	1

Table 2: Computing the XOR or the OR function. The values of S_{tot} and T_{tot} (see Figure 4) determine the range of values of $[\mathrm{Sp}] + [\mathrm{Tp}]$ and by that the value of $[\mathrm{Yp}]$ for the possible combinations of input values $[\mathrm{A}]$ and $[\mathrm{B}]$.

rylation cycles. However, this is beyond the scope of this paper.

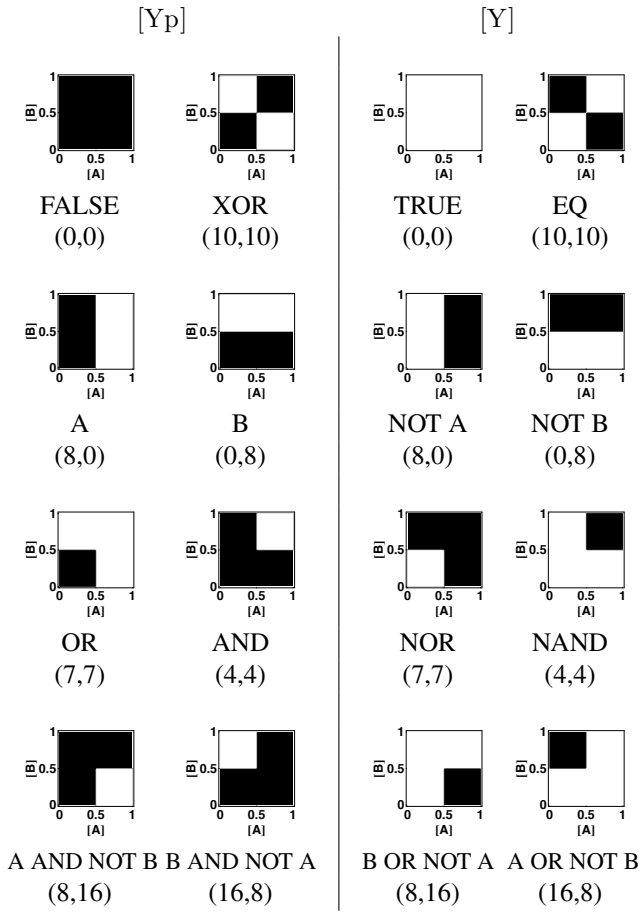


Figure 5: Depending on the network configuration, the steady-state concentrations of Yp and Y correspond to one of the 16 logical functions for 2 inputs $[\mathrm{A}]$ and $[\mathrm{B}]$. (Gray levels indicate concentration: Black ≈ 0 , White ≈ 1). All sub-figures were plotted using $[\mathrm{E}_3] = 6$, $[\mathrm{E}_4] = 16$, $[\mathrm{E}_5] = 0.5$, $U_{\mathrm{tot}} = 1$, $V_{\mathrm{tot}} = 10$ and the values for S_{tot} and T_{tot} that are given in brackets for each sub-figure.

We have shown how the values of only two parameters determine the response of the network. The importance of parameter values is even more strikingly exemplified if we use the same network as a unary logic operator. In order to do that, we consider $[\mathrm{B}]$ as a system parameter instead of an input, choose S_{tot} and T_{tot} within the 'XOR region' and leave the other parameter values unchanged. In that case changing the value of $[\mathrm{B}]$ is sufficient to switch the output $[\mathrm{Yp}]$ of the network from identity to negation of input $[\mathrm{A}]$.

Biological Plausibility

Although the topology and parameters of the model are optimized to display specific idealized behavior rather than to describe any known intracellular pathway, the model consists of building blocks that are common in intracellular signaling. It may therefore be possible to find a similar topology within existing biological networks. Indeed, in the Human Protein Reference Database (see Peri et al. (2003) and Mishra et al. (2006)), we have found a number of protein kinases and phosphatases that show similar patterns of interactions. More specifically, we have searched this database (Release 7, downloaded from http://www.hprd.org) for combinations of five proteins S,T,U,V and Y for which the proteins S and T are each known to phosphorylate at least one site on the proteins U and V. In addition, protein U has to phosphorylate protein Y and protein V has to dephosphorylate Y. Furthermore, we require that both S and T are targets for phosphorylation by other protein kinases. We did not discriminate between multiple phosphorylation sites on the same protein.

With these requirements, we have identified the combinations of S,T,U,V and Y, that are listed in Table 3. Note that proteins S and T are interchangeable. For each of the possible proteins for S and T, known kinases (i.e. potential network inputs A and B) are listed in Table 4. On protein PRKACA there are at least 5 different sites that are known targets for phosphorylation, but for which no upstream kinases are known. Finding these networks with similar topology makes it more plausible that this type of multifunctional modules can occur in nature. However, due to the current limited knowledge about parameters of signalling networks, it remains unclear whether such functionality is indeed used in biology.

S	T	U	V	Y
FYN	LCK	SHC1	ACP1	ZAP70
LCK	LYN	PRKCD	PTPN6	EGFR
MAPK1	PRKACA	RAF1	PTPN7	MAPK1
PRKCA	PRKACA	SRC	PTPN12	ABL1
PRKCA	PRKACA	SRC	PTPN12	PTK2

Table 3: Real networks with same topology. S and T are interchangeable.

Substrate (S or T)	Kinase (A or B)
FYN	FYN, CSK, PDGFRB
LCK	CSK, LCK, PRKCA, MAPK1, SYK, PRKACA, MAPK3
LYN	LYN, MATK, CSK
MAPK1	MAPK1, RET, MAP2K1, RAF1
PRKACA	Kinase unknown
PRKCA	PRKCZ, SYK

Table 4: Kinases working on possible substrates S and T.

Discussion

In order to get some grip on the overwhelming complexity of biochemical interactions within a cell, it is common practice in biology to analyze separate pathways or modules instead of the whole system. We have defined a module in such a way that it can be analyzed separately from the larger network to which it belongs. Changes outside the module are therefore considered as changes in parameters or boundary conditions, rather than changes of the state of the module itself. Because of this, the same module may have different parameters in two different cell types, and may therefore display different functionalities. Since a small module, such as the conceptual network presented in this paper, can already behave totally different depending on the values of two parameters, it is likely that also in real networks changes in protein expression result in changes in observed functionality. Note that, from a mathematical point of view, there is no such thing as a change in functionality, as the network of interactions is still the same. However, when dealing with small parts of the system, different behavior can be observed as different functionalities. The model presented in this paper gives an example how small changes in parameters of a phosphorylation network can already yield a different functionality. For example, the presence or absence of the compound B is sufficient to switch the input-output relation between $[\mathrm{A}]$ and $[\mathrm{Yp}]$ from identity to negation.

Although this model is not based on any known intracellular pathway, we have shown that similar topologies can be found in the Human Protein Interaction Database. Despite the idealized sigmoidal response of the phosphorylation cycles, this small network can already behave in at least 8 different ways, depending on how it is configured. This makes it plausible that modules of real biological networks display such multifunctional properties as well, which gives a clue how the same protein in different cells can be involved in different processes and can even show contradictory behavior among cell types. Moreover, this multifunctionality may also be exploited by evolution to use the same modules for different purposes in different cell types. Spatial isolation of proteins may even allow the exploitation of different functionalities in separate regions in the same cell. On the other hand, the same functionality may be needed in many different cell types. In that case it appears to be advantageous if the involved genes are co-regulated, as this would preserve the ratios between the protein concentrations and by that the functionality of the module.

As for many signaling networks quantitative data is lacking or unreliable, an often used technique is to use Boolean networks to model positive and negative interactions between nodes (Kauffman (1969), see de Jong (2002) for a review). Although this technique is useful to understand some interactions within a complicated network, a lot of qualitative effects are missing with this approach. We can illustrate this with our conceptual model. Cycles U-Up and V-Vp are both positively influenced by cycles S-Sp and T-Tp and indirectly by external concentrations $[\mathrm{A}]$ and $[\mathrm{B}]$. As in a classical Boolean network interactions only have a sign but not a weight, the U-Up and V-Vp cycles are identical. The Y-Yp cycle receives positive influence from Up and negative influence from Vp. As the topology and interactions remain the same, a purely qualitative approach would be insufficient to describe the different 'modes' of the network. This also shows that for understanding the dynamics of these complex networks, it is necessary to perform multiple quantitative measurements at the same time.

Acknowledgements

This work is supported by the European Commission through the Evolving Cell Signaling Networks in Silico (ESIGNET) project of the Sixth Framework Programme.

References

Alberts, B., Johnson, A., Lewis, J., Raff, M., Roberts, K., and Walter, P. (2002). *Molecular Biology of the Cell*. Garland Science, 4th edition.

Bhalla, U. S., Ram, P. T., and Iyengar, R. (2002). MAP kinase phosphatase as a locus of flexibility in a mitogen-activated protein kinase signaling network. *Science*, 297(5583):1018–1023.

Bongard, J. (2002). Evolving modular genetic regulatory networks. In *Proceedings of the Evolutionary Congress Computation on 2002 (CEC 2002)*, pages 1872–1877.

Cohen, P. (2000). The regulation of protein function by multisite phosphorylation–a 25 year update. *Trends Biochem Sci*, 25(12):596–601.

de Jong, H. (2002). Modeling and simulation of genetic regulatory systems: A literature review. *Journal of Computational Biology*, 9(1):67–103.

Fersht, A. (1999). *Structure and mechanism in protein science : a guide to enzyme catalysis and protein folding*. W.H. Freeman and Co., New York.

Goldbeter, A. and Koshland, D. E. (1981). An amplified sensitivity arising from covalent modification in biological systems. *Proc Natl Acad Sci USA*, 78(11):6840–6844.

Hanahan, D. and Weinberg, R. A. (2000). The hallmarks of cancer. *Cell*, 100(1):57–70.

Hartwell, L. H., Hopfield, J. J., Leibler, S., and Murray, A. W. (1999). From molecular to modular cell biology. *Nature*, 402(6761 Suppl):C47–52.

Kauffman, S. (1969). Metabolic stability and epigenesis in randomly constructed genetic nets. *J. Theor. Biol.*, 22:437–467.

Kitano, H., Oda, K., Kimura, T., Matsuoka, Y., Csete, M., Doyle, J., and Muramatsu, M. (2004). Metabolic syndrome and robustness tradeoffs. *Diabetes*, 53 Suppl 3:S6–S15.

Mishra, G., Suresh, M., Kumaran, K., Kannabiran, N., Suresh, S., Bala, P., Shivakumar, K., Anuradha, N., Reddy, R., Raghavan, T., Menon, S., Hanumanthu, G., Gupta, M., Upendran, S., Gupta, S., Mahesh, M., Jacob, B., Mathew, P., Chatterjee, P., Arun, K., Sharma, S., Chandrika, K., Deshpande, N., Palvankar, K., Raghavnath, R., Krishnakanth, R., Karathia, H., Rekha, B., Nayak, R., Vishnupriya, G., Kumar, H., Nagini, M., Kumar, G., Jose, R., Deepthi, P., Mohan, S., Gandhi, T., Harsha, H., Deshpande, K., Sarker, M., Prasad, T., and Pandey, A. (2006). Human protein reference database–2006 update. *Nucleic Acids Res.*, 34:D411–414.

Peri, S., Navarro, J., Amanchy, R., Kristiansen, T., Jonnalagadda, C., Surendranath, V., Niranjan, V., Muthusamy, B., Gandhi, T., Gronborg, M., Ibarrola, N., Deshpande, N., Shanker, K., Shivashankar, H., Rashmi, B., Ramya, M., Zhao, Z., Chandrika, K., Padma, N., Harsha, H., Yatish, A., Kavitha, M., Menezes, M., Choudhury, D., Suresh, S., Ghosh, N., Saravana, R., Chandran, S., Krishna, S., Joy, M., Anand, S., Madavan, V., Joseph, A., Wong, G., Schiemann, W., Constantinescu, S., Huang, L., Khosravi-Far, R., Steen, H., Tewari, M., Ghaffari, S., Blobe, G., Dang, C., Garcia, J., Pevsner, J., Jensen, O., Roepstorff, P., Deshpande, K., Chinnaiyan, A., Hamosh, A., Chakravarti, A., and Pandey, A. (2003). Development of human protein reference database as an initial platform for approaching systems biology in humans. *Genome Res.*, 13:2363–2371.

Polani, D., Dauscher, P., and Uthmann, T. (2005). On a quantitative measure for modularity based on information theory. *Proceedings of the VIIIth European Conference on Artificial Life (ECAL 2005)*, pages 393–402.

Schilstra, M. and Nehaniv, C. (2008). Bio-logic: gene expression and the laws of combinatorial logic. *Artif. Life*, 14:121–133.

Strack, G., Pita, M., Ornatska, M., and Katz, E. (2008). Boolean Logic Gates that Use Enzymes as Input Signals. *Chembiochem*, 9:1260–1266.

Tyson, J. J., Chen, K. C., and Novak, B. (2003). Sniffers, buzzers, toggles and blinkers: dynamics of regulatory and signaling pathways in the cell. *Curr Opin Cell Biol*, 15(2):221–231.

Variano, E., McCoy, J., and Lipson, H. (2004). Networks, dynamics, and modularity. *Phys. Rev. Lett.*, 92:188701.

An Artificial Chemistry-based Model of Economies

Bas Straatman[1,2], Roger White[2] and Wolfgang Banzhaf[3]

[1]Energy and Environmental Systems Group, ISEEE, University of Calgary, Calgary, AB, T2N 1N4, CANADA
[2]Department of Geography, Memorial University, St. John's, NL, A1B 3X5, CANADA
[3]Department of Computer Science, Memorial University, St. John's, NL, A1B 3X5, CANADA
bas.straatman@ucalgary.ca

Abstract

Economies can be modelled using Artificial Chemistry approaches. In this contribution we discuss the development of such a model starting from the well-known von Neumann's technology matrices. Skills and technologies that allow the transformation of raw materials into products are introduced in a form akin to chemical reactions. The dynamic flow of materials in such a system is simulated and connected through an agent-based market mechanism that assigns value to raw materials, labour, and products. Starting from a fixed set of raw materials, energy and labor, we observe the appearance of new products, the use of consumables and the general increase in complexity of such a system. Real evolutionary dynamics including waves of innovation can be demonstrated.

Introduction

Economies are notoriously difficult to understand. Multiple approaches have been tried, yet progress has been very slow and today we are still not able to understand or predict the dynamics of an economy. much less its structural evolution.

Part of the reason is that economists are notoriously conservative in their approaches to elucidate these problems. While many disciplines have whole-heartedly embraced the new non-linear paradigms of Science, self-organization, the emergence of new function, chaos and complexity, non-equilibrium systems dynamics and evolution, economists have been skeptical and mostly seem to be concerned with equilibria, exact mathematical solutions to differential equations and systems without surprises.

Another reason is simply that economies are complex, dynamic, non-linear, and innovative. New products and companies arise constantly, grow in dominance in the marketplace, get competition, weaken - perhaps gradually - and are finally replaced by others. This unabating renewal process is one of the most fascinating, yet poorly understood aspects of economies. Schumpeter called it the "gales of creative destruction" (Schumpeter, 1939).

While there is a field of "Evolutionary Economics" (Metcalfe, 1998; Witt, 2006), it can be argued that evolution really does not play a key role in that area, since the key aspect of innovation is not appropriately modeled in Evolutionary Economics to date.

What we are going to do here is to take the evolutionary aspect of an economy seriously. For that to happen, we need the ability of our system to generate new products, new technologies and new companies. Artificial Chemistry has for a long time been proposed as a means to study constructive (innovative) aspects of systems. An artificial chemistry in its broadest definition (Dittrich *et al*., 2001) consists of a collection of objects, transformation rules and an algorithm that drives the dynamics of their transformation. Here we will make use of an artificial chemistry to model the production system of the economy to be simulated. Objects can be goods such as raw materials (or labour and consumables), products of production processes created through rules of transformation, and technologies (which can be considered catalysts in a chemical sense).

The next section will introduce the von Neumann technology matrices and how we can use this method as a starting point for a production system. We will then introduce a simple system that will allow us to simulate an economy. It will be based on natural numbers being symbols of raw materials (prime numbers), products (products of natural numbers) and skills/technologies (in the form of indices). The fourth section will introduce the production agents, i.e. the "companies", and the marketplace, key elements of a viable economy which bring a valuation to the various goods of the production system introduced previously. The fifth section will demonstrate a run of the system, and explain how new products or technologies can be created (and observed). The 6th section will discuss typical runs watching for waves of innovation, competition for market dominance, and in general, the evolutionary dynamics.

Von Neumann Technology Matrices

Von Neumann proved the existence of a general economic equilibrium in an economic system undergoing balanced growth (von Neumann, 1946). He used matrices to model the transformation of input to output in an economic system. Before we can make use of this notation, we need to

explain it in some detail.

Suppose we have m activities and n commodities in an economy. The activities can be regarded as production processes, skills or technologies and are denoted by a vector $t_1, ..., t_m$, whereas the commodities can be regarded as labour, capital, raw materials, or products, and are denoted by a vector $c_1, ..., c_n$.

We can then formulate an input matrix I to our production process, and an output matrix O. Here is an example with $m = 4$ and $n = 5$.

Table 1: Input and output matrices representing technology and the products involved

commodities	*input matrix*				*output matrix*			
	t_1	t_2	t_3	t_4	t_1	t_2	t_3	t_4
c_1	1	1	2	0	0	0	0	6
c_2	0	1	0	0	1	0	0	0
c_3	0	0	1	0	0	4	0	0
c_4	0	0	1	0	0	0	2	0
c_5	0	0	0	1	0	0	$\frac{7}{3}$	0

The columns of Table 1 can be read the following way: $c_1 \overrightarrow{t_1} c_2$, $c_1 + c_2 \overrightarrow{t_2} 4 \times c_3$, $2 \times c_1 + c_3 + c_4 \overrightarrow{t_3} 2 \times c_4 + \frac{7}{3} \times c_5$, and $c_5 \overrightarrow{t_4} 6 \times c_1$, that is, each of the pairs of columns in Table 1 indicates a process that transforms a combination of commodities into a set of output commodities.

The original purpose of the von Neumann Technology matrices was to be able to formally show that an equilibrium can be achieved with certain production factors appropriately chosen. It later turned out that a "balanced growth" scenario could be supported by this formalism as well in which all production processes expand at the same rate. For example, the matrices in Table 1 allow a constant growth rate of 2 when the activity vector z that denotes the number of times each of the m processes is executed per time step equals $\{6, 3, 6, 7\}$. Indeed $(O - 2I).z = 0$.

However, the notation is not restricted to equilibrium situations at all. We can make use of the same model, and include innovation by the addition of new columns (an innovation in technology) or new rows (new products appearing). Most of the time, a mixture of both would be necessary. A good outline of technology matrices is provided by Blatt (1983).

A Simple Economy based on Natural Numbers

The particular production system we are going to use here is inspired by the $\mathbb{N}$ economy using natural numbers to represent commodities (Herriot and Sawhill, 2008). The idea is to have multisets of objects, in this case natural numbers, which can be transformed into others using the operation of multiplication.

This system is indeed an Artificial Chemistry, with the reactions being the multiplication, and the numbers the equivalent to chemical species. As such, the system has some similarity with the prime number reaction system introduced in Banâtre *et al*. (1988); Banzhaf *et al*. (1996).

Just to briefly recall, an Artificial Chemistry can be represented by a triple (S, R, A), where S is the set of all possible molecules, R is a set of collision or reaction rules, and A is an algorithm describing the domain and how the rules are applied to the molecules (Dittrich *et al*., 2001). This general notation for an artificial chemistry can be applied to develop a framework for an artificial economy. In a free-market economy, the incentives for production are provided by the market. The algorithm should therefore represent a market in order to provide the rules for what is produced, when, and where. In addition, S – the set of all products – and R – the set of all production processes – need to be defined.

In the $\mathbb{N}$ economy the set S of molecules is the set P of products (which consists of commodities such as raw materials, products, capital and labour), with each product being represented by a natural number. Suppose the product set P consists of the following goods:

$$P = \{2, 3, 5, 7, 11, 13, 130, 260, 104\}$$

Here we have chosen to represent commodities by integers: prime numbers for raw materials, and products of integers for composites. This allows us to use the structure offered by the natural numbers, and products literally can be decomposed into their prime factorisation to see what raw materials are involved in making them. In this model, labour is treated explicitly as a commodity (the number `2`). Furthermore, there is a special product, which serves as money, and this is the number `3`.

Suppose we have a small product set, as above, and five production processes per time interval, as displayed in Table 2.

Table 2: Overview of the production processes during one time interval

2× 2							→	2× 5
2× 2	+	2× 5	+	2× 13			→	3× 130
3× 2	+	3× 130					→	6× 260
6× 2	+	6× 260					→	6× 104
6× 104							→	13× 2

Note that there is no production of commodity `13`, yet it is required in one of the production processes. This product is assumed to be a free good, such as sunlight. It can easily be verified that the production in Table 2 is in balance; total labour (product `2`) used equals total labour generated, and the products required for the generation of this labour are

also covered, so that the required input per time interval is exactly the same as the output at the end of the time interval. Based on these production activities per unit of time, it is possible to determine a price set that will allow the system to function. In order for this to be the case, a producer needs to be able to finance the inputs for the next time interval with the sale of the current production. In other words, the price of the output should be greater than or equal to the price of the input. Assuming that activity levels for the next time step will remain the same, and since there is no surplus in the system, this gives the following set of equalities. When p_i stands for the price of product i, then

$$\begin{aligned} 2p_{5} &= 2p_{2} \\ 3p_{130} &= 2p_{2} + 2p_{5} + 2p_{13} \\ 6p_{260} &= 3p_{2} + 3p_{130} \\ 6p_{104} &= 6p_{2} + 6p_{260} \\ 13p_{2} &= 6p_{104} \end{aligned}$$

When the price of one product (the numeraire good) is taken as a price unit, for example the price of labour $p_2 = 1$ and the free good $p_{13} = 0$, then this system of equalities has a unique solution, namely $p_2 = 1, p_5 = 1, p_{13} = 0, p_{130} = 4/3, p_{260} = 7/6, p_{104} = 13/6$.

The von Neumann Technology matrix for the example above is shown in Table 3.

Table 3: Input and output matrices representing technology and the products involved

products	*input matrix*					*output matrix*				
2	1	$\frac{2}{3}$	$\frac{1}{2}$	1	0	0	0	0	0	1
3	0	0	0	0	0	0	0	0	0	0
5	0	$\frac{2}{3}$	0	0	0	1	0	0	0	0
13	0	$\frac{1}{3}$	0	0	0	0	0	0	0	0
130	0	0	$\frac{1}{2}$	0	0	0	1	0	0	0
260	0	0	0	1	0	0	0	1	0	0
104	0	0	0	0	$\frac{6}{13}$	0	0	0	1	0

Now that we have described the production system in some detail, let us turn our attention to what happens with all these products.

Economic Agents and their Marketplace

A production system alone is not sufficient to simulate an economy. In addition to production processes and products/raw materials, active entities like companies, here called agents, are necessary to actually produce quantities of these objects. Further, a validation process for objects/products will be required that allows to close the circle by exchanging objects between agents. This is the price setting mechanism which in our economy is realized by an auctioneer who keeps an eye on stock levels and who reports prices to agents.

Space

At the cost of labour, raw materials are extracted from the land, which has been divided into cells of equal size. As far as raw materials are concerned, the space is homogeneous. If desired, it is possible to experiment with more interesting distributions of resources.

Each cell provides a free resource, `13`, which can be thought of as energy from sunlight. The distribution of free energy can be varied, but currently it is such that there is always an abundance.

On top of the land is a connection network, which represents the presence of trade links between cells. Links go both ways, so if a cell a is connected to a cell b, then b is connected to a, and all agents on a can trade with all agents on b, and vice versa. Initially the network is empty, but as agents obtain skills/technologies and require inputs for these skills, they can establish connections with providers. A wide range of rules of how to expand or contract an agent's trade network is possible. For example, an agent can randomly select one of its current providers and subsequently do a local search around the selected trade partner in an attempt to find an additional provider. Or an agent may be allowed to connect to the nearest provider. In addition, it is possible that an agent looses one of the more distant connections and subsequently attempts to find a supplier nearer by. Such rules obviously attempt to keep the social network consisting of trading partners compact. The rules according to which agents behave are described in the next section.

Agents

The grid space is home to a number of economic agents. They all have a fixed location and an identity number, so that we can distinguish different agents on one location (cell). Furthermore, agents possess assets: resources, other products present in the economy and skills. All assets are stored in a list. Therefore, an agent can be represented by

$$\{\{x, y, z\}, \{c_1, \ldots, c_n\}, \{t_1, \ldots, t_m\}\}$$

where x, y is the location, z is the identity number for that agent, $c_1, \ldots, c_n$ lists the possession of n commodities, and $t_1, \ldots, t_m$ is a boolean list to specify which of the m technologies the agent possesses for some $m, n \in \mathbb{N}$. Different agents have different skills.

Having a technology should be regarded as having the skill or knowledge to perform a certain transformation when you have the resources to do so. For actions that require capital, in addition to having the skill, the agent must also possess the appropriate capital goods in order to execute that particular action. Capital goods are a subset of the product

list P. In particular, they are products, analogous to catalysts, that are necessary for the production of other products, but which are not used up in the process – that is, they are recovered (less a fraction representing depreciation) at the end of the production process.

As mentioned before, some products are consumables, while others are capital or intermediate products. The consumables can be converted into labour, `2`, according to certain columns in the technology matrices. The actions of these particular columns represent consumption, and consumption is restricted to a certain type of agent, namely the group of consumers. We thus distinguish two types of agents: consumers and producers. The latter can have any of the other technologies. Producers depend on consumers for the required labour, while consumers depend on producers for the consumables. Neither group can have skills that belong to the other group: an agent is either consumer, or producer, but never both. This is not crucial to the functioning of the model and could easily be relaxed later.

Agents in action

The production of goods, mentioned in the previous section, does not simply happen by itself. It occurs because somebody (something) somewhere actually does the job. Therefore, we need actual agents in space to serve as economic individuals to drive the economy. The whole environment of space, networks, agents, product set and technology is called the economy. All changes that take place in the economy occur because agents perform some or all of the following actions:

Win free resources: Free resources are distributed over the cells; some agents will use them.

Expand network: An agent asks around among its trading partners (all agents it is linked with) to inquire what they have in store. It compares this list with what is possibly required for executing its own skills. If there are any products it does not have access to, it will randomly choose a trading partner, then scan the surroundings of the trading partner for interesting new partners. A connection to the cell that contains the agent that offers most of what was not yet available will be added to the connection network.

Make plan: What an agent can do, firstly depends on its skills. Secondly, resources need to be available for the transformation of commodities into products. Resources can be bought, but the agent is limited by what its partners have to offer. A surplus of products can be sold to generate money to buy the required resources. However, the quantity of products being sold is dependent on the monetary resources of trading partners. In the end, the agent's initial possession, plus what is acquired, minus what is sold, minus what is used in the production process, plus what is produced must be positive. Given these constraints, and given the prices of all commodities (determined by an auctioneer explained below), every agent uses linear optimization to determine what and how much to sell, what and how much to buy, and what and how much to produce in order to achieve maximum possession.

Buy: Once an agent has optimized its plan, it will look for a partner that has what it is looking for, and they will exchange q units of product l for $q \times p_l$ units of money, where p_l is the current value of product l.

Sell: Once an agent has optimized its plan, it will look for a partner that has the money to buy what the agent wants to sell, and they will exchange q units of product k for $q \times p_k$ units of money, where p_k is the current value of product k.

Produce: After the exchange of products and money, the agent has all the resources required for the production plan and it can transform the input commodities into its output.

Update: An agent executes all of these steps, and the state of the agent and the agents engaged in trade are updated.

Unlike cells in cellular automata, the agents cannot be updated synchronously, because one agent's action will change the state of its partners with whom it engages in trade. Therefore, the agents perform sequentially. All the agents are randomly ordered, and in turn they go through the above list of actions. When everybody has had a turn, a new randomly ordered list is created for the next iteration. A complete sequence of actions of all agents' defines one iteration. The random list of agents generated anew for each iteration prevents any bias due to specific ordering of the agents.

Suppose the input matrix and output matrix are I and O. Dimensions of these matrices are $n \times m$, meaning that there are m production processes and that the economy consists of n products, with $m, n \in \mathbb{N}$. As in previous examples, the first product is labour and the second product represents one unit of money. Now expand these matrices by adding the matrices M and $\mathbb{I}$:

$$\begin{aligned} A &= \left(I \;\middle|\; M \;\middle|\; \mathbb{I} \right) \\ B &= \left(O \;\middle|\; \mathbb{I} \;\middle|\; M \right) \end{aligned}$$

where $\mathbb{I}$ is the identity matrix of size n, and M is the square matrix of size n with all elements equal to 0 except for the second row which is equal to $p = \{p_1, p_2, \ldots p_n\}$ indicating the quantities of commodity 3 that need to be paid for each of the products. The addition of these two matrices to I and O represent all possible actions involving production, buying and selling. Just as columns in I and O represent the input and output for production processes, the columns

in M and $\mathbb{I}$ represent input and output for buying products. When we write $z = \{z_1, z_2, \ldots z_{m+2n}\}$ as the vector of all actions of an agent, that is, z consists of elements $z_1, \ldots z_m$ to indicate the activities regarding the m production processes, $z_{m+1}, \ldots, z_{m+n+1}$ to indicate the quantities of products that need to be bought, and $z_{m+n+2}, \ldots, z_{m+2n}$ the quantities of products that need to be sold, then $A \cdot z$ lists the quantities of products required for the execution of vector z and $B \cdot z$ lists the quantities of products generated by vector z. The quantities $A \cdot z$ and $B \cdot z$ are named the input and output, respectively, of the vector of action z. Likewise, when the vector $p = \{p_1, p_2, \ldots p_n\}$ lists the prices for each of the products, then $p \cdot (B - A)$ is a vector that gives the profit of each of the actions and $p \cdot (B - A) \cdot z$ equals the profit generated by activity z.

Every agent's behaviour is described by

$$\max_{z \in \mathbb{Q}^{m+2n}} profit(z) = \max_{z \in \mathbb{Q}^{m+2n}} p \cdot (B - A) \cdot z$$

under the conditions that:

- a positive balance is maintained both in production and in trade,
- the total activity per turn per agent is capped and
- consumers meet their basal metabolic rate; i.e. the minimum level of consumption necessary for them to remain alive and productive.

The prices $p = \{p_1, p_2, \ldots p_n\}$ mentioned above are determined by an auctioneer who attempts to find a set of prices such that, for those products for which there is a shortage, the production process becomes profitable (cost of input lower than price of output). For the production processes of products for which there is a surplus, prices are set such that these can be manufactured, but without profit (cost of input equals price of output). To a certain extent such prices are similar to Sraffa's market clearing prices based on the cost of input. When dealing with a shortage, the value of the output is such that it enables the sector to buy the required input and produce with a profit. Through this, agents are encouraged to reduce shortages and surpluses. The presence of surplus and shortage is determined as follows: the auctioneer keeps track of what input is required during the last w iterations (often $w = 50$). Subsequently, the auctioneer attempts to keep in store the required input for those w iterations. There are more advanced models of agent-based market dynamics, see for example Tesfatsion (2007), but these focus on all aspects of procurement. As such, much effort goes into the bottom-up determination of the market price through a cyclical process of matching offers and bids, which is much more than we need in this paper.

A Sample Run

The above described model of an artificial chemistry-based economy is illustrated here by a presentation of a typical simulation run. The agent population consists of 15 producers with random skills (the know-how to execute particular transformations given by the technology matrices), and 15 consumers. All agents are connected to all other agents such that the spatial configuration is irrelevant for the time being. The technology matrices are comparable to those in Table 3, that is they deal with the same products but have slightly different output coefficients to allow a surplus production such that the basal metabolic rate of consumers can be satisfied.

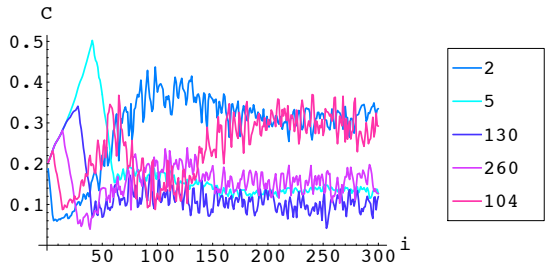


Figure 1: Product concentrations C per iteration i of the commodities 2, 5, 130, 260 and 104.

Figure 1 graphs the resulting product concentrations, where a concentration of a product is the proportion of the total product population. At the start of the simulations each of the agents was supplied with equal amount of each of the products. Initially only the stock levels of the labour (product 2) is reduced due to the basal metabolic rate of the consumers, and thus its concentration goes down while the other concentrations go up. This phase includes merely consumption, and thus the concentrations change linearly. However, the reduced stock of labour leads to a profitable production of this "product", which in turn leads to reduced levels of the consumable (commodity 104) and subsequently of other commodities involved. Agents do not necessarily have access to the required inputs, depending on the random order in which they are allowed to act for example, and thus the production of commodities appears a little erratic. However, this simple economy converges to a more or less stable state which corresponds to the Leontief stable state.

The next figure graphs the price dynamics for one of the commodities (product 104), and its close relation to the stock levels (see Figure 2). The blue line indicates the price, the red line indicates a surplus or a shortage in stock levels. Appropriate stock levels depend on the past activity of the agents. Note that a higher price does not guarantee an immediate reaction in stock levels, but that every time stock levels do become positive this is preceded by an increased price.

The temporarily higher prices trigger agents to engage in

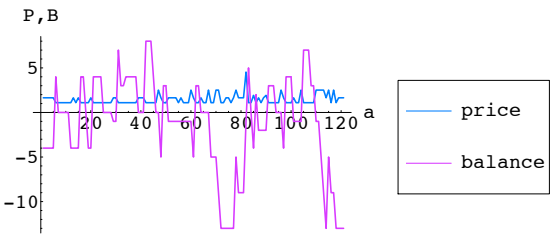


Figure 2: Price development P and balance B of product 104 graphed per agent's action a.

production activity if they possess the appropriate skills and if they have access to the required inputs. Initially the agents are using the surplus of stocks present at the start of the simulation and no manufacturing takes place. Once stocks are depleted, one by one the different sectors and thus the agents that compose the sectors have to come into action (see Figure 3). The figure shows the number of agents per iteration that are involved (either heavily or just a little) in a particular production process.

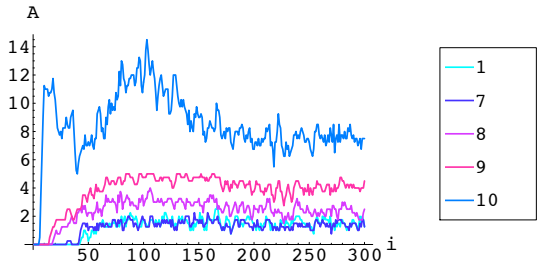


Figure 3: Number of agents A per iteration i participating in the production of the different commodities of Figure 1. The numbers indicating the different production processes correspond to the columns of the input and output matrices, such as in Table 3.

The peak in the generation of labour (sector 5) around iteration 100 can be explained by the absence of an appropriate record of past activity at start up. First the surplus is consumed and none of these goods have to be generated. When the system begins to run out of intermediates these have to be produced, but not even the stocks to do so are there since appropriate stock levels depend on the past activity levels. With a short period of higher activity the system is able to catch up.

New Products, Waves of Innovation, Evolutionary Dynamics

The actual innovation process of the evolving economy is composed of two steps: first the technology matrices have to be expanded to include the production of the new commodity or the new use of existing products. Subsequently agents have to start using the new technology.

At random times, new technology is generated. With a certain probability this new technology involves the creation of new consumables. If not a consumable, the new technology aims at producing products involved in the existing production process, be it capital or intermediates. In order to make something from which the system can benefit, we introduce new technology (i.e. a tool and the skill to use it) to produce something of which there is currently a shortage, where shortage is defined as in the price mechanism. Once one of the products in short supply is identified, the input for the new tool t can be composed. More precisely, when the product in excess demand p is factorized $p = f_1^{q_1} \cdot f_2^{q_2} \cdot \ldots \cdot f_n^{q_n}$ for some n and $q_. \in \mathbb{N}$, it is clear which factors are required in the input. A random list of products $i_1, \ldots, i_m$ that contain those factors is generated such that the product of these inputs $i = i_1 \cdot i_2 \cdot \ldots \cdot i_m$ will be divisible by the sought after product p, thus $\frac{i}{p} = g \in \mathbb{N}$. There are then two possibilities: the tool t sets the production of the required product p based on the new input i, or the tool requires the different parts $i_1 \cdot i_2 \cdot \ldots \cdot i_m$ and uses these without product i first being assembled. g is considered garbage and ignored.

The entry of new technology is based on Bruckner *et al.* (1989), who describe a general model to study evolutionary processes. The model consists of a countable set of fields $F = \{F_1, F_2, \ldots\}$ with, for each field F_i, a number N_i indicating the number of elements in the field. The state of the whole system is given by the set of occupancy numbers $\{N_1, N_2, \ldots\}$. Changes in occupancy numbers are discrete and occur in the smallest steps possible; fields gain or lose one element. The probabilities of these changes depend on the current state of the system. Bruckner *et al.* show that the Markov process of interactions between fields is capable of generating a wide variety of evolutionary dynamics. In particular, they write that this type of model is capable of simulating the dynamics of evolutionary systems, including the dynamics of technological evolution. A later paper experiments with the parameters for the stochastic economic substitution model and show that realistic substitution dynamics can be obtained (Bruckner *et al.*, 1996).

Innovation by new agent: When new technology becomes available, its establishment is affected by some of the existing technologies. Parameter A_{ij} describes the inclination of technology j to establish technology i by means of a new agent: $W(N_i + 1, N_j | N_i = 0, N_j) = A_{ij} N_j$

Innovation by existing agent: When an agent expands its skills, be it with a new technology or an existing technology, the agent innovates its production process. The choice of additional technology is affected by the existing technologies. Parameter M_{ij} describes the inclination of technology j to establish technology i by means of an existing agent: $W(N_i + 1, N_j | N_j) = M_{ij} N_j$

Growth of existing technology by spontaneous new agent: An increase in the number of agents using technology i independent of the state of the system: $W(N_i + 1) = \phi_i$

Growth of existing technology by new agent: Increase in the number of agents using technology i due to self-reproduction or sponsoring by technology j: $W(N_i + 1, N_j|N_i, N_j) = A_i^{(0)} N_i + A_i^{(1)} N_i N_i + B_{ij} N_i N_j$

Replacement of technology by existing agent: Agents imitate the successful technologies of other agents, and subsequently use these to replace less successful skills. The parameters A_{ij} represent a measure of success and failure. Furthermore, the probability is influenced by the current size of the the technology fields. The larger the field, the more occurrences of the technology in question, and the greater the probability of replacement: $W(N_i + 1, N_j - 1|N_i, N_j) = A_{ij}^{(0)} N_j + A_{ij}^{(1)} N_i N_j$

Technologies that are not used for a specified length of time are forgotten by the agent, and when all agents lack a specific technology, this technology can be removed from the model. The same applies to products that have become obsolete. Unlike the random replacement of functionality in Jain and Krishna (1999), here skills and products slowly disappear.

The following figures illustrate the results of this process in a typical run. At random times new commodities and/or new production techniques are added to the technology matrices, and subsequently probabilistic rules distribute new and existing skills over the agents. The simulation begins with a small economy such as was illustrated in the previous section, and as time progresses the model constructs novel functionality and elements, analogous to a constructive artificial chemistry. After a 1000 or so iterations the economy consisted of 35 products and 71 production processes.

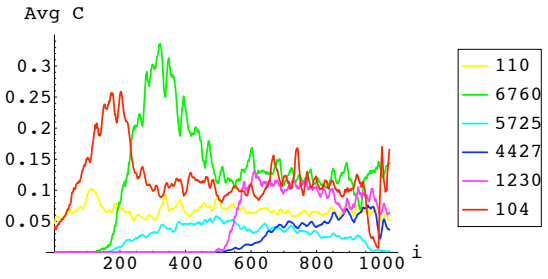


Figure 4: Product concentrations of a selected number of products averaged over a gliding window of length 10 to smooth the curves. Only the first four digits of the product appear in the legend.

Figure 4 illustrates the product concentrations of a selected number of products. Some products are present at the start of the simulations (110 and 104), other products appear later. Product 6760 is a consumable that becomes a serious competitor to the initial consumable, commodity 104. Later it diminishes its importance as other consumables become available. Product 57257200 is only temporarily successful and disappears from the stage. Other new products seem, at least for now, to be adopted more permanently.

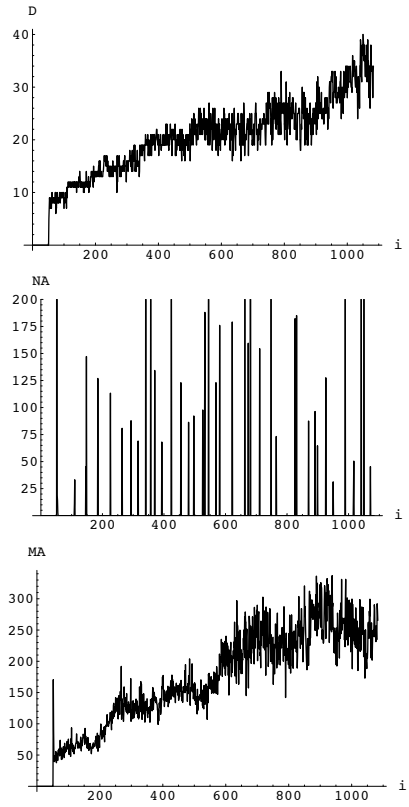


Figure 5: Bedau's measures applied to the economic activity of the present model. The top graph displays the number of active technologies (diversity D), the graph in the middle shows the activity of new technology "NA" expressed in value of output at first application, and the bottom graph displays the mean activity "MA" per active technology .

The innovations introduced to the system by the agents are not without effect. This can be illustrated by a measure developed in Bedau *et al*. (1997). That paper describes a system to classify evolutionary dynamics based on the increase of diversity and the effect the increased diversity has on the average productivity of the system. This measure is capable of distinguishing between a system with truly beneficial adaptive behaviour and a system with merely an increase in diversity. The first is characterized by an increase in average productivity as diversity is non-decreasing, while the latter system displays bounded diversity combined with bounded average productivity. For the application of this measure here, average productivity is defined as the total value of output divided by the number of technologies re-

quired to generate the output, and the first results suggest that the artificial economy displays an increased diversity in terms of technology used, while at the same time the average productivity increases. Therefore the system classifies as one with unbounded evolutionary activity (see Figure 5).

We conclude with an illustration of the competition between technologies that produce a single commodity (Figure 6). It concerns a newly introduced commodity, and it is quite successful as a whole row of innovations is triggered by its introduction. 5 alternative production processes jump the band wagon. As in the real economy market shares are far from stable and the most efficient production technique does not necessarily become dominant.

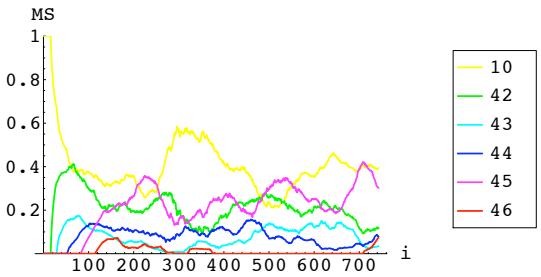


Figure 6: The market share of different technologies to produce one single product (product 7741173440000 in this case) coexist and compete for the market. The graph shows the proportion of the total production of the commodity produced by each of the 6 production techniques. The numbers on the right correspond to the column numbers in the technology matrices.

Conclusions and Perspectives

Though there are still a number of issues that need improving or further exploration, the framework developed in this paper is one with many merits. The main virtue is that this framework demonstrates that it is possible to treat the economic system as a constructive dynamic system, in other words, as a system undergoing continual structural evolution. The framework developed here is capable of dealing with endogenous change, something that is very important in our economy, but that until now has been essentially ignored in research. The application of a constructive dynamic system to the field of economics opens up new opportunities in economics.

In addition, the application of the Bedau measure suggests that the evolutionary economic dynamics generated by the model is appropriately named evolutionary.

The framework is also very useful to complement existing research in evolutionary economics. For example, descriptive approaches on the entry and exit of new enterprises can be assisted by an experiment that can indicate what parameters are responsible for the observed patterns. The model can shed light on why market shares and market size can fluctuate so wildly, and why in the production of one single good there are a variety of technologies being applied.

The result is a broad, abstract framework applicable to the study of evolving economic systems. It has the potential to elucidate many aspects of our economy, from dispersal of technology, to location strategies, to pricing, to the development of higher level organisation.

Acknowledgments

This work was supported by SSHRC grant No. 410-2004-1843 to R. White.

References

Banâtre, J.-P., Coutant, A., and Metayer, D. L. (1988). A parallel machine for multiset transformation and its programming style. *Future Generation Computer Systems*, 4(2):133–144.

Banzhaf, W., Dittrich, P., and Rauhe, H. (1996). Emergent computation by catalytic reactions. *Nanotechnology*, 7(1):307–314.

Bedau, M. A., Snyder, E., Brown, C. T., and Packard, N. H. (1997). A comparison of evolutionary activity in artificial evolving systems and in the biosphere. In *Proceedings of the fourth European Conference on Artificial life*, pages 125–134.

Blatt, J. M. (1983). *Dynamic Economic Systems*. M.E. Sharpe, Inc., New York.

Bruckner, E., Ebeling, W., Jiménez Montano, M., and Scharnhorst, A. (1996). Nonlinear stochastic effects of substitution - an evolutionary approach. *Journal of Evolutionary Economics*, 6:1–30.

Bruckner, E., Ebeling, W., and Scharnhorst, A. (1989). Stochastic dynamics of instabilities in evolutionary systems. *System Dynamics Review*, 5(2):176–191.

Dittrich, P., Ziegler, J., and Banzhaf, W. (2001). Artificial Chemistries A Review. *Artificial Life*, 7:225–275.

Herriot, J. and Sawhill, B. (Accessed May 9, 2008). Web-Economy as a self-organizing information system. http://www.redfish.com/research/webEconomyIdeas/.

Jain, S. and Krishna, S. (1999). Emergence and growth of complex networks in adaptive systems. *Computer Physics Comm.*, 121-122:116–121.

Metcalfe, J. S. (1998). *Evolutionary economics and Creative destruction*. Routledge, London, New York.

Schumpeter, J. A. (1939). *Business Cycles: A theoretical, historical and statistical Analysis of the Capitalist Process*. MacGraw-Hill.

Tesfatsion, L. (2007). *Handbook of computational economics. Agent-based computational economics*, chapter Agent-based computational economics: A constructive approach to economic theory. Elsevier, Amsterdam.

von Neumann, J. (1945-1946). A model of general economic equilibrium. *Review of economic studies*, 13(1):1–9.

Witt, U. (2006). Evolutionary economics. Technical Report 0605, MPI of economics.

Homeodynamics in the Game of Life

Keisuke Suzuki and Takashi Ikegami

General Systems Sciences,
The Graduate School of Arts and Sciences,
The University of Tokyo,
3-8-1 Komaba, Tokyo, 153-8902, Japan

Abstract

In this paper we study the emergence of homeodynamics and adaptation in a two-layer system of the Game of Life in which the Game of Life in the first layer couples with another cellular automata system in the second layer. Homeodynamics is defined here as a space-time dynamic that regulates the number of cells in state-1 in the Game of Life layer. A genetic algorithm is used here to evolve the rules of the second layer to control the pattern of the Game of Life. We discovered that there are two antagonistic attractors that control the numbers of cells in state-1 in the first layer. The homeodynamics sustained by these attractors are compared with the homeostatic dynamics observed in Daisy world.

Introduction

Living systems require a stable and sustainable structure on top of unstable and highly chaotic open environments. The maintenance of such a structure is called "homeostasis", as named by Cannon (1932), and became one of the central themes in Cybernetic studies(Wiener, 1948). Several mechanisms underlying homeostasis have been proposed and they have become a guiding principle of our everyday technology. For example, positive/negative feedback loops and afferent/efferent copies are well studied and developed.

The study of homeostasis has revealed those mechanisms, but they are often introduced as a controlling device and the evolution of homeostasis itself has not been discussed seriously. People continue to study ecological homeostasis, in particular after Lovelock (1972) proposed his Gaia hypothesis. The Gaia hypothesis posits that the complex and global network of living/nonliving systems we observe self-organizes into homeostatic states. The Gaia hypothesis has been theoretically examined by Watson and Lovelock (1983) by developing the Daisy world model, a simple implementation of the Gaia theory. In Daisy world, temperature should be sustained at a certain range independent of the environmental temperature. Harvey (2004) calls the mechanism underlying the Daisy world a "rein control," a controlling mechanism which serves to pull the temperature toward the viability zone.

What has been missing thus far in the study of Daisy world is the self-organizing and dynamic nature of homeostasis. Ikegami and Suzuki (2008) studied a dynamic version of Daisy world controlled by spatio-temporal chaos. Because the homeostasis here is dynamically sustained, we refer to this as homeodynamics. Moreover, Homeodynamics doesn't simply hold the average temperature constant, as in a conventional Daisy world simulation, but instead aims to keep the temperature variation around the average. Holding variation brings adaptability into the Homeodynamic system, as it can respond to novel environmental conditions. This is the most significant characteristic of homeodynamics, which we will also focus on in this study.

With respect to this adaptability of homeo-systems, Ashby (1960) proposed an interesting design principle for the brain and for life forms as a whole which was mainly driven by homeostasis. He posited that the adaptive behavior of life is only an outcome of homeostatic properties and proposed a different type of homeostatic system called an ultra-stable system. This new system has two feedback loops. The primary feedback loop is driven by a mutual interaction between an organism's complex sensory and motor channels and the environment. Another feedback loop develops from the interaction between viability constraints and the relevant reacting parts via the essential variables that control the reacting parts. Usually, the second feedback loop is intended to change the meta-parameters of the system. When parameter values are outside of the viability constraints, the second feedback loop adjusts the essential parameters to let the system move towards a more stable state. These characteristics of the ultra-stable system share common features with our homeodynamic systems. That is, two dynamics co-exist in the same system with different time scales and they cooperatively control the homeostasis by keeping sufficient fluctuations in the system. In other words, we need both stable and unstable dynamics to develop homeostasis and adaptation at the same time.

In this paper, we study the notion of homeodynamics and adaptation by using Conway's Game of Life. A major drawback of most homeostatic models, including ours, is that

many systems can be too stable in the sense that they can survive without paying significant costs (or in the other word a system never dies). Therefore our challenge is to see how homeostasis can emerge even in a very unstable world, as in the Game of Life. A second objective of the paper is to see how robust homeostatic behavior is balanced with purposeful behavior such as memorizing the initial states. Robust homeostasis (keeping the system's state density constant) can be achieved by making a system insensitive to the initial Life density. But memorizing the initial state density means that a system should become sensitive to the initial Life patterns. These two opposite properties must be balanced within the same system.

In the next section we describe how to use the Game of Life to study homeostasis. In §3, we describe and analyze the results, and in §4, we discuss the observed characteristics of homeodynamics and adaptation in the Game of Life and also attempt to apply these results to the more general notions of homeodynamics.

The Model

The basic idea of the model is inspired from work by Taylor (2004). In his model, the system under examination consists of two layers of cellular automata: one is the Game of Life, and the other layer serves to control the Game of Life pattern. The dominating cell layer does not have to be governed by the rules of Life, but instead can be driven by a different rule set. Taylor evolved the rules of this layer by using an evolutionary algorithm to control a virtual sensorimotor flow arranged on the first layer. For example, when an input bit on the first layer is state 1, the output (target) area should have many state-1 cells. This contingency between input and output bits is mediated by the intermediate area, in which some of the bits are governed by the rules in the second layer.

The purpose of Tayler's study is to examine an unseparated body-environment boundary and to see the emergence of the boundary itself. We will also investigate this point, but here we will use his approach to study homeostasis. Our setup is described below.

The model consists of a 2D cellular automata running the Game of Life. Extra rules encoded in the genome can override Life states in a certain part of the CA space. A *target area* is also designated, which is significant for the central tasks of the simulation.

In Taylor's work there are two kinds of rules for genes: conditional and temporal genes. The conditional gene is activated when a certain requirement of neighboring cells is satisfied. The temporal gene is activated when a certain time step passes. In both cases, the genes have a coordinate specifying the target cell.

Here we used only the conditional rule. We modified the rule to become a so-called "totalistic rule" in which the gene only takes into account the number of neighboring state-1 cells, and not any specific neighboring patterns. Figure 1 illustrates the gene components we used. Note that in our model, all the genes are activated every time step, due to the construction of the genome and the use of this totalistic rule. The gene consists of a 26-bit binary string, 18 bits of which represent a totalistic rule. As is usually the case, the 2D CA rule is specified by the pair of numbers Bx/Sy, which specifies when to change the cell's state to 1, whether its own state is 1 or 0, respectively. For example, the Game of Life is represented by B3/S23. We use 9 bits to represent those two parameters.

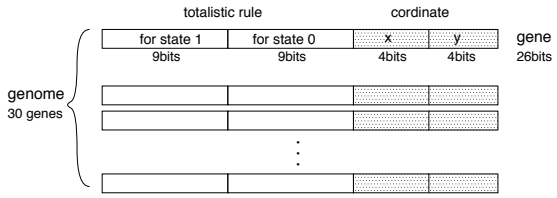


Figure 1: A gene component consists of a 26-bit binary string, 18 bits of which encode the totalistic CA rules. The remaining bits encode the coordinate of the site where the rule activates. A group of these genes is called the genome. The simulation is run using Game of Life dynamics and one genome.

The remaining 8 bits encode the spatial position of the site controlled by the second layer (4 bits for each x and y coordinate). The length of each gene is fixed and each gene specifies a particular single site in the 16×16 cell space.

The total cell space is given as a square of the size 40×40 and the intermediate area controlled by those genes is given as a square of the size 16×16. The target area is defined as a square space and all three squares share a common center. The size of the target area is 32×32 bits and includes the intermediate area (See Fig. 2). The boundary of the space is always set to state-0.

The target behavior of the model

While evolving the rule set of the intermediate cell space we will study homeostatic behaviors observed in the Game of Life. Instead of a temperature value as in Daisy world, we use the density of cells in the state-1 as the target variable to keep constant.

In Daisy world, the system consists of two types of flowers, black and white daisies, with their local temperature values. The black daisies increase the temperature and the white daisies decrease the temperature. By setting the growth rate of these flowers according to the local temperature, both flowers show positive feedback effects in their relationship to changing local temperatures. While the black daisies increase both the temperature and their population, the white daisies decrease the temperature and increase their population. Loosely linked by the the two local tempera-



Figure 2: The space used for the simulation. Filled squares represent the area governed by Game of Life rules. Lined squares represent the area which the genome specifies. The inner square shows the area in which genes override the Life states, and the outer square depicts the target area. The dynamics is only evaluated in this target area.

ture, the global temperature is sustained constantly while both population of daisies change according to environmental temperature changes.

This result shows that the homeostatic behavior does not result from insensitivity to the environmental stimulus, but is actively achieved by an adaptive coupling of components which are sensitive to the environment.

In order to observe the underlying dynamics of homeostasis in the Game of Life, we constructed three different tasks.

task A Sustain the same density of the target area regardless of the initial Life pattern density.

task B Control the density of the target area, making it proportional to the given initial Life pattern density.

task C Control the density of the target area, making it inversely proportional to the given initial Life pattern density.

The first task is designed to directly aid the development of a homeostatic behavior in the Game of Life patterns. We will see how this behavior is achieved in the Game of Life space. The second and third tasks are intended to facilitate the development of sensitivity to the given conditions. These target behaviors in task B and C are not directly connected to homeostatic behavior, but might be connected to the property of adaptability in homeostasis.

GA

In order to observe those behaviors, the CA rules encoded in the genes are evolved by a simple genetic algorithm (GA). We prepared 30 genomes in a population, each of which consists of 30 genes, which specify each spatial location and the rule content of the 30 CA rules in the intermediate area. Mutation occurs at every site of the gene at a certain rate per bit, which modifies the spatial locations and the rule content. The number of genes also changes during this process, so the number of CA sites in the intermediate area also varies. For the selection algorithm, we used a roulette selection procedure and chose an elite strategy.

Parameter	Values
population	30
the mutation rate	0.05
the crossover rate	0.01
the mutation rate for genome length	0.01
the number of elite	5
initial density(higher)	0.5
initial density(lower)	0.0
evaluated duration(time steps)	500

Table 1: The parameters used in our simulation

The GA goes through both an evolving phase and a testing phase. In the evolving phase, 30 genomes are evolved as a unit against two different initial states with lower and higher density patterns. Here the lower density is set to 0 and the higher density is set to 0.5.

Fitness of the genome is calculated by finding how the average density of state-1 within the target area compares to the specified *target densities*. The target densities in the three tasks are set as follows.

$$\begin{aligned} 0.5 \quad & (\text{for task A}) \\ d_I \quad & (\text{for task B}) \\ 1 - d_I \quad & (\text{for task C}) \end{aligned}$$

Here, d_I is the intial density of state-1 cells, set as either 0 or 0.5.

Note that we only use a fixed random initial pattern for all evolutionary processes. Once the system evolves, it develops a sufficient generalization capability; the system can do well with new initial patterns. However, a full generalization capability is difficult to obtain. We will revisit this point later. Table 1 shows the parameter values used in this experiment.

Result

Evolved Dynamics

In each of three tasks, the genomes in our population were evolved for higher fitness. Figure 3 show the temporal changes of the state-1 density of the fittest genome in each task.

Two lines on the graphs are shown, one for the low initial density case (null pattern) and another for the high density case (0.5). These density values were used in the GA dynamics.

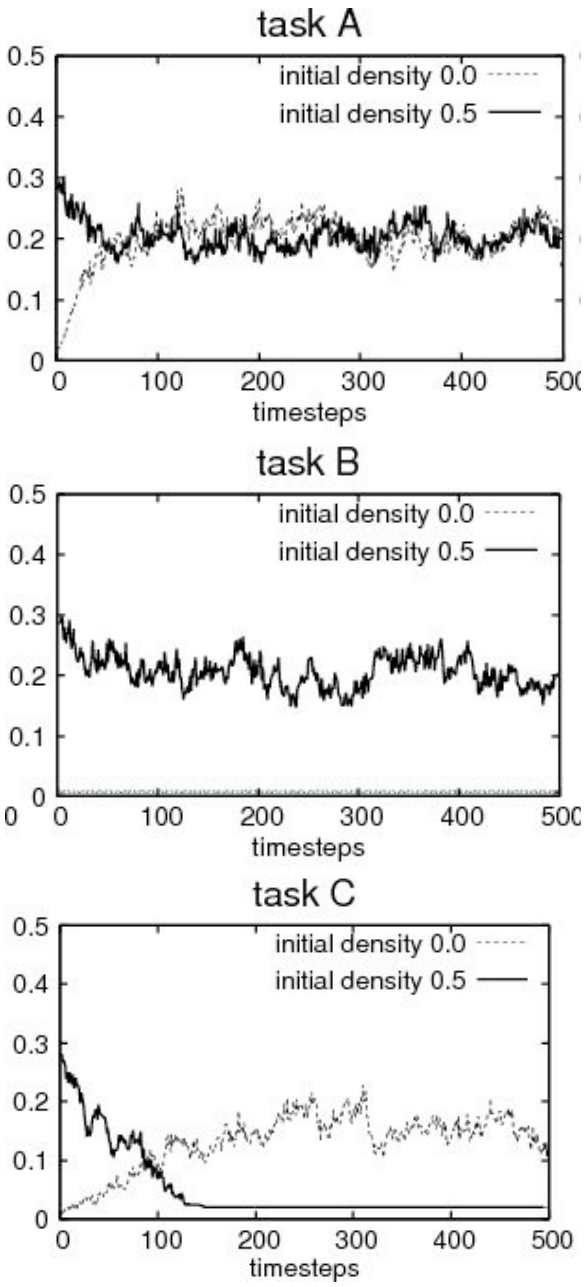


Figure 3: Temporal changes of the state-1 density of the fittest genome in each task. The initial Life patterns used here are the same as those used during the GA procedure. When begun with both low and high density initial states, the state-1 densities are maintained at around 0.2(task A). With the higher-density initial state, the state-1 density is kept around 0.2, but with the lower-density initial state, the state-1 density decreases to 0(task B). The result observed here is the inverse of task B. The higher-density initial state shows a state-1 density which drops almost to 0, but the lower-density initial state causes a growth in state-1 density until the density reaches approximately 0.15(task C).

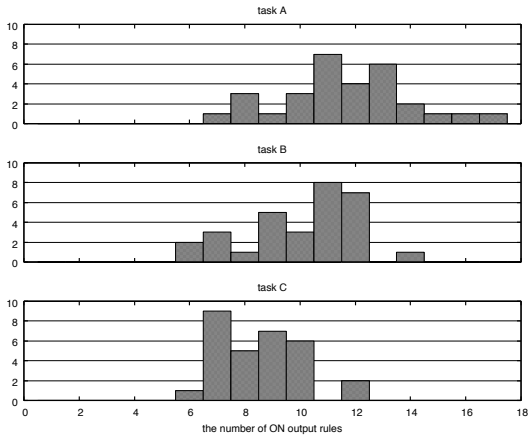


Figure 4: Histograms of the number of state-1 outputs out of the overall outputs for the fittest genomes in each of the three tasks. Task A and B genomes have a biased distribution towards larger values, while task C is biased in the opposite direction.

In task A, the density almost always reaches the same value of approximately 0.2 regardless of the initial density. We can see here that this attracting state is maintained by the generators of the Game of Life pattern, which will be discussed later. Cloud-like patterns in the Life space are generated by the evolved CAs.

In task B, the initial low-density state almost always creates a sparse pattern in the target area. Using the higher initial density state, the average resultant density state tends to fluctuate around a value of 0.2. The genomes generally increase the density if the target area is surrounded by a high-density pattern; the density tends to decrease in the sparse case. A generator of this type creates cloud-like patterns in the higher-density environment, but suppresses cloud-like patterns in the lower-density environment.

In task C, the densities are altered to be inversely proportional to the initial states. When the initial state has a low density, the evolved CA creates a high density state. In contrast, when presented with the high-density initial state, the density is decreased until all Life patterns are diminished. In the higher-density initial state, we expect an increase in the resultant density. The evolved CAs have to inhibit the spontaneous generator to decrease the state-1 density. Thus, the genome in task C behaves like an activator in the initial low-density state, but behaves like an inhibitor in the initial high-density state.

CA rules of the second layer

Each of the 30 sites in the second layer has a different CA ruleset. One way to characterize them is to compute the number of state-1 outputs as a fraction of overall outputs (e.g. the Game of Life has 3/16). Figure 4 shows a histogram comparison of these state-1 outputs for the evolved

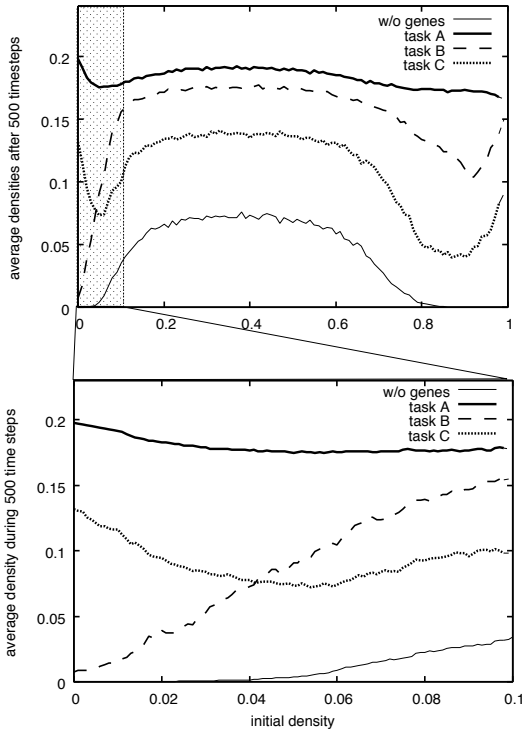


Figure 5: The average densities observed during 500 time steps when starting from different initial densities. The evolved genomes in task A, B, C and runs in which only Game of Life rules exist are compared. Each value is averaged over 100 different initial Life patterns.

genome sets.

Before evolving genome sets, the output is normally distributed around 8. The rulesets of tasks A and B are biased toward higher state-1 outputs, but those of task A are more biased towards larger values than those of task B. In task C, the rule sets are biased in the opposite direction.

In task A, the genome generates a cloud-like pattern regardless of the initial pattern. The generator is a strong activator of the state-1 LIFE pattern. Task B (proportion) also generates a cloud-like pattern, but only in the higher-density initial condition. The decrease of the density in the lower initial pattern may require a weaker activator in this task than that in task A. In task C (inverse proportion), the genome has to inhibit the cloud-like patterns in the higher-density initial state and activate the cloud patterns in the lower-density initial states.

The genomes in task B and C can be compared with the black and white daisies of Daisy world. The genome in task B has a tendency to output state-1, which can be regarded as similar to the black daisy which increases the local temperature. Likewise, the genome in task C has a tendency to suppress state-1, which can be regarded as similar to the white daisy which decreases the local temperature. How-

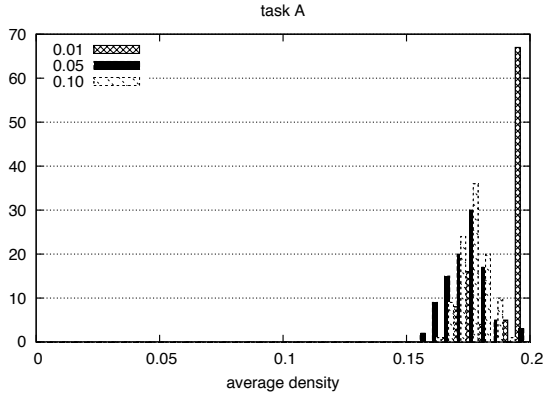


Figure 6: Histogram of the average densities observed from the task A genome in 100 samples with different initial Life patterns. Here, initial densities of 0.01, 0.05, and 0.10 are used.

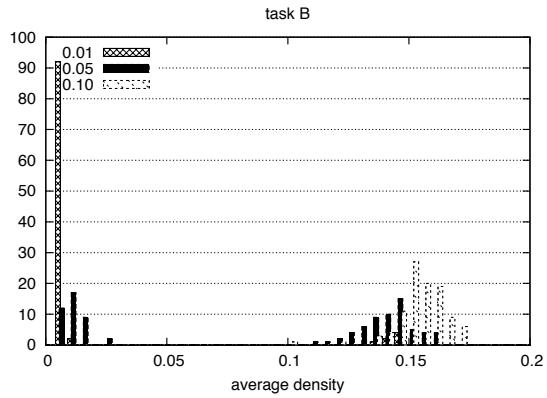


Figure 7: Histogram of the average densities observed from the task B genome in 100 samples with different initial Life patterns. Here, initial densities of 0.01, 0.05, and 0.10 are used.

ever, it should be noted that the behavior of the evolved CA rule is also affected by their spatial configuration. Thus, the state-1 output levels do not fully reflect how many cloud-like patterns these evolved CA rules can make by themselves.

Generalization

While the genomes under discussion here have been trained with only two fixed initial Life patterns, the final genome acquired a generalization capability to some extent. After training the genome set with these two different initial densities of 0.1 and 0.5, we have tested the evolved genome against other initial densities. Figure 5 shows the average density after 500 GA time steps for the given initial density. Each density is averaged over 100 different initial Life patterns of a given state-1 density. For a comparison, we also show the average density obtained with only the origi-

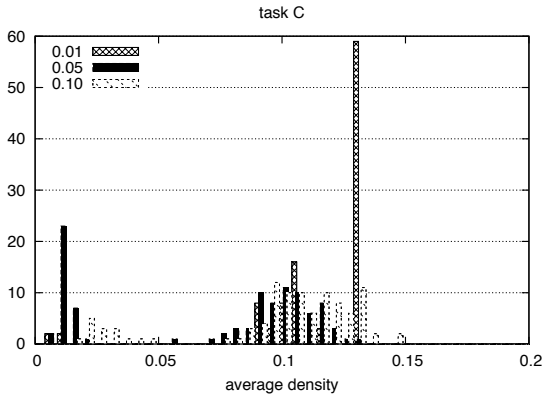


Figure 8: Histogram of the average densities observed from the task C genome in 100 samples with different initial Life patterns. Here, initial densities of 0.01, 0.05, and 0.10 are used.

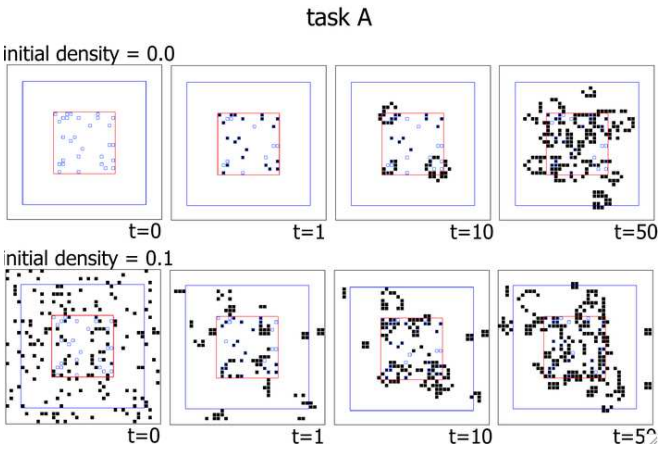


Figure 9: Snapshots of the Life dynamics of the fittest genome in task A. The top columns show the results when the initial density is zero, whereas in the bottom column the initial density is 0.1. In both cases, clouded Life patterns spread out from the genome area.

nal Life dynamics, without any second layer.

The genome evolved by task A achieves almost identical values of high average density against a wide range of initial densities from 0.0 to 1.0. Compared with the original Game of Life, this evolved genome sustains a higher density, particularly around the lowest and highest initial state. This is achieved by a state-1 generator which increases the state-1 density regardless of the initial states. However, similar behavior can be seen when one adds noise to the Game of Life. So the genome does not regulate state-1 density, but rather works as a "random generator".

We are not certain how many attractors this two-layered system has when starting from identical initial density states. Fig. 6 shows that there are two peaks in the histogram which correspond to different attractors in the system. Regardless of the final attractor, the genome evolved via task A has similar final density states, i.e. 0.15 -0.19.

Figure 9 shows snapshots of the fittest genome found in task A with the two different initial densities. In both cases, the Game of Life patterns become similar cloud-like shapes after time passes.

Tasks B and C also inherit the tendency to increase and decrease the state-1 density from the condition in which evolution occurs.

When the initial density is small(< 0.1), the genome evolved via task B shows a linear dependence on that initial density, while the genome evolved via task C shows the inverse dependency. The genome from task B develops a state-1 generator proportional to the increase in the initial Life-pattern density, but it saturates in higher densities. Also, the evolved genome from task C has the capability to suppress the density of the initial Life pattern if it is high. However, such suppression is competing with the original Game of Life and thus is only effective for lower initial density cases.

Concerning the histograms in Fig. 7, we notice that there are two attractors induced by the second layer: one associated with very low-density states and the other associated with states around a density of 0.15. In fig. 8, there are also two attractors created by the genome activity: one near to a density value of 0 and the other near to a density value around 0.1.

How are these attractors chosen from the initial densities? Figure 10 and 11 are snapshots of genomes evolved from tasks B and C in the two initial density conditions. They reveal the appearance of the two attractors due to a subtle difference found in the starting configuration. For example, in Fig. 10, the two snapshots at t=1 have only a few bits which differ from each other, yet the higher-density initial pattern only makes state-1 clouds, thus creating a high-density resultant pattern.

This occurs because in the task B genome, the state-1 generator is only activated by few state-1 cells initially located in the space. Without these cells, the genome does not activate this generator and almost all the cells remain at state-0. Higher initial densities increase the probability that these cells become state-1. So the resultant densities are proportional to the initial density.

Similarly, the task C genome has a state-1 generator which is only activated when initial density is low. However, when only a few specific cells are state-1, the generator stops creating cloud-like patterns. Higher initial densities increase the probability of this event. Consequently, the resultant densities are inversely proportional to the initial density. When the initial density is higher than 0.1, the genome makes another spontaneous generator derived from the Game of Life, resulting in higher densities in a similar

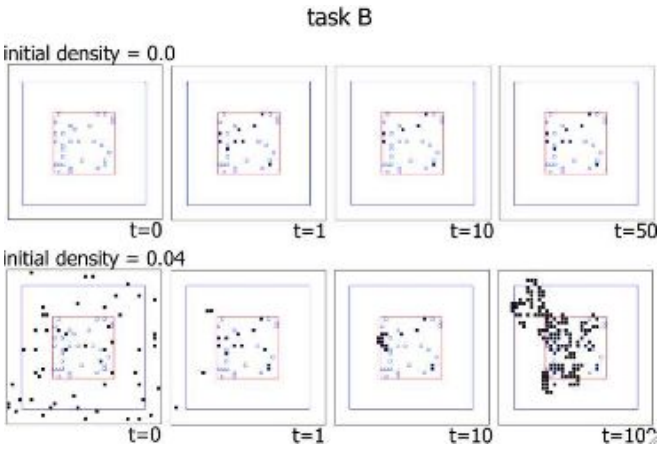


Figure 10: Snapshots of the Life dynamics of the fittest genome in task B. The top columns show the results when the initial density is zero, whereas in the bottom column the initial density is 0.04. In the zero-density condition, the Life pattern stays almost at zero activity. However, in the higher-density condition, a clouded pattern of state-1 cells emerges after t=10.

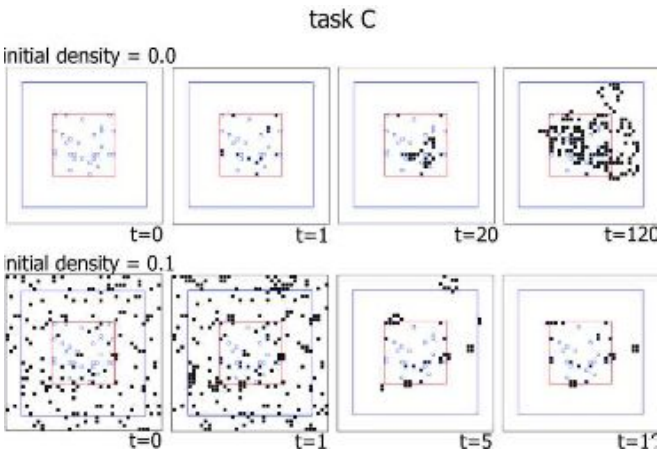


Figure 11: Snapshots of the Life dynamics of the fittest genome in task C. The top columns show the results when the initial density is zero, whereas in the bottom column the initial density is 0.1. In the zero-density condition, a clouded Life pattern emerges, but in the 0.1 condition, the initial state-1 cells eventually disappear.

manner to task B.

Discussion

In this paper, we studied homeostasis and adaptability with respect to cell states in the Game of Life as opposed to temperature as in Daisy world. There are some lessons to be learned here, both about homeostasis and the effect of noise.

The CA rule evolved in task A has the capability to generate a state-1 density regardless of the initial Life density. The behavior of this rule can be compared to a "random generator" which randomly updates the cell states at a certain probability. In task A, the evolved CA ruleset mimics a random generator using a deterministic rule. In tasks B and C there also have evolved generators of state-1 cells, but they are more sensitive to the initial state density.

Since we do not have any external noise in this simulation, it can be compared to deterministic chaos in continuous-state dynamical systems. Comparing the upper and lower figures in Fig. 10, we notice that almost identical initial Life patterns lead to different attractors. Such sensitivity to minute differences is also reminiscent of chaotic dynamics. Now we will discuss the nature of those two attractors in greater detail.

The initial Life states are evolved into either lower-density or higher-density states. Those two types of attractors are controlled by the evolved CA rules. We assume that the evolved CA ruleset may have both activators and inhibitors. Some CA rules tend to generate more state-1 Life patterns than the others and increase the state-1 density as the result, which we call "activator" rulesets. In contrast, some CA rules show the opposite behavior and lower the state-1 density, which we call "inhibitor" rulesets.

Daisy World	**Our model**
temperature	state-1 density
black daisy	activator CA
white daisy	inhibitor CA
growth rate as a function of the temperature	generators as a function of LIFE pattern

Table 2: A comparative chart between Daisy world model and the present LIFE game model

These two opposite behaviors remind us of the black and white daisies in Daisy World. Both these rulesets and those daisies can cooperatively make homeostatic states. Because black and white daisies have opposite responses to the sunlight, they can self-regulate the temperature by tuning their population size. If there are more black daisies, the temperature goes up as the average albedo gets lower, whereas if there are more white daisies, the temperature goes down due to the higher albedo value.

This simple scenario is also realized in the present Life game system. We simply take the activator rulesets as black daisies and the inhibitor rulesets as white daisies. The correspondence between Daisy world and this Game of Life system is shown in Table 2.

In Daisy world, the growth rates of daisies are determined by the local temperatures. The concept of temperature is not implemented in our system explicitly. Instead, local Life patterns determine the behavior of the CA rule sets. Note that the equivalent of Daisy world's local temperature in

our system is not just a one-dimensional variable which explicitly specifies the growth rate of state-1, but is instead a spatio-temporal pattern which drives responses from the evolved CA rulesets. This dynamical property of the Daisy World has also been discussed in our previous model of the mobile daisy agent(Ikegami and Suzuki, 2008).

Ideally, task A should be achieved by coupling the evolved CA rules in task B and C, however, we have not completed that task in this paper. Detailed analysis of that work will be reported in a follow-up to this research.

Instead of using chaos-like attractor in this study, it may be interesting to use unique Game of Life creatures such as oscillators, breeders and guns to generate homeostasis in more complex ways. We might then expect to see alternative homeodynamic mechanisms which are very different from those seen in Daisy world.

Acknowledgments

We thank Dr. Eric Silverman for his intense proof-reading of the manuscript. This work is partially supported by Grant-in aid (No. 19300104)fromtheMinistryofEducation, Science, Sports andCulture

References

Ashby, W. (1960). *Design for a Brain: The Origin of Adaptive Behaviour*. Chapman and Hall, London.

Cannon, W. B. (1932). *The wisdom of the body*. W.W. Norton and Company, New York.

Harvey, I. (2004). Homeostasis and rein control: From daisyworld to active perception. In Pollack, J., Bedau, M., Husbands, P., Ikegami, T., and Watson, R. A., editors, *Artificial Life IX : Proceeding of the 9th International Conference on the Simulation and Synthesis of Living Systems*, pages 309–314. Cambridge, MA: MIT Press.

Ikegami, T. and Suzuki, K. (2008). From homeostatic to homeo dynamic self. *BioSystems*, 91(2):388–400.

Lovelock, J. E. (1972). Gaia as seen through the atmosphere. *Atmos. Environ.*, 6:579–580.

Taylor, T. (2004). Redrawing the boundary between organism and environment. In Pollack, J., Bedau, M., Husbands, P., Ikegami, T., and Watson, R. A., editors, *Artificial Life IX : Proceeding of the 9th International Conference on the Simulation and Synthesis of Living Systems*, pages 268–273. Cambridge, MA: MIT Press.

Watson, A. J. and Lovelock, J. E. (1983). Biological homeostasis of the global environment: the parable of daisyworld. *Tellus*, 35B:284–289.

Wiener, N. (1948). *Cybernetics or Control and Communication in the Animal and the Machine*. Paris: Hermann et Cie - MIT Press, Cambridge, MA.

How Learning Can Guide Evolution of Communication

Reiji Suzuki and Takaya Arita
Graduate School of Information Science, Nagoya University,
Furo-cho, Chikusa-ku, Nagoya 464-8601, Japan
{reiji, arita}@nagoya-u.jp

Abstract

The Baldwin effect is known as a possible scenario of the genetic acquisition process of a learned trait without the Lamarckian mechanism. However, it is still controversial how learning can facilitate evolution in dynamically changing environments caused by internal factors. Our purpose is to clarify whether and how leaning can facilitate evolution in dynamic environments which arise from communicative interactions among individuals. We constructed a simple computational model for the evolution of communication ability and its phenotypic plasticity. In the model, the levels of adaptive communication, which correspond to the expected fitness value when the communication results in success, of signalling and receiving processes are determined by different sets of traits under the assumption of the correlation between their fitness and the effects of epistatic interactions among traits. A communication is successful only when the levels of the signaller and the receiver are the same, and the individuals try to improve their communication levels through the learning process in which the values of plastic traits can be modified from their genetically determined values. The evolutionary experiments clearly showed that the Baldwin effect repeatedly occurred and facilitated the adaptive evolution of communication in this type of dynamic environments.

Introduction

The Baldwin effect (Baldwin, 1896, 1902) and the role of phenotypic plasticity in evolution have been drawing much attention in evolutionary studies (West-Eberhard, 2003; Crispo, 2007). The Baldwin effect is typically interpreted as a two-step evolution of the genetic acquisition of a learned trait without the Lamarckian mechanism: individuals that have successfully adapted their own trait to the environment through their lifetime learning processes occupy the population (1st step), and then the evolutionary path finds the innate trait that can replace the learned trait (2nd step) because of the cost of learning (Turney et al., 1996; Maynard-Smith, 1987). The second step is also known as genetic assimilation (Waddington, 1953), or a kind of genetic accommodation (West-Eberhard, 2003; Crispo, 2007).

Since the study by Hinton and Nowlan (Hinton and Nowlan, 1987), the computational approaches on this effect have contributed to understanding of how learning can affect evolution. An important finding of these studies is that the balances between the benefit and cost of learning can smooth the fitness landscape, and as a result, can either facilitate or slow down the adaptive evolution. Especially, it has been reported that there can be situations in which learning is not always beneficial for genetic evolution (Mayley, 1997; Paenke et al., 2006). For example, if there is no cost for learning an adaptive trait, there is no difference in the fitness between the learned one and the genetically acquired one. In this case, the learning behavior can retard the genetic evolution of such a trait because the selection pressure cannot distinguish between these traits. Thus, it is an important issue how learning can become necessary or unnecessary for adaptive genetic evolution depending on various states of a population and its environment.

Recently, we discussed whether and how learning can facilitate the adaptive evolution of population on rugged fitness landscapes (Suzuki and Arita, 2007b). We constructed a simple fitness function that represents a multi-modal fitness landscape as typically illustrated in Fig. 1, in which there is a correlation between the adaptivity of individual and the effects of epistatic interactions among its traits. The evolutionary experiments of the individual traits and their phenotypic plasticity on this landscape clearly showed that the Baldwin effect repeatedly occurred through the evolutionary process of the population, and facilitated its adaptive evolution as a whole.

Also, the effects of learning on evolution have been discussed in the context of dynamically changing fitness landscapes. In such situations, we can expect that more complex scenarios of interactions between evolution and learning emerge because the balances between the benefit and cost of learning also change dynamically. While several studies focused on the effects of changes in the environmental conditions caused by the external factors (Sasaki and Tokoro, 1997; Ancel, 1999), we can also assume more complex situations in which the fitness landscapes can be changed by internal factors (Suzuki and Arita, 2004). The evolution and emergence of communication is one of the typical cases of

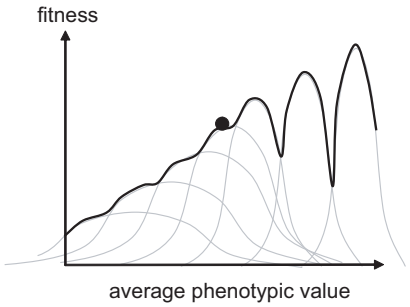


Figure 1: A rough image of the fitness landscape. The horizontal axis corresponds to the average phenotypic value among all phenotypes. Each peak (in gray) corresponds to the fitness which can be acquired when each trait group becomes adaptive. The black line is the actual fitness. As the fitness of the population increases, it tends to need to cross deeper valleys to reach the next optimum.

this situation because the fitness of the individuals are determined by the benefit of the successful communications among them. This topic has been discussed in ALife studies (Noble et al., 2001) from various viewpoints such as the emergence of lexicons through language games among agents (Steels, 1996), the adaptivity and diversity of the mating signals (Werner and Todd, 1997), the emergence of communication in embodied agents (Nolfi, 2005), the complexity of the birdsongs grammar (Sasahara and Ikegami, 2007) and so on. Also, several studies discussed the effects of learning on evolution in the context of language evolution (Arita and Koyama, 1998; Kirby, 2002; Munroe and Cangelosi, 2002; Yamauchi, 2007; Watanabe et al., 2008). For example, Watanabe et al. recently constructed a computational model into which both cultural learning of language and genetic evolution of language ability are incorporated (Watanabe et al., 2008). They found that the factors specific to language evolution (such as adaptive shift in language or overlearning to a variety of parents) are important for the occurrence of the second step in the Baldwin effect.

Among various roles and aspects of communication these previous studies have focused on, the frequency dependence of the individual fitness is one of the common key mechanisms in the evolution of communication. Here, we assume a communication as a process in which one individual generates and sends a kind of signals, then another individual receives and interprets that signal, which can potentially increase the fitness of both individuals. For example, if the individuals can correctly interpret the signals generated by conspecific ones, there is a positive frequency dependent selection on them in that the fitness of such individuals increases as they become more common. This selection pressure facilitates the increase in the number of them in the population, and thus increases the communicative coherence in the population. Nowak discussed the evolution of language (grammar) by using a simple mathematical model in which the language of each individual can genetically or culturally evolve depending on the success in communication with other individuals (Nowak et al., 2001). The results showed that if the accuracy of the language acquisition through genetic or cultural evolution exceeds a certain threshold, the dominant language emerges as a result of positive frequency dependent selection on that language. However, it is expected that once the population is occupied by such individuals, new or different individuals are difficult to express their adaptivity and invade into the population even when their communication is more adaptive than the existing one because of the strong positive selection pressure on the existing individuals.

Furthermore, we focus on the difference in the processing mechanisms between signalling and receiving behaviors, which has often been overlooked in previous studies. The generation and interpretation of a signal are different ecological and cognitive behaviors and the individuals use the different set of traits for generating and sending a signal from that for receiving and interpreting it in general. In addition to the fact that animals have different phonatory and auditory organs, it is also known that the generation and interpretation ability of human language depend on different parts of the human brain such as Broca's and Wernicke's areas in the cortex although there are strong interactions between them (Deacon, 1997). It means that these mechanisms can evolve separately at least in part, and thus the individuals are not always able to correctly interpret the signal generated by the individual itself or conspecific ones. The frequency dependent selection works negatively on such communicatively incoherent individuals. However, how learning can affect evolution under the assumption of these dynamic factors has not been clearly discussed so far.

Our purpose is to clarify how learning can facilitate evolution in dynamically changing environments caused by internal factors. For this purpose, we constructed a simple computational model for the evolution of communication ability and its phenotypic plasticity by using the fitness function adopted in (Suzuki and Arita, 2007b). In the model, the levels of adaptive communication, which is the expected fitness value when the communication results in success, of signaling and receiving processes are determined by different sets of traits under the assumption of the correlation between their fitness and the effects of epistatic interactions among traits. The communication is successful only when the communication levels of the signaller and the receiver are the same, and the individuals try to improve their communication levels through the learning process in which the plastic traits can be modified from their genetically determined values. The evolutionary experiments clearly showed that the Baldwin effect repeatedly occurred and facilitated the adaptive evolution in this kind of dynamic environments.

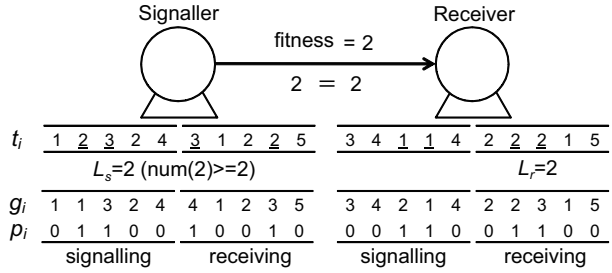


Figure 2: Example of genetic information and communication (M=5). The underlined values in t are plastic traits.

Model

The level of adaptive communication

There are N individuals in a population and each individual has $2M$ traits t_i (i=0, $\cdots$, $2M$-1) as shown in Fig. 2. Each gene g_i (i=0, $\cdots$, $2M$-1) in a $2M$-length chromosome GI represents the initial value of the corresponding trait t_i which consists of an integer value within the range $[1, M]$. Also, each individual has another $2M$-length chromosome GP which decides whether the corresponding trait is plastic ("1") or not ("0"). Each trait t_i whose corresponding bit p_i in GP equals to "1" can be changed through the communication process.

Among $2M$ traits, the former part of M traits determines the cognitive ability for generating and sending signals to others, and the latter part of M traits determines the cognitive ability for receiving and interpreting signals from others. Thus, this model can be regarded as a coevolutionary model of traits for sending and receiving signals. Here, we define the individual's level of adaptive communication of signalling (receiving) a signal L_s (L_r) as follows:

$$L_s(L_r) = \arg\max(f(n)), \tag{1}$$

$$f(n) = \begin{cases} n & \text{if } num(n) \geq n, \\ 0 & \text{otherwise,} \end{cases} \tag{2}$$

where $num(n)$ is the number of traits of which phenotypic value is n among the former (latter) half of M traits. $\arg\max(f(n))$ is the value of n which maximizes the function $f(n)$. This function is basically similar to the one adopted in (Suzuki and Arita, 2007b), and typically describes the following situation: The corresponding M traits of the individual are divided into several groups each in which the phenotypic values are identical. The trait group of n expresses its ability for sending (receiving) the signal of the level n if its group size ($num(n)$) is greater than or equals to n. The actual level of adaptive communication is defined as the highest value among adaptive trait groups. Fig. 2 shows an example of the levels of adaptive communication. Eq. (1) and (2) show that the higher the level of a trait group is, the larger its minimum size that is needed for its ability to express becomes. The increase in the minimum size means that such a group becomes difficult to be acquired because it needs interactions with larger number of phenotypes. Thus, there is a positive correlation between the level of adaptive communication and the effects of epistatic interactions.

Table 1: Example of the set of pairs (N=8). Each number represents the serial number of the individual.

Signaller	2	1	7	4	6	3	5	8
Receiver	8	7	3	1	2	5	6	4

Communication and lifetime learning

In each generation, the N pairs of a signaller and a receiver are randomly arranged under the condition that each individual becomes a signaller once and also becomes a receiver once as shown in Table 1. The communication is successful only when the communication level of the signaller (L_s) and the receiver (L_r) are the same. Both individuals obtain the following fitness value:

$$fitness = \begin{cases} L & \text{if } L_r = L_s(= L), \\ 0 & \text{otherwise,} \end{cases} \tag{3}$$

where L is the shared level between the signaller and the receiver. Fig. 2 shows an example image of success in communication. The individuals obtain the fitness value 2 because both communication levels of signaller and receiver are 2.

For each pair, the $C + 1$ steps of a learning and a communication are conducted. In the initial step, the fitness is evaluated by using the initial communication levels of the signaller and the receiver, which are determined by the initial phenotypic values g_i. Then, during the C steps, both individuals try to communicate by using their corresponding traits t_i all of which phenotypic values are determined by the following equation:

$$t_i = \begin{cases} g_i + rand() & \text{if } p_i = 1, \\ g_i & \text{otherwise,} \end{cases} \tag{4}$$

where $rand()$ is the function that returns a randomly selected value from {-1, 0, 1}. Note that, if a generated phenotypic value exceeds its domain, another randomly selected value is added to the initial value. This equation shows that the values of plastic traits can slightly deviate from their genetically specified values at each step.

The actual fitness of the individual at each step c ($c = 1, \cdots, C + 1$) is defined as the highest value among all c fitness values which are previously measured during the communications in each pair. It means that, in each step, the pairs first try to communicate by using the sets of generated traits, and then adopt the most adaptive trait sets so far.

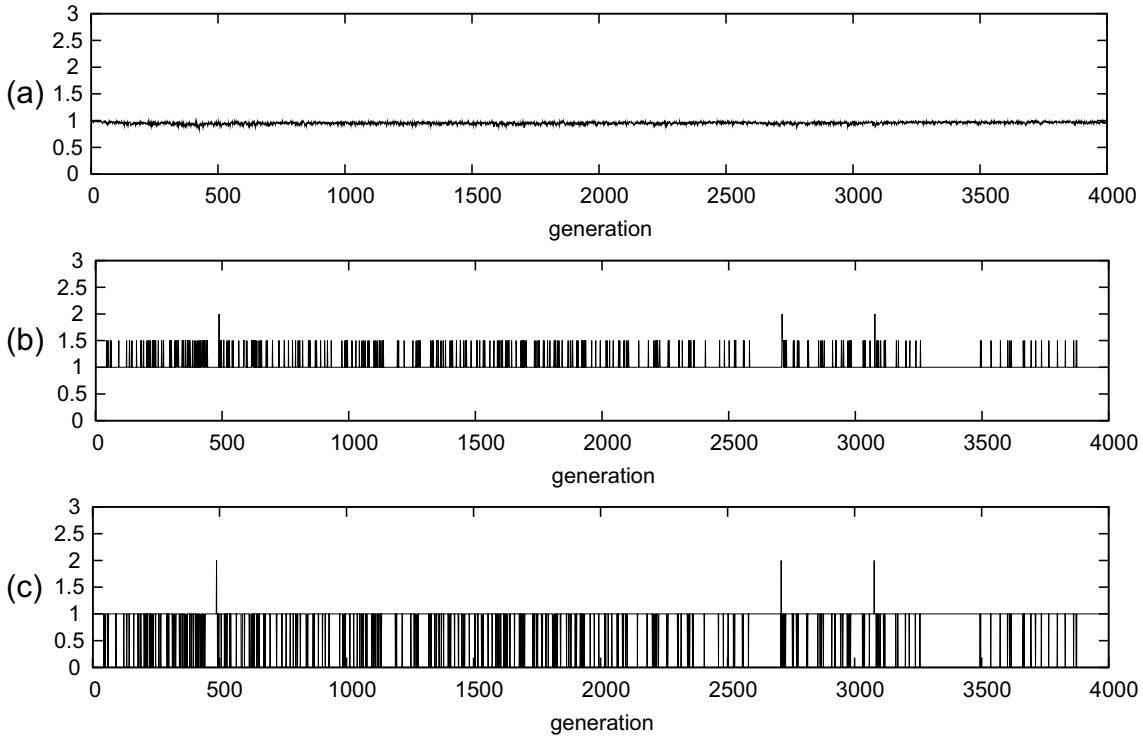


Figure 3: Evolutionary dynamics without learning (C=0). (a) Average lifetime (innate) fitness. (b) Lifetime (innate) fitness of the best individual. (c) Potential fitness of the best individual.

Evolution

This learning and communication process of $C + 1$ steps is conducted in all pairs. The lifetime fitness of each individual, which is used for reproductive process, is defined as the average among the fitness values in its all participating steps. Also, so as to measure the innate adaptivity of each individual, we pick up the (genetically determined) fitness values at the initial step in its all participating processes, and defined the average among them as the innate fitness.

The offsprings in the next generation are selected by the "roulette wheel selection" (in which the probability that an individual will be chosen as an offspring is proportional to its lifetime fitness) from the current population. Then, every gene of all offsprings is mutated with a probability p_{mi} for GI and p_{mp} for GP respectively. A mutation in GI adds a randomly selected value from {-1, 1} to the current value. If a generated value exceeds its domain, another mutation is operated on the original value again. A mutation in GP flips the current binary value.

Results

We conducted evolutionary experiments using the following parameters: N=200, M=10, p_{mi}=0.002 and p_{mp}=0.005. The initial population was generated on condition that initial values in GI were 1 and the genetic values in GP were randomly decided. We adopted this initial condition so as to observe the adaptive evolution from the state in which the individuals have established a successful communication but their level is the lowest.

Experiments without learning

First, we conducted experiments without learning process (C=0). Fig. 3 (a) shows a typical example of the evolution of the lifetime fitness over 4000 generations. The horizontal axis represents the generation, and the line shows the average lifetime fitness at each generation. In this case, the lifetime fitness is the same as the innate fitness. We see that the average lifetime fitness did not increase from the initial value 1.0, thus the population was never able to improve its shared communication level.

We also depicted the lifetime fitness of the best individual (who has the best lifetime fitness among individuals in each generation) in Fig. 3 (b), and its potential fitness in Fig. 3 (c). The potential fitness is the expected value of the fitness when the individual tries to communicate with the focal individual itself. In Fig. 3 (b), we see that the lifetime fitness was basically 1.0 but often increased to 1.5. It means that there appeared several adaptive individuals who succeeded in establishing higher communication level of 2 once during their lifetime, but they could not invade into the population. There are supposed to be two factors for this phenomenon.

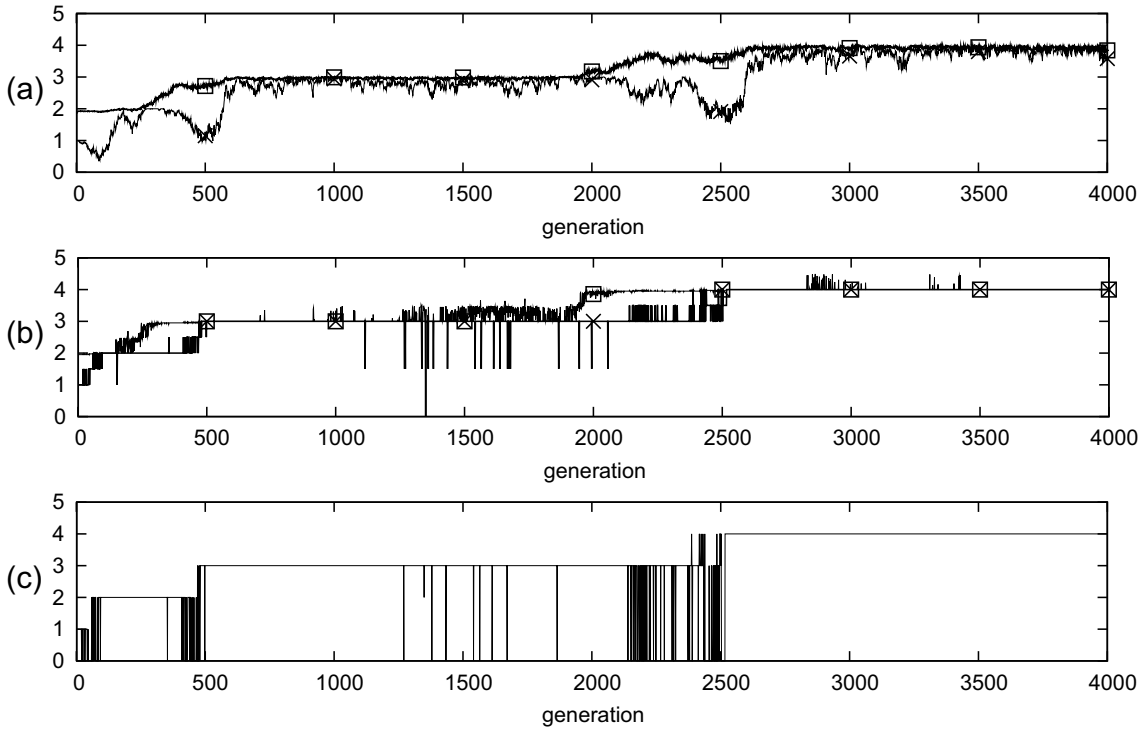


Figure 4: Evolutionary dynamics with learning (C=20). (a) Average lifetime (upper) and innate (lower) fitness. (b) Lifetime (upper) and innate (lower) fitness of the best individual. (c) Potential fitness of the best individual.

The first is the strong positive frequency dependent selection on majority individuals. The appeared individuals mostly fail in communication and disappear at the next generation because they rarely have another chance to meet a similar partner again because they are minority. The second is the negative selection pressure on them caused by the incoherence of their communication levels between signalling and receiving processes. As we see in Fig. 3 (c), their potential fitness was 0. It means that the increase in their proportion rather decreases their own fitness. In addition, we also observe that both best lifetime and potential fitness sometimes reached 2; but such individuals also failed to invade. This is supposed to be due to the fact that the effect of the first factor was quite strong even when the second factor was resolved by chance.

We can say that the population is never be able to improve its shared communication level due to these strong frequency dependent selection pressures if the individuals do not learn.

Experiments with learning

Next, we conducted experiments with learning. Figure 4 shows the typical transitions of the lifetime, innate and potential fitness in case of C=20.

Fig. 4 (a) clearly shows that the average lifetime fitness gradually increased to 4.0. How could the population successfully increase its shared level of adaptive communication despite the fact that the population was never able to increase in the case without learning? This adaptive evolution was due to the repeated occurrences of the Baldwin effect. In Fig. 4 (a), we see the several transitions of the average fitness through which the lifetime fitness increased while the innate fitness decreased, and then the innate fitness subsequently caught up with the lifetime fitness. Each transition can be regarded as a single occurrence of the Baldwin effect.

Here, take the evolution of the population from around 1800th to 2700th generation for example. Around the 1800th generation, both the average lifetime and innate fitness is almost 3.0; all individuals innately established the adaptive communication level of 3. From around 1900th to 2500th generation, the lifetime fitness slowly increased to about 3.5, and the innate fitness gradually decreased. This phenomenon can roughly be regarded as the first step of the Baldwin effect in that the adaptive property of the whole population became dependent on learning process.

The transitions of the lifetime, innate and potential fitness of the best individual in Fig. 4 (b) and (c) give us more detailed information about the evolutionary dynamics of the population during this period. From the 1900th generation, the best lifetime fitness increased to almost 4.0 while the best innate fitness basically remained 3.0 but sometimes

fluctuated. It shows that these best individuals could improve their communication level through learning without discarding successful communications with the majority individuals. It is because that they start to communicate with the level 3 at the initial step, which is coherent with the communication level of the majority individuals, and then try to establish higher level of adaptive communication through learning depending on the level of the partner. Thus, their fitness are basically larger than those of majority individuals 3.0. This benefit of learning enabled these individuals to gradually invade into the population. This can be regarded as the typical first step of the Baldwin effect in that the individuals which could obtain higher level of communication through learning process occupied the population.

Furthermore, if we look at the fluctuation of the best innate fitness in detail, we see that it gradually tended to increase to 3.5 especially after the 2150th generation. This means that as the individual of which the innate levels of signalling and receiving were 3.0 and 4.0 (or 4.0 and 3.0) became more adaptive and invaded into the population. In this model, the more quickly an adaptive communication level is established, the larger the lifetime fitness becomes because it is defined as the average values over all steps. Thus, when the most of individuals can express the communication levels of 4 through learning process, it becomes beneficial for the individuals to genetically acquire the communication level of 4 because they can establish adaptive communications more quickly. In this sense, there is the implicit cost of learning in this model.

We also see the gradual decrease in the average innate fitness and it became about 1.5 at around the 2500th generation as shown in Fig. 4 (a). This is due to the decrease in the potential fitness of the best individuals as shown in Fig. 4 (c). The increase in their number brought about the decrease in the expected innate fitness because they cannot establish the communication with each other without the help of learning. In this sense, the population became more strongly dependent on learning process during the latter generations of this step despite that the genetic assimilation of either level of communication occurred as explained above.

Finally, when such individuals occupied the population, the individuals of which both communication levels of signalling or receiving were 4.0 appeared at around the 2500th generation and then rapidly occupied the population until about the 2700th generation. Both the innate and potential fitness caught up with the lifetime fitness. This is also due to the cost of learning explained above. We can say that the typical second step of the Baldwin effect occurred during this period in that the established communication level of 4 through learning in the first step became genetically acquired in this step completely.

We observed the similar scenarios of the Baldwin effect around the 1-200th and the 250-650th generations, and each process brought about the increase in the communication level of the whole population. In other words, the result of the Baldwin effect became the scaffold for the next Baldwin effect to occur. We also observed that each scenario took longer generations as the lifetime fitness increased because of the increase in the epistasis of adaptive trait group. It also should be noted that when we conducted the experiments with the condition in which all traits were always plastic, it tended to took longer generations for the Baldwin effect to occur (not shown). This means that the evolution of the phenotypic plasticity has an important role for the occurrence of these scenarios although we did not observe significant increase and subsequent decrease in the proportion of plastic traits in our model.

As a whole, we can say that the Baldwin effect repeatedly occur and facilitate the adaptive evolution in this kind of a dynamic environment.

Conclusion

Hinton and Nowlan's pioneering work (Hinton and Nowlan, 1987) clarified that learning can facilitate the evolution on a "needle in a haystack" fitness landscape, and our previous work (Suzuki and Arita, 2007b) also showed that the Baldwin effect also facilitate evolution on a static but rugged fitness landscape as in Fig. 1. In this paper, we further discussed whether and how leaning can facilitate evolution on dynamically changing fitness landscapes which arise from communicative interactions among individuals. We have constructed a simple evolutionary model of the adaptive communication levels and their phenotypic plasticity in which the levels of signalling and receiving processes are determined by different sets of traits under the assumption of the correlation between their adaptivity and the effects of epistatic interactions among traits.

The evolutionary experiments showed that the population with learning successfully increased its shared level of adaptive communication while the population was never able to increase in the case without learning. Here we summarize the observed scenario of evolution by using an image of transition of the population on the dynamic fitness landscape in Fig. 5. The innate communication levels of each individual is represented as a connected set of a circle (signalling) and a square (receiving) filled in dark gray, and the x-axis corresponds to the value of their levels (L or $L+1$). The learned level of the individual is also represented as an open circle or square, which is connected with the innate one. Each arrow represents the communication between two individuals. The vertical axis roughly represents the fitness of the individuals which has corresponding level of adaptive communication.

Let us start from the state in which all individuals have innately established the adaptive communication level of L as shown in Fig. 5 (i). In this state, the population have converged to a single peak of the level L. The individuals which can improve their level from L to L+1 through learning process invade into the population as in Fig. 5 (ii). It is because

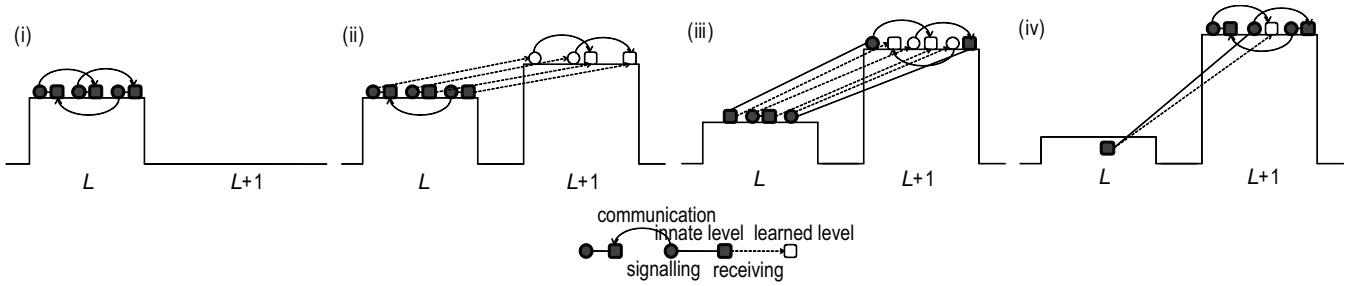


Figure 5: An image of transition of the population on the dynamic fitness landscape.

that such individuals can improve their communication level without discarding successful communications with the majority individuals. This corresponds to the phenomenon that the individuals reach out one of their communication levels to another peak of the level L+1 by using their phenotypic plasticity.

When the most of individuals come to establish the communication levels of L+1 through learning process, the individuals which innately acquired either level of L+1 gradually invade into the population due to the implicit cost of learning as in Fig. 5 (iii). During this process, the peak of the level $L + 1$ gradually becomes higher and that of the level L becomes lower because the individuals have begun to innately use the communication level of L+1. At the same time, the innate fitness decreases because they are communicatively incoherent.

Finally, as in Fig. 5 (iv), the individuals of which both innate levels are L+1 occupy the population. The genetic assimilation of learned traits completely occurs and the peak of the level L disappears at last. This scenario can repeatedly occur and an occurrence of the Baldwin effect can become the scaffold for the next Baldwin effect to occur. This implies that the repeated occurrences of the Baldwin effect might be a general phenomenon which can emerge in both static and dynamic environments.

As explained before, the Baldwin effect has sometimes been discussed in the context of language evolution (Pinker and Bloom, 1990; Nowak et al., 2002; Kirby, 2002). Languages are composed of several levels of syntactic representations as Chomsky clarified (Chomsky, 1957). The level of communication in this model corresponds to a kind of finely classified ones of such syntactic representations. Pinker and Bloom pointed out that comprehension abilities do not have to be in perfect synchrony with production abilities because comprehension can use cognitive heuristics to decode word sequences even in the absence of grammatical knowledge, and a selection pressure on such an adaptive decoding process bring about a kind of innate grammatical module through the Baldwin effect (Pinker and Bloom, 1990). The process in which the learned level for receiving becomes innate one can be regarded as an example of such scenario. Dennett points out that the Baldwin effect is essential to explain the genetic acquisition process of a complex trait such as the innate ability for language acquisition, which is impossible to acquire by evolution alone (Dennett, 2003). He regards Hinton and Nowlan's experimental result (Hinton and Nowlan, 1987) as a typical case of such a scenario. Our results further supports in part his claim in that the Baldwin effect can occur in the context of the evolution of communication among individuals. On the other hand, Deacon also points out that the genetic evolution that can support symbolic communication and the cultural evolution of language can mutually facilitate their evolution, although learning becomes more and more important in his scenario (Deacon, 2003). Although the cultural evolution of the learned traits is not introduced into our model, the repeated occurrences of the Baldwin effect supports his claim in that the acquired adaptive communication through learning process brings about the genetic evolution of the innate communication ability, which results in a further acquisition of more adaptive communication through learning process.

We believe the observed scenarios reflect the general dynamics of interactions between evolution and learning in dynamically changing environments caused by internal factors.

References

Ancel, L. W. (1999). A quantitative model of the Simpson-Baldwin effect. *Journal of Theoretical Biology*, 196:197–209.

Arita, T. and Koyama, Y. (1998). Evolution of linguistic diversity in a simple communication system. *Artificial Life*, 4(1):109–124.

Baldwin, J. M. (1896). A new factor in evolution. *American Naturalist*, 30:441–451.

Baldwin, J. M. (1902). *Development and Evolution*. Blackburn Press.

Chomsky, N. (1957). *Syntactic Structures*. Mouton.

Crispo, E. (2007). The baldwin effect and genetic assimilation: Revisiting two mechanisms of evolutionary

change mediated by phenotypic plasticity. *Evolution*, 61(11):2469–2479.

Deacon, T. W. (1997). *Symbolic species*, W. W. Norton & Company Inc.

Deacon, T. W. (2003). Mutilevel selection in a complex adaptive system: The problem of language origins. In Weber, B. H. and Depew, D. J., editors, *Evolution and Learning -The Baldwin Effect Reconsidered* -, pages 81–112. MIT Press.

Dennett, D. (2003). The baldwin effect: A crane, not a skyhook. In Weber, B. H. and Depew, D. J., editors, *Evolution and Learning -The Baldwin Effect Reconsidered* -, pages 69–79. MIT Press.

Hinton, G. E. and Nowlan, S. J. (1987). How learning can guide evolution. *Complex Systems*, 1:495–502.

Kirby, S. (2002). Natural language from artificial life. *Artificial Life*, 8(2):185–215.

Mayley, G. (1997). Guiding or hiding: explorations into the effects of learning on the rate of evolution. In Husbands, P. and Harvey, I., editors, *Proceedings of the Fourth European Conference on Artificial Life*, pages 135–144. MIT Press.

Maynard-Smith, J. (1987). When learning guides evolution. *Nature*, 329:761–762.

Munroe, S. and Cangelosi, A. (2002). Learning and the evolution of language: The role of culutural variation and learning costs in the baldwin effect. *Artificial Life*, 8(4):311–339.

Noble, J., Di Paolo, E. A. and Bullock, S. (2001). Adaptive factors in the evolution of signaling systems. In Cangelosi, A. and Parisi, D., editors, *Simulating the Evolution of Language*, pages 53–78. MIT Press.

Nolfi, S. (2005). Emergence of communication in embodied agents: Co-adapting communicative and noncommunicative behaviours. *Connection Science*, 17(3-4):231–248.

Nowak, M. A., Komarova, N. L. and Niyogi, P. (2001). Evolution of universal grammar. *Science*, 291:114–118.

Nowak, M. A., Komarova, N. L. and Niyogi, P. (2002). Computational and evolutionary aspects of language. *Nature*, 417:611–617.

Paenke, I., Kaercki, T. and Sendhoff, B. (2006). On the influence of lifetime learning on selection pressure. In Rocha, L. M., Yaeger, L. S., Bedau, M. A., Floreano, D., Goldstone, R. L., and Vespignani, A., editors, *Proceedings of the Tenth International Conference on the Simulation and Synthesis of Living Systems*, pages 500–506. MIT Press.

Pinker, S. and Bloom, P. (1990). Natural language and natural selection. *Behavioral and Brain Sciences*, 13(4): 707-784.

Sasahara, K. and Ikegami, T. (2007). Evolution of birdsong syntax by interjection communication. *Artificial Life*, 13:259–277.

Sasaki, T. and Tokoro, M. (1997). Adaptation toward changing environments: Why darwinian in nature. In Husbands, P. and Harvey, I., editors, *Proceedings of the Fourth European Conference on Artificial Life*, pages 145–153. MIT Press.

Suzuki, R. and Arita, T. (2004). Interactions between learning and evolution: Outstanding strategy generated by the Baldwin effect. *Biosystems*, 77(1-3):57–71.

Suzuki, R. and Arita, T. (2007a). The dynamic changes in roles of learning through the Baldwin effect. *Artificial Life*, 13(1):31–43.

Suzuki, R. and Arita, T. (2007b). Repeated occurrences of the Baldwin effect can guide evolution on rugged fitness landscapes. In *Proceedings of the First IEEE Symposium on Artificial Life (IEEE-ALife'07)*, pages 8–14. IEEE Press.

Steels, L. (1996). Emergent adaptive lexicons. In *Proceedings of the Fourth International Conference on Simulation and Adaptive Behavior*, pages 562–567. MIT Press.

Turney, P., Whitley, D. and Anderson, R. W. (1996). Evolution, learning, and instinct: 100 years of the Baldwin effect. *Evolutionary Computation*, 4(3):4–8.

Waddington, C. H. (1953). Genetic assimilation of an acquired character. *Evolution*, 7:118–126.

Watanabe, Y., Suzuki, R. and Arita, T. (in press). Language evolution and the Baldwin effect. *Artificial Life and Robotics*, 12(1):65–69.

Werner, G. and Todd, P. (1997). Too many love songs: Sexual selection and the evolution of communication. In Husbands, P. and Harvey, I., editors, *Proceedings of the Fourth European Conference on Artificial Life*, pages 434–443. MIT Press.

West-Eberhard, M. J. (2003). *Developmental Plasticity and Evolution*. Oxford University Press.

Yamauchi, H. (2007). How does niche construction reverse the Baldwin effect? In *Proceedings of the Ninth European Conference on Artificial Life*, pages 315–324. Springer.

Self-Assembly in Physical Autonomous Robots: the Evolutionary Robotics Approach

Elio Tuci[†], Christos Ampatzis[†], Vito Trianni[††], Anders Lyhne Christensen[†††] and Marco Dorigo[†]

[†]IRIDIA, CoDE, Université Libre de Bruxelles, Brussels, Belgium
[††]ISTC-CNR, Rome, Italy
[†††]DCTI-ISCTE, Lisbon, Portugal
{etuci,campatzi,mdorigo}@ulb.ac.be, vito.trianni@istc.cnr.it, anders.christensen@iscte.pt

Abstract

This research work illustrates the details of a methodological approach to the design of homogeneous neuro-controllers for self-assembly in physical autonomous robots in which no assumptions are made concerning how agents allocate roles. Artificial evolution is used to set the parameters of a dynamical neural network that when ported on two physical robots allows them to coordinate their actions in order to decide who will grip whom. The neural network directly controls the state of all the actuators. To the best of our knowledge, this work is the first example in which physical robots manage to self-assemble without relying on a priori injected morphological or behavioural heterogeneities. The results shed a light on the minimal requirements necessary to achieve self-assembly in autonomous robots.

Introduction

According to Whitesides and Grzybowski (2002), self-assembly is defined as "the autonomous organisation of components into patterns or structures without human intervention". Nature provides many examples of animals forming collective structures by connecting themselves to one another. Individuals of various ant, bee and wasp species self-assemble and manage to build complex structures such as bivouacs, ladders, etc. Self-assembly in social insects typically happens in order to accomplish some function (e.g., defence, object transport, passage formation, etc.; see Anderson et al., 2002). Ants of the species *Œcophylla longinoda* can form chains composed of their own bodies which are used to pull leaves together to form a nest, or to bridge a passage between branches in a tree (Hölldobler and Wilson, 1978). Self-assembly is also widely observed at the molecular level (e.g., DNA molecules). Although ubiquitous in nature, self-assembly remains in general a phenomenon whose operational principles are not easy to grasp, both in non-living and living organisms, at any scale. This is because "it is impractical to change many of the parameters that determine the behaviour of the system components" (see Whitesides and Grzybowski, 2002). However, self-assembly is particularly appealing to various scientific disciplines. For example, understanding the mechanisms of self-assembly in the cell may provide further insights into the emergence of life starting from chemical reactions. From an engineering point of view, understanding self-assembly may inspire the design of artificial self-assembling components. The application of such systems can potentially go beyond research in laboratories, space applications being the most obvious challenge (e.g., multi-robot planetary exploration and on-orbit self-assembly, see Izzo and Pettazzi, 2007).

Building artificial models that capture the main properties of natural phenomena can provide the means to formulate and test hypotheses concerning the underlying mechanisms of the observed phenomena (see Webb, 2000). Several examples of robotic platforms in the literature consist of connecting modules[1]. Among the various autonomous self-assembling systems that have been proposed in the literature, the work done by Groß et al. (2006) using the robots called *s-bot* is particularly relevant to the subject of our study. Groß et al. (2006) presented experiments improving the state of the art in self-assembling robots concerning mainly the number of robots involved in self-assembly, the generality and reliability of the controllers and the assembly speed. A significant contribution of Groß et al. work is in the design of distributed control mechanisms for self-assembly relying only on local perception. In particular, self-assembly is accomplished with a modular approach in which some modules are evolved and others handcrafted. The approach is based upon a signalling system which makes use of colours. For example, the decision concerning which robot makes the action of gripping (i.e., the *s-bot-gripper*) and which one is gripped (i.e., the *s-bot-grippee*) is made through the emission of colour signals, according to which the robots emitting blue light are playing the role of *s-bot-grippers* and those emitting red light the role of *s-bot-grippees*. Thus, it is the heterogeneity among the robots with respect to the colour displayed, a priori introduced by the experimenter, that triggers the self-assembly process. That is, a single robot "born" red among several

[1]The reader can find comprehensive reviews of the work on autonomous self-assembling systems in (Yim et al., 2002; Groß and Dorigo, 2008b; Tuci et al., 2006).

robots "born" blue is meant to play the role of *s-bot-grippee* while the remaining *s-bot-grippers* are progressively assembling. Once successfully assembled to another robot, each blue light emitting robot is programmed to turn off the blue LEDs and to turn on the red ones. The switch from blue to red light indicates to the yet non-assembled robots the "metamorphosis" of a robot from *s-bot-gripper* to *s-bot-grippee*. This system is therefore based on the presence of a behavioural or morphological heterogeneity. In other words, it requires either the presence of an object lit up in red or the presence of a robot not sharing the controller of the others, which is forced to be immobile and to signal with a red colour. O'Grady et al. (2005) bypassed this requirement by handcrafting a decision-making mechanism based on a probabilistic transition between states. More specifically, the allocation of roles (which robot lights up red and triggers the process) depends solely on a stochastic process.

The research works presented in (Groß et al., 2006) and in (O'Grady et al., 2005) showed how assembled structures can overcome limitations of the single robots, for instance in transporting a heavy object or in navigating on rough terrain. However, the modularised control architecture used in these works to allow the robots to self-assemble is based on a set of a priori assumptions concerning the specification of the environmental/behavioural conditions that trigger the self-assembling process. For example, (a) the objects that can be grasped must be red, and those that should not be grasped must be blue; (b) the action of grasping is carried out only if all the "grasping requirements" are fulfilled (among others, a combination of conditions concerning the distance and relative orientation between the robots, see Groß et al., 2006, for details). If the experimenter could always know in advance in what type of world the agents will be located, assumptions such as those concerning the nature of the object to be grasped would not represent a limitation with respect to the domain of action of the robotic system. However, since it is desirable to have agents that can potentially adapt to variable circumstances or conditions that are partially or totally unknown to the experimenter, it follows that the efficiency of autonomous robots should be estimated also with respect to their capacity to cope with "unpredictable" events (e.g., environmental variability, partial hardware failure, etc.). For example, failure to emit or perceive red light for robots guided by the controllers presented above would significantly hinder the accomplishment of the assembly task.

In this work we aim at designing control structures by which the self-assembly mechanisms do not rely on a priori designer-specified morphological or behavioural differences between the robots, and the individual behaviours are not triggered by a priori designer-specified agents' perceptual cues. To accomplish our objective we exploit the properties of a particular type of design method referred to as Evolutionary Robotics (ER). ER is a methodological tool to automate the design of robots' controllers based on the use of artificial evolution to find sets of parameters for artificial neural networks that guide the robots to the accomplishment of their task (Nolfi and Floreano, 2000). With respect to other design methods, ER provides the methodological tools to generate control structures for artificial agents such as autonomous robots, in a relatively prejudice-free fashion. For example, ER does not require the designer to make strong assumptions concerning what behavioural and communication mechanisms are needed by the robots. The experimenter defines the characteristics of a social context in which robots are required to cooperate. The agents' mechanisms for solitary and social behaviour are determined by an evolutionary process that favours (through selection) those solutions which improve the fitness (i.e., a measure of an agent's or group's ability to accomplish its task).

In this work, we study self-assembly in a setup where the robots interact and eventually differentiate by allocating distinct roles (i.e., *s-bot-gripper* versus *s-bot-grippee*). In particular, two identical robots, placed in a boundless arena at 25/30 cm from each other with random orientations, are required to approach each other and to self-assemble; that is, a robot should physically connect to the other one via a gripper. Instead of a priori defining the mechanisms leading to role allocation and self-assembly, we let behavioural heterogeneity emerge from the interaction among the system's homogeneous components. We show that an integrated (i.e., non-modularised) dynamical neural network in direct control of all the actuators of the robots can successfully tackle real-world tasks requiring fine-grained sensory-motor coordination, such as self-assembly. We show with physical robots that coordination and cooperation in self-assembly do not require explicit signalling of internal states, as assumed, for example, by (Groß et al., 2006). Coordination and role allocation in our system is achieved solely through minimal sensory information and without explicit communication. Groß and Dorigo (2008a) have reached a similar conclusion in a cooperative transport task with simulated robots. Also, due to the nature of the sensory system used, the robots cannot sense the orientation of their group-mates. In this sense, our approach is similar to (and largely inspired from) the one of (Quinn, 2001; Quinn et al., 2003), where role allocation (leader-follower) is achieved solely through infrared sensors. In addition, we also show that the evolved mechanisms are as effective as the modular and hand-coded ones described in (Groß et al., 2006; O'Grady et al., 2005) when controlling two physical robots.

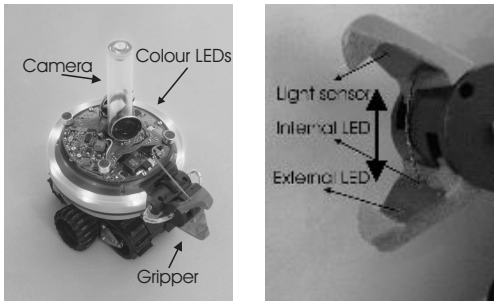
Simulated and Real *S-bot*

An *s-bot* is a mobile autonomous robot equipped with many sensors useful for the perception of the surrounding environment or for proprioception, a differential drive system, and a gripper by which it can grasp various objects or another *s-bot* (see Figure 1a, and Mondada et al., 2004, for further

details on the robot). The main body is a cylindrical turret with a diameter of 11.6 cm, which can be actively rotated with respect to the chassis.

In this work, to allow robots to perceive each other, we make use of the omni-directional camera. The image recorded by the camera is filtered in order to return the distance of the closest red, green, or blue blob in each of the eight $45°$ sectors. Each sector is referred to as C_i with $i \in [1, 8]$. Thus, an *s-bot* to be perceived by the camera must light itself up in one of the three colours using the LEDs mounted on its turret. Notice that the camera can clearly perceive coloured blobs up to a distance of approximately 50 cm, but the precision above 30 cm is rather low. Moreover, the precision with which the distance of coloured blobs is detected varies with respect to the colour of the perceived object. We also make use of the optical barrier which is a hardware component composed of two LEDs and a light sensor mounted on the gripper (see Figure 1b). By post-processing the readings of the optical barrier we extract valuable information concerning the status of the gripper and about the presence of an object between the gripper claws. More specifically, the post-processing of the optical barrier readings defines the status of two virtual sensors: a) the *GS* sensor, set to 1 if the optical barrier indicates that there is an object in between the gripper claws, 0 otherwise; b) the *GG* sensor, set to 1 anytime a robot has gripped an object, 0 otherwise. We also make use of the *GA* sensor, which monitors the gripper aperture. The readings of the *GA* sensor range from 0 when the gripper is completely closed to 1 when the gripper is completely open.

The controllers are evolved in a simulation environment which models some of the hardware characteristics of the real *s-bots*. The simulator used is based on a specialized 2D dynamics engine (see Christensen, 2005). In order to evolve controllers that transfer to real hardware, we overcome the limitations of the simulator by following the approach proposed in (Jakobi, 1997); motion is simulated with sufficient accuracy, collisions are not. Self-assembly relies on rather delicate physical interactions between robots that are integral to the task (e.g., the closing of the gripper around an object can be seen as a collision). Instead of trying to accurately simulate the collisions, we force the controllers to minimise them and not to rely on their outcome. In other words, in case of a collision, the two colliding bodies are repositioned to their previous positions, and the behaviour is penalised by the fitness function if the collision can not be considered the consequence of an accepted grasping manoeuvre. Having taken care of the collisions involved with gripping, the choice of a simple and fast simulator instead of one using a 3D physics engine significantly speeds up the evolutionary process.



(a) (b)

Figure 1: (a) The s-bot. (b) The gripper and sensors of the optical barrier.

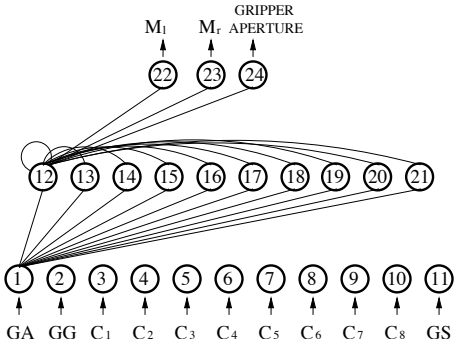


Figure 2: Neural network architecture: only the efferent connections of the first node of each layer are drawn. See text for the meaning of the labels.

Controller and Evolutionary Algorithm

The agent controller is composed of a continuous time recurrent neural network (CTRNN) of ten hidden neurons and an arrangement of eleven input neurons and three output neurons (see Figure 2 and also Beer and Gallagher, 1992). At each simulation cycle, the activation values y_i of input neurons correspond to: the reading of the *GA* sensor for $i = 1$; the reading of the *GG* sensor for $i = 2$; the normalised reading of the eight camera sectors C_j with $j \in [1, 8]$ for $i \in [3, 10]$; the reading of the *GS* sensor for $i = 11$. Hidden neurons are fully connected. Additionally, each hidden neuron receives one incoming synapse from each input neuron. Each output neuron receives one incoming synapse from each hidden neuron. There are no direct connections between input and output neurons. The state of each hidden and output neuron is updated as follows:

$$\tau_i \dot{y_i} = \begin{cases} \sum_{j=1}^{11} \omega_{ji} y_j + \sum_{k=12}^{21} \omega_{ki} \sigma(y_k + \beta_k) - y_i; & i \in [12, 21] \\ \sum_{j=12}^{21} \omega_{ji} \sigma(y_j + \beta_j) - y_i; & i \in [22, 24]; \end{cases}$$

$$\text{with } \sigma(x) = \frac{1}{1 + e^{-x}} \tag{1}$$

In these equations, τ_i is the decay constant, ω_{ij} the strength of the synaptic connection from neuron j to neuron i, β_i the bias term. τ_i with $i \in [12, 24]$, β_i with $i \in [12, 24]$, and all the network connection weights ω_{ij} are genetically specified networks' parameters. $\sigma(y_{22})$ and $\sigma(y_{23})$ linearly scaled into [-3.2s^{-1}, 3.2s^{-1}] are used to set the speed of the left and right motors (M_l, and M_r). $\sigma(y_{24})$ is used to set the gripper aperture in the following way: if $\sigma(y_{24}) > 0.75$ the gripper closes; if $\sigma(y_{24}) < 0.25$ the gripper opens. Cell potentials are set to 0 when the network is initialised or reset, and circuits are integrated using the forward Euler method with an integration step-size $\Delta T = 0.2$.

Each genotype is a vector comprising 263 real values (i.e., 240 genes for the weights, 13 genes for the time constants, 10 genes for the biases). Initially, a random population of vectors is generated by initialising each component of each genotype to values randomly chosen from a uniform distribution in the range [0,1]. The population contains 100 genotypes. Generations following the first one are produced by a combination of selection, mutation, and elitism. For each new generation, the five highest scoring individuals from the previous generation are chosen for breeding. The new generations are produced by making twenty copies of each highest scoring individual with mutations applied only to nineteen of them. Mutation entails that a random Gaussian offset is applied to each real-valued vector component encoded in the genotype, with a probability of 0.25. Genotype parameters are linearly mapped to produce CTRNN parameters with the following ranges: biases $\beta_i \in [-10, 10]$ and weights $\omega_{ji} \in [-10, 10]$. Decay constants are firstly linearly mapped onto the range $[-1.0, 1.5]$ and then exponentially mapped into $\tau_i \in [10^{-1.0}\ 10^{1.5}]$.

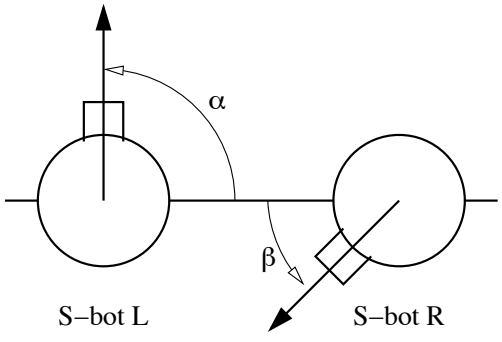


Figure 3: This picture shows how the *s-bots*' starting orientations are defined given the orientation duplet (α,β). *S-bot* L and *s-bot* R refer to the robots whose initial orientations in any given trial correspond to the value of α and β, respectively.

The Task and the Fitness Function

During evolution, each genotype is translated into a robot controller, and cloned onto each agent. At the beginning of each trial, two *s-bots* are positioned in a boundless arena at a distance randomly generated in the interval [25 cm, 30 cm], and with predefined initial orientations α and β (see Figure 3). Our initialisation is inspired from the initialisation used in (Quinn, 2001). In particular, we define a set of orientation duplets (α,β) as all the combinations with repetitions from the set:

$$\Theta_n = \left\{ \frac{2\pi}{n} \cdot i \mid i = 0, \ldots, n-1 \right\}, \tag{2}$$

where n is the cardinality of the set. In other words, we systematically choose the initial orientation of both *s-bots* drawing from the set Θ_n. The cardinality of the set of all the different duplets—where $(\alpha,\beta) \equiv (\beta,\alpha)$—corresponds to the total number of combinations with repetitions, and can be obtained by the following formula:

$$\frac{(n+k-1)!}{k!(n-1)!}, \tag{3}$$

where $k = 2$ indicates that combinations are duplets, and $n = 4$ lets us define the set of possible initial orientations $\Theta_4 = \{0°, 90°, 180°, 270°\}$. From this, we generate 10 different (α,β) duplets. Each group is evaluated 4 times at each of the 10 starting orientation duplets for a total of 40 trials. Each trial (e) differs from the others in the initialisation of the random number generator, which influences the robots initial distance and their orientation by determining the amount of noise added to the orientation duplets (α,β). During a trial, noise affects motors and sensors as well. In particular, uniform noise is added in the range ± 1.25 cm for the distance, and in the range $\pm 1.5°$ for the angle of the object perceived by the camera. Note that, in simulation, colours are not considered. The camera returns distances and angles of the closest object in each sector. 10% uniform noise is added to the motor outputs $\sigma(y_{22})$, $\sigma(y_{23})$. Uniform noise randomly chosen in the range $\pm 5°$ is also added to the initial orientation of each *s-bot*. Within a trial, the robots life-span is 50 simulated seconds (250 simulation cycles). A trial can be terminated earlier if the robots successfully self-assemble in less than 50 simulated seconds, or if they incur in 20 collisions. In each trial e, each group is rewarded by an evaluation function $F_e = A_e \cdot C_e \cdot S_e$ which seeks to assess the ability of the two robots to get closer to each other and to physically assemble through the gripper.

A_e is the aggregation component, computed as follows:

$$A_e = \begin{cases} \frac{1.0}{1.0+atan(\frac{d_{rr}-16}{16})} & \text{if } d_{rr} > 16 \text{ cm;} \\ 1.0 & \text{otherwise;} \end{cases} \tag{4}$$

where d_{rr} is the distance between the two *s-bots* at the end of the trial e.

C_e is the collision component, computed as follows:

$$C_e = \begin{cases} 1.0 & \text{if } n_c = 0; \\ 0.0 & \text{if } n_c > 20; \\ \frac{1.0}{0.5+\sqrt{n_c}} & \text{otherwise;} \end{cases} \quad (5)$$

where n_c is the number of robot-robot collisions recorded during trial e.

S_e is the self-assembly component, computed at the end of a trial ($t = T$ with $T \in (0, 250]$), as follows:

$$S_e = \begin{cases} 100.0 & \text{if } GG(T) = 1, \text{ for any robot;} \\ 1.0 + \frac{29.0 \sum_{t=0}^{T} K(t)}{T} & \text{otherwise;} \end{cases} \quad (6)$$

where $K(t)$ is set to 1 for each simulation cycle t in which the sensor *GS* of any *s-bot* is active, otherwise $K(t) = 0$. Notice that, given the way in which F_e is computed, no assumptions are made concerning which *s-bot* plays the role of *s-bot-gripper* and which one the role of *s-bot-grippee*. The way in which collisions are modelled in simulation and handled by the fitness function is an element that favours the evolution of assembly strategies in which the *s-bot-gripper* moves straight while approaching the *s-bot-grippee*. This has been done to ease transferability to real hardware. The fitness assigned to each genotype after the evaluation of the robots is given by $FF = \frac{1}{E} \sum_{e=1}^{E} F_e$, with $E = 40$.

Results

As stated in the Introduction, in this work we aim at designing through evolutionary computation techniques dynamical neural networks to allow a group of two homogeneous *s-bots* to physically connect to each other. To pursue our objective, we run for 10,000 generations twenty randomly seeded evolutionary simulations. Although several evolutionary runs produced genotypes that obtained the highest fitness score (i.e., $FF = 100$), the ranking based on the evolutionary performances has not been used to select a suitable controller for the experiments with real robots. The reason for this is that during evolution, the best groups may have taken advantage of favourable conditions, determined by the existence of between-generation variation in the starting positions and relative orientation of the robots and other simulation parameters. Thus, the best evolved genotype from generation 5,000 to generation 10,000 of each evolutionary run has been evaluated again on a series of 136,000 trials, obtained by systematically varying the *s-bots*' starting orientations.

In particular, we evaluated the evolved genotypes using a wider set of 16 initial orientations Θ_{16}, defined by equation 2. From this set, equation 3 tells us that we can derive 136 different duplets (α,β). Each starting condition (i.e., orientation duplet) was tested in 1,000 trials, each time randomly choosing the robots' distance from a uniform distribution of values in the range [25 cm, 30 cm]. Noise is added to initial orientations, sensors readings and motor outputs.

The best performing genotype resulting from the set of post-evaluations described above was decoded into an artificial neural network which was then cloned and ported onto two real *s-bots*. In what follows, we provide the results of post-evaluation tests aimed at evaluating the success rate of the real *s-bots* at the self-assembly task as well as the robustness of the self-assembly strategies in different setups.

Post-evaluation Tests on Real *S-bots*

The *s-bots*' controllers are evaluated four times on each of 36 different orientation duplets (α,β), obtained drawing α and β from Θ_8. The cardinality of this set of duplets is given by equation 3, with $n = 8, k = 2$. In each post-evaluation experiment, successful trials are considered those by which the robots manage to self-assemble, that is, when one robot manages to grasp the other one. Note that, for real *s-bots*, the trial's termination criteria was changed with respect to those employed with the simulated *s-bots*. We set no limit on the maximum duration of a trial, and no limit on the number of collisions allowed. In each trial, we let the *s-bots* interact until physically connected.

In a single case we terminated the trial before the robots self-assembled because the *s-bots* ended up outside the perceptual range of their respective camera. This trial has been terminated after one minute of robot-robot distance higher than 50 cm and the trial has been considered unsuccessful. As illustrated later in this Section, these new criteria allowed us to observe interesting and unexpected behavioural sequences. In fact, the *s-bots* sporadically committed inaccuracies during their self-assembly manoeuvres. Unexpectedly, the robots show to possess the required capabilities to autonomously recover from these inaccuracies. In what follows, we provide the reader a detailed description of the performance of the real *s-bots* in these post-evaluation trials.[2] The first two tests with physical robots are referred to as test G25 and test G30. These are tests in which the *s-bots* light themselves up in green and are initialised at a distance from each other of 25 cm and 30 cm, respectively. The *s-bots* proved to be 100% successful in both tests. That is, they managed to self-assemble in all trials. Table 1 gives more details about the *s-bots*' performances in these trials. In particular, we notice that the number of successful trials at the first gripping attempt is 28 and 29 trials out of 36 respectively for G25 and G30 (see Table 1, 2^{nd} column). In a few trials, the *s-bots* managed to assemble after two/three grasping attempts (see Table 1, 3^{rd} and 7^{th} column). The failed attempts were mostly caused by inaccurate manoeuvres—referred to as inaccuracies of type I_1—, in which a series of maladroit actions by both robots makes impossible for the *s-bot-gripper* to successfully grasp the *s-bot-grippee*'s cylindrical turret. In a few other cases, the group committed

[2] Movies of the post-evaluation tests on real *s-bots* can be found at `http://iridia.ulb.ac.be/supp/IridiaSupp2008-002/`

a different inaccuracy—referred to as I_2—, in which both robots assume the role of *s-bot-gripper*. In such circumstances, the *s-bots* head towards each other until a collision between their respective grippers occurs. Note that, in both G25 and G30, the *s-bots* always managed to recover from the inaccuracies and end up successful.

The *s-bots* have to turn on their coloured LEDs in order to perceive each other through the camera. However, a significant advantage of our control design approach is that the specific colour displayed has no functional role within the neural machinery that brings forth the *s-bots*' actions. In order to empirically demonstrate that the mechanisms underpinning the *s-bots* self-assembling strategies do not depend on the specific colour displayed by the LEDs, we repeated a third and a fourth time the 36 post-evaluation trials, both times by deliberately changing the colour of the *s-bots*' LEDs. The *s-bots* are placed at an initial distance of 30 cm from each other, and they are evaluated with the LEDs displaying blue light—this test is referred to as B30—and with the LEDs displaying red light—this test is referred to as R30.

The *s-bots* proved to be very successful both in B30 and R30 (see Table 1). In the large majority of the trials the *s-bots* managed to self-assemble at the first grasping attempt. In a few trials, two or three grasping manoeuvres were required (see Table 1, 3^{rd} and 7^{th} column). A new type of inaccuracy emerged in test R30. That is, in three trials, after grasping, the connected structure got slightly elevated at the connection point. We refer to this type of inaccuracy as I_3. In a single trial in test B30, the *s-bots* failed to self-assemble. In this case, the *s-bots* ended up outside the perceptual range of their respective cameras. This trial in which the *s-bots* spent more than 1 minute without perceiving each other has been terminated, and it was considered unsuccessful.

Table 1: Results of post-evaluation tests on real *s-bots*. G25 and G30 refer to the tests in which the *s-bots* light themselves up in green and are initialised at a distance from each other of 25 cm and 30 cm, respectively. B30 and R30 refer to the tests in which the *s-bots* light themselves up in blue and red respectively, and are initialised at a distance of 30 cm from each other. Trials in which the assembly between the *s-bots* requires more than one gripping attempt, due to inaccurate manoeuvres I_i, are still considered successful. I_1 refers to a series of maladroit actions by both robots which hinder the *s-bot-gripper* from successfully grasping the *s-bot-grippee*'s turret. I_2 refers to those circumstances in which both robots assume the role of *s-bot-gripper* and collide at the level of their grippers. I_3 refers to those circumstances in which, after grasping, the connected structure gets slightly elevated at the connection point.

Test	Number of successful trials per gripping attempt and types of inaccuracy								
	1^{st}	2^{nd}				3^{rd}			
	$N.^\circ$	$N.^\circ$	I_1	I_2	I_3	$N.^\circ$	I_1	I_2	I_3
G25	28	7	6	1	0	1	2	0	0
G30	29	6	3	3	0	1	1	1	0
B30	26	5	3	2	0	4	8	0	0
R30	21	12	10	0	2	4	7	0	1

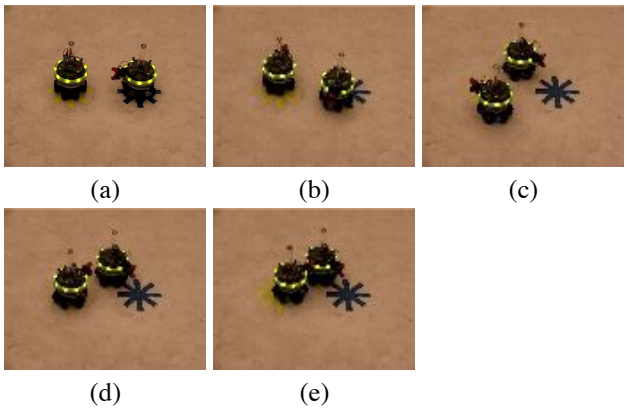


Figure 4: Snapshots from a successful trial. (a) Initial configuration; (b) Starting phase; (c) Role allocation phase; (d) Gripping phase; (e) Success (grip).

For each single test (i.e., G25, G30, B30, and R30), the sequences of *s-bots*' actions are rather different from one trial to the other. However, these different histories of interaction can be succinctly described by a combination of few distinctive phases and transitions between phases which exhaustively "portray" the observed phenomena. Figure 4 shows some snapshots from a successful trial which represent these phases. The robots leave their respective starting positions (see Figure 4a) and during the starting phase (see Figure 4b) they tend to get closer to each other. In the great majority of the trials, the robots move from the starting phase to what we call the role allocation phase (RA-phase, see Figure 4c). In this phase, each *s-bot* tends to remain on the right side of the other. They slowly move by following a circular trajectory corresponding to an imaginary circle centred in between the two *s-bots*. Moreover, each robot rhythmically changes its heading by turning left and right. The RA-phase ends once one of the two *s-bots*—that is, the one assuming the role of the *s-bot-gripper*—stops oscillating and heads towards the other *s-bot*—that is, the one assuming the role of the *s-bot-grippee*—which instead orients itself in order to facilitate the gripping (gripping phase, see Figure 4d). The *s-bot-gripper* approaches the *s-bot-grippee*'s turret and, as soon as its *GS* sensor is active, it closes its gripper. A successful trial terminates as soon as the two *s-bots* are connected (see Figure 4e). As mentioned above, in a few trials the *s-bots* failed to connect at the first gripping attempt by committing what we called inaccuracies I_1 and I_3. These inaccuracies seem to denote problems in the sensory-motor coordination during grasping. Recov-

ering from I_1 can only be accomplished by returning to a new RA-phase, in which the *s-bots* re-establish again their respective roles, and eventually self-assemble. Recovering from I_3 is accomplished by a slight backward movement of both *s-bots* which restores a stable gripping configuration. Given that I_3 has been observed only in R30, it seems plausible to attribute the origin of this inaccuracy to the effects of the red light on the perceptual apparatus of the *s-bots*. In particular, it could be that, due to the red light, the *s-bot-gripper* perceives through its camera the *s-bot-grippee* at a farther distance than the actual one. Alternatively, it could be that the red light perturbs the regular functioning of the optical barrier and consequently the readings of the *GS* and *GG* sensors. Both phenomena may induce the *s-bot-gripper* to keep on moving towards the *s-bot-grippee* up to the occurrence of I_3, even though the distance between the robots and the status of the gripper of the *s-bot-gripper* would require a different response. I_2 seems to be caused by the effects of the *s-bots*' starting positions on their behaviour. In those trials in which I_2 occurs, after a short starting phase, the *s-bots* head towards each other until they collide with their grippers without going through the RA-phase. The way in which the robots perceive each other at starting positions seems to be the reason why they skip the RA-phase. Without a proper RA-phase, the robots fail to autonomously allocate between themselves the roles required by the self-assembly task (i.e., *s-bot-gripper* and *s-bot-grippee*), and consequently they incur in I_2. In order to recover from I_2, the *s-bots* move away from each other and start a new RA-phase in which roles are eventually allocated. In the future we will further investigate the exact cause of the inaccuracies.

As shown in Table 1, except for a single trial in test B30 in which the *s-bots* failed to self-assemble, the robots proved capable of recovering from all types of inaccuracies. This is an interesting result because it is evidence of the robustness of our controllers with respect to contingencies never encountered during evolution. Indeed, in order to speed up the evolutionary process, the simulation in which controllers have been designed does not handle collisions with sufficient accuracy. In those cases in which, after a collision, the simulated robots had another chance to assemble, the agents were simply re-positioned at a given distance from each other. In spite of this, *s-bots* guided by the best evolved controllers proved capable of engaging in successful recovering manoeuvres which allowed them to eventually assemble.[2]

Conclusion

In this article, we have presented the results of an evolutionary methodology for the design of control strategies for self-assembling robots. To the best of our knowledge, the control method we have proposed for the physical connection of two robots is the only existing in the literature where the role allocation between gripper and grippee is the result of an autonomous decision-making process between two homogeneous robots; there is no a priori injected behavioural or morphological heterogeneity in the system. Instead, the behavioural heterogeneity emerges through the interaction of the robots. Moreover, the communication requirements of our approach are reduced to the minimum; simple coordination by means of the dynamical interaction between the robots—as opposed to explicit communication of internal states—is enough to bring forth differentiation within the group. We believe that reducing the assumptions on necessary conditions for assembly is an important step to obtain more adaptive and more general controllers for autonomous self-assembly. The results of this work are a proof-of-concept: they proved that dynamical neural networks shaped by evolutionary computation techniques directly controlling the robots' actuators can provide physical robots all the required mechanisms to autonomously perform self-assembly. Contrary to the modular or hand-coded controllers described in Groß et al. (2006) and in O'Grady et al. (2005), the evolutionary robotics approach did not require the experimenter to make any a priori assumptions concerning the roles of the robots during self-assembly (i.e., either *s-bot-gripper* or *s-bot-grippee*) or about their status (e.g., either capable of moving or required not to move). The evolved mechanisms proved to be robust with respect to changes in the colour of the light displayed by the LEDs. Furthermore, we have designed a self-assembling system that exhibits recovery capabilities that have not been selected during the evolutionary design phase and that were not coded or foreseen by the experimenter. Such a feature in our case comes for free, while in the case of Groß et al. (2006) a recovery mechanism had to be designed as a specific behavioural module to be activated every time the robots failed to achieve assembly.

Our system is not as "transparent" as a hand-coded or modular rule-based one, as we can not break its behaviour down to a set of rules or states. Such an endeavour seems to be very challenging and particularly difficult, especially when the network sizes are large and/or the movement of the robots takes place in a continuous and noisy world, such as the real world. However, preliminary results not shown in the paper suggest that there is an effect of the starting configuration on the final outcome of a trial (how roles are allocated). In short, our analysis revealed that, in those trials in which the two robots have different initial perceptions ($\alpha \neq \beta$), the role that each *s-bot* assumes can be predicted knowing the combination of α and β. However, it is important to notice that perceiving the other robot at a specific distance and through a given camera sector does not inform a robot about the role it will assume during the trial. In other words, it is this combination of α and β which determines the roles. In those cases in which the robots start with an identical perception ($\alpha = \beta$), this symmetry does not seem to hinder the robots from autonomously allocating different roles to successfully accomplish their goal. At the moment, it is unclear how the initial symmetry is broken.

Perhaps, the driving forces have to be searched in the way in which the robots mutually affect each other's behaviour. Perhaps, the random noise injected into the system is the causal factor that drives the system through sequences of actions that turn out to be successful. Stochastic phenomena may take over any causal relationship between environmental structures (i.e., how the robots perceive each other at the beginning of a trial) and the role allocation process. Future analyses are certainly required to see whether any invariants can be found among the history of interactions between the robots and what significance can be attributed to them. We would also like to test the scalability of our system. Can the controllers still manage to achieve assembly if there are more than two robots involved? Some initial experimentation[2] looks very promising. However, we plan to introduce coordinated motion capabilities to the robots behavioural repertoire before we systematically address this issue. In other words, the assembled structure of two or more robots must be able to move coordinately, in order to actively participate in the assembly process. For example, it could interact with other assembled structures or individual robots by either receiving connections from them or grasping them. We will also study more complex scenarios in which self-assembly is functional to the achievement of particular objectives that are beyond the capabilities of a single robot.

Acknowledgements

The authors thank Francisco Santos, Roderich Groß, Marco Montes de Oca and their colleagues at IRIDIA for stimulating discussions and feedback during the preparation of this paper. E. Tuci and M. Dorigo acknowledge European Commission support via the *ECAgents* project, funded by the Future and Emerging Technologies programme (grant IST-1940). M. Dorigo acknowledges support from the Belgian FNRS, of which he is a Research Director. M. Dorigo and C. Ampatzis acknowledge support from the "ANTS" project, an "Action de Recherche Concertée" funded by the Scientific Research Directorate of the French Community of Belgium. The information provided is the sole responsibility of the authors and does not reflect the Community's opinion. The Community is not responsible for any use that might be made of data appearing in this publication.

References

Anderson, C., Theraulaz, G., and Deneubourg., J. (2002). Self-assemblages in insect societies. *Insectes Sociaux*, 49(2):99–110.

Beer, R. D. and Gallagher, J. C. (1992). Evolving dynamical neural networks for adaptive behavior. *Adaptive Behavior*, 1:91–122.

Christensen, A. (2005). Efficient neuro-evolution of hole-avoidance and phototaxis for a swarm-bot. DEA thesis TR/IRIDIA/2005-14, Université Libre de Bruxelles, Bruxelles, Belgium.

Groß, R., Bonani, M., Mondada, F., and Dorigo, M. (2006). Autonomous self-assembly in swarm-bots. *IEEE Transactions on Robotics*, 22(6):1115–1130.

Groß, R. and Dorigo, M. (2008a). Evolution of solitary and group transport behaviors for autonomous robots capable of self-assembling. *Adaptive Behavior*. In press.

Groß, R. and Dorigo, M. (2008b). Self-assembly at the macroscopic scale. *Proceedings of the IEEE*. In press.

Hölldobler, B. and Wilson, E. O. (1978). The multiple recruitment systems of the african weaver ant, Oecophylla longinoda (Latreille) (Hymenoptera: Formicidae). *Behavioural Ecology and Sociobiology*, 3:19–60.

Izzo, D. and Pettazzi, L. (2007). Autonomous and distributed motion planning for satellite swarm. *Journal of Guidance Control and Dynamics*, 30(2):449–459.

Jakobi, N. (1997). Evolutionary robotics and the radical envelope of noise hypothesis. *Adaptive Behavior*, 6:325–368.

Mondada, F., Pettinaro, G., Guignard, A., Kwee, I., Floreano, D., Deneubourg, J.-L., Nolfi, S., Gambardella, L., and Dorigo, M. (2004). Swarm-bot: A new distributed robotic concept. *Autonomous Robots*, 17(2–3):193–221.

Nolfi, S. and Floreano, D. (2000). *Evolutionary Robotics: The Biology, Intelligence, and Technology of Self-Organizing Machines*. MIT Press, Cambridge, MA.

O'Grady, R., Groß, R., Mondada, F., Bonani, M., and Dorigo, M. (2005). Self-assembly on demand in a group of physical autonomous mobile robots navigating rough terrain. In Capcarrere, M., Freitas, A., Bentley, P., Johnson, C., and Timmis, J., editors, *Proc. of the* 8^{th} *European Conference on Artificial Life (ECAL05)*, volume 3630 of *LNCS*, pages 272–281. Berlin, Germany:Springer-Verlag.

Quinn, M. (2001). Evolving communication without dedicated communication channels. In *Advances in Artificial Life: Sixth European Conference on Artificial Life (ECAL01)*, volume 2159 of *LNCS*. Berlin, Germany:Springer-Verlag.

Quinn, M., Smith, L., Mayley, G., and Husbands, P. (2003). Evolving controllers for a homogeneous system of physical robots: Structured cooperation with minimal sensors. *Philosophical Transactions of the Royal Society of London, Series A: Mathematical, Physical and Engineering Sciences*, 361:2321–2344.

Tuci, E., Groß, R., Trianni, V., Bonani, M., Mondada, F., and Dorigo, M. (2006). Cooperation through self-assembling in multi-robot systems. *ACM Transactions on Autonomous and Adaptive Systems*, 1(2):115–150.

Webb, B. (2000). What does robotics offer animal behaviour? *Animal Behaviour*, 60:545–558.

Whitesides, G. M. and Grzybowski, B. (2002). Self-assembly at all scales. *Science*, 295:2418–2421.

Yim, M., Zhang, Y., and Duff, D. (2002). Modular robots. *IEEE Spectrum*, 39(2):30–34.

Evolution, Development and Environment Toward Adaptation through Phenotypic Plasticity and Exploitation of External Information

Gunnar Tufte

Norwegian University of Science and Technology
Department of Computer and Information Science
Sem Selandsvei 7-9 7491 Trondheim Norway
gunnart@idi.ntnu.no

Abstract

Biological organisms have an inherent ability to respond to environmental changes. The response can emerge as organisms that can develop into structural and behavioural different phenotypes. To achieve such properties in an artificial developmental setting external environmental information is included in the gene regulation of the developmental model. This implies interplay between evolution, development and the environment. An experimental approach is taken to investigate this interplay. The test case chosen is evolution of robustness to environmental fluctuations. Development models with and without environmental information included in the gene regulation are compared. Further, the developing organisms of the two models are exposed to environmental fluctuations for a more extensive investigation. The results indicate that including external information in the gene regulation can be favourable and exploitable, particularly for organisms developing in a dynamic environment.

Introduction

A developmental mapping is an example of an indirect mapping. In biological development, an initial unit—a cell, holds the complete building plan (DNA) for an organism. It is important to note that this plan is generative—it describes how to build the system, not what the system will look like. Similarly in a developmental mapping, the artificial organism starts out as a single cell where the genome provides the cell's DNA. The processing of the genome may be based on gene regulation (Lantin and Fracchia, 1995). Each development step, or stage in the mapping, produces a candidate phenotype, i.e. an emerging phenotype. Gene regulation implies that different parts of the genome are expressed in different cells at different times in the emerging phenotype.

An important feature of natural development is that the developing organism develops within an environment. In Tufte and Haddow (2007a) environment was discussed at different levels. Intra-cell environment that the DNA resides in, also referred to as the *cell's metabolism* (Federici, 2004; Gordon and Bentley, 2005). The next level of environment, found in most development models, is the neighbour environment referring to the *inter-cell environment*, enabling communication between neighbouring cells (Bongard and Pfeifer, 2003; Tufte and Haddow, 2003; Miller, 2004; Federici, 2004). Further, the environment may also affect the phenotype emerging from the development process.

Phenotypic plasticity (Larsen, 2004) is a property of organisms which enables adaptation or response to the environment. The adaptation or response is expressed as changes in the phenotypic structure and/or behaviour. It is important to note that this adaptation occurs during the development phase. That is, the genome develops in an environment where the emerging phenotype is influenced by the environment in which it develops. This implies that developing organisms may adapt their structure and/or functionality according to information provided by the external stimulus of the environment — see Tufte and Haddow (2007b).

In an artificial setting, environmental adaptation may be regarded as an emerging tolerance to external fluctuations. In the work of Miller (2003) and Federici and Downing (2006) such robustness appeared to be a shadow effect (Hogeweg, 2000) of evolution and development as it was not specified as a target behaviour or included environmental information in the processes of evolution and development. However, environmental influence may be targeted by evolution to find robust genomes (Tufte and Haddow, 2007a). Additionaly, environmental information can be exploited by development for further adaptation of the emerging organism (Tufte and Haddow, 2007b).

To further investigate the relation between evolution of robustness and possible emergent robustness beyond the scope of the environmental fluctuations induced during evolution, the results of Tufte and Haddow (2007a) are compared to results obtained introducing the possibility of exploiting phenotypic plasticity in the development model. Further, the evolved genomes are exposed to large environmental fluctuations during development to reveal possible emergent robustness to such dynamic environmental fluctuations. As such, the evolved genomes are exposed to different environmental fluctuations during development then what fluctuations the population was exposed to during evolution.

The focus of this paper is to investigate if a developmen-

tal model capable of exploiting environmental information, i.e. phenotypic plasticity, indicates increased tolerance to extended environmental fluctuations compared to a developmental model with no such mechanisms. The possible presence of such extended emergent robustness may indicate exploitation of environmental information in interplay with the development of structure and behaviour. However, the larger goal of the work is toward an understanding of the interplay between evolution, development and environment toward artificial organisms capable of computation. Herein a cellular computational machine Sipper (1997). As such, if the environmental information is treated as data input and output from the functional developing organism a clear separation between the emergent structural organism and the data transformation of the functional parts of the organism may not be feasible or desirable.

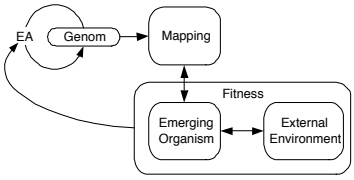
The article is laid out as follows: Section II introduces the roles of environment in artificial development models. The cellular developmental model is presented in Section III. Experimental results are presented in Section IV. Finally, Section V concludes the work.

Environmental Information

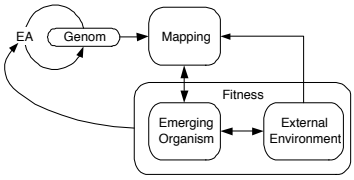
The "environment" of a naturally developing organism usually refers to the external environment affecting the developing organism. In Tufte and Haddow (2007a) this environment was expressed as a combination of both an *initial environment* and an *external environment*. When a cell is grown it is effected by the environment (at that place in the environment) — initial environment. The status of the initial environment thus affects the path of development for any given cell and thus affects the organism as a whole. However, when the organism is developed it has to survive in an environment and thus it is important that the environment beyond the growing organism can affect the developing organism. Such an environment is defined as the external environment. As such, both the growing organism and its "external environment" can be measured during evaluation (Tufte and Haddow, 2007a).

A further implication of emerging organisms is that the phenotype may be evaluated at a given step of development, defined as the finalised phenotype, as in Gordon and Bentley (2005) or at each or any stage during development (Tufte and Haddow, 2003). The latter takes the actual process of developing the emergent structure (Viswanathan and Pollack, 2005) or functionality (Tufte and Haddow, 2003) into the evaluation process i.e. life-time evaluation.

One notable feature of the work presented in Tufte and Haddow (2007a) is that although an external environment is introduced, such an environment only affects the developing phenotype indirectly i.e. through evolution. There is no direct influence on the developing phenotype unlike the initial environment which directly affects the development path of all cells. However, in biology the external environment has a direct affect on the developing phenotype.



(a) Indirect environmental influence through evolution.



(b) Direct environmental influence exploitable by the mapping process.

Figure 1: Evolution of developmental genomes with indirect and direct exploitation of environmental information.

Figure 1(a) illustrates the inclusion of external environment as implemented in Tufte and Haddow (2007a). The organism emerges as a product of the interplay between the genome and the emerging organism. This interplay is represented as the "mapping" box where at any point in time the information about the genome and the organism (at that point in time) are available to the mapping process. Fitness measures the emerging organism together with its environment, as shown, at each stage of the development process. The accumulated fitness, after the mapping process is stopped, is fed back to the evolutionary algorithm (EA). As such, the external environment does not influence the outcome of the development process (mapping) but rather the fitness evaluation thus providing an indirect dependence on the external environmental, i.e. a system with no mutual perturbatory channels (Quick et al., 1999).

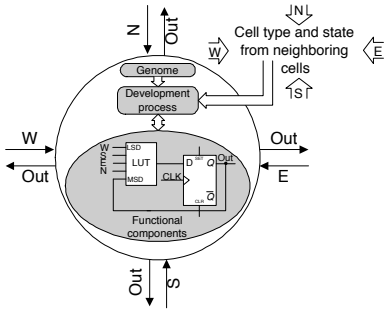
In Figure 1(b) a similar mapping process is described except that the external environment information is available to the development process. As shown, the mapping process can exploit external environment information, in addition to the information coded in the genome and provided by the developing organism. As such, the emerging organism is a product of the interplay between the genome, the organism (at that point in time) and the present environment i.e. mutual perturbatory channels exist. In such systems, a genome can develop into different organisms depending on the environment present, i.e. phenotypic plasticity is achievable (Tufte and Haddow, 2007b).

In the work presented the two different principles for exploiting environmental information shown in Figure 1 are

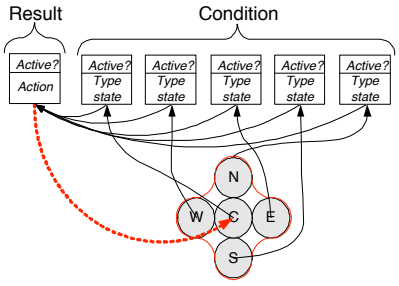
compared to measure the ability to evolve robust phenotypes. The results of the two approaches are taken further by introducing environmental fluctuations during development, i.e. robustness in a dynamic environment.

Development Model

The development model is based on cellular development. This implies that the genome is present and processed autonomously in every cell. In the model, the cell also contains the functional building blocks. For the experiments herein the application sought is that of a digital circuit (phenotype). Figure 2(a) illustrates the developmental system — the cell. The cell is divided into three parts: the genome (the building plan); the development process (mechanisms for cell growth and differentiation) and the functional component of the cell. The information in the functional components represents the type of the cell and the cell's state is described by the outputs of the functional components.



(a) Components of the cell.



(b) Gene regulation shown for a single rule

Figure 2: The basic cell and a rule showing the gene regulation in the cellular development model.

The genome consists of a set of rules. Rules are restricted to expressions consisting of the type and state of the target cell and the types and state of the cells in its von Neumann neighbourhood. There are two types of rules i.e. change and growth rules. Cell growth is a mechanism to expand the organism. A growth rule result provides the direction of growth: grow from north **Gn**; east **Ge**; south **Gs** or west **Gw**. It is important to note that these rules are expressed in terms of where the source of the cell growing into the target cell is. Describing where a cell is growing from enables a fully parallel implementation of the system to be created whilst retaining the possibility that cells in effect may grow in all four directions simultaneously. Growth rules have two restrictions. First, the target cell must be empty – this is to prevent growing over an existing cell and thus specialising the cell with a new cell type. Secondly, the cell to be copied into the target can not be empty.

Differentiation changes a cell's type i.e. its functionality. The result part of a change rule states the type of cell the target is going to be changed into. Cells have the following types: valid cell types, don't care (DC) or empty. However, the empty cell is not a valid target cell type.

Each rule consists of a result and a condition. The conditional part provides information about the cell itself and each of the neighbouring cells. In the development model presented in (Tufte and Haddow, 2005), the type of the cell was applied to describe these cells. However, to introduce external environment, state information is also needed. State information provides a way to include information relating to the functionality of the organism at a given point in time as well as information about the external environment — the empty cells in the environment also have state information. As such, a cell is represented in the condition of a rule by two genes representing its type and its state. However, a target cell is only represented by one gene: it's type for change rules or growth direction for growth rules. The state of cell may be 0, 1 or DC. DC is introduced to provide the possibility to turn on or off this environmental influence. The development model is applied with and without the information from the external environment and functional organism.

Firing of a rule can cause the target cell to change type, die (implemented as a change of type) or cause another cell to grow into it. Figure 2(b) illustrates the process of evaluating a rule. For each cell condition, the cell type and state are compared and if the conditions are true then that part of the rule is active. If all conditions are active then the result will become active and the rule will fire. Activation of the result gene is expressed in the emerging phenotype according to the action specified.

In a development genome multiple rules are present. Multiple rules imply that more than one rule of a given cell may be activated at the same time if their conditions hold. To ensure unambiguous rule firing, rule regulation is part of the development process. If the first rule is activated, the second rule can not be activated. Activation of the second rule prevents activation of the third rule, etc.

The functional components of the cell is an Sblock (Haddow and Tufte, 2000). The content of the look-up table (LUT) defines functionality and is, herein, also used to define the cell type. The LUT is the combinatorial component and the flip-flop is the memory element — capable of storing the cell state. The output value of an Sblock is synchronously updated and sent to all its four

neighbours and as a feedback to itself.

One update of the cell's type under the execution of the development process is termed a development step (DS). A development step is thus a synchronous update of all cells in the cellular array. The update of the cell's functional components i.e. one clock pulse on the flip-flop, is termed a state step (SS). A development step is thus made up of a number of state steps.

The initial condition is applied before development starts. This means that all empty cells are set or reset depending on the given initial condition. To avoid empty cells updating their output values from their von Neumann neighbourhood, all cells of type Empty are set to update their outputs based on only their own output value at the previous clock pulse. A empty cell will retain its initial state — environmental information, until the emerging organism grows into it.

Experiments

The experiments are separated into three different experiments. In the two first experiments each genome was exposed to a set of ten different randomly generated environments. As such, the development of a given genome is repeated ten times, one for each environment. The fitness score was calculated as the mean fitness of the genome in the ten different environments. A genome is thus explicitly being evaluated and, therefore, evolved to tolerate different environments. Ten runs were conducted resulting in a collection of the ten best developmental genomes and their respective developed organisms. The use of environmental information in the development model for the two experiments is illustrated in Figure 1(a) and 1(b).

Extending the developmental model to include environmental information in the gene regulation has several implications. The environmental information requires an extension of the information processing in the gene regulation. As such, the genomes for experiment one and two may not be directly comparable. The larger genome required to include environmental information changes the search space. The genome with no environmental regulation may be defined with parts, e.g. genes, set to don't care to maintain an even genome size, i.e. a redundant representation. Another possibility is to remove the genetic parts that are included in the environmental regulation. In the work of Shipman et al. (2000) and Rothlauf and Goldberg (2003) such redundant representation was shown to be non-favourable or to decrease the performance of the EA. As such, the later solution was chosen.

Further, the resulting genomes from the two experiments were re-developed and re-evaluated in ten other randomly generated environments. Each genome's performance on each of the ten new environments was then compared to the fitness value obtained from the base experiments.

In the third experiment genomes of the two first experiments were re-developed and re-evaluated in an environment were changes was introduced during the life time of the organism. The change in external information was inserted at three fixed steps during development of the organism.

Experimental Setup

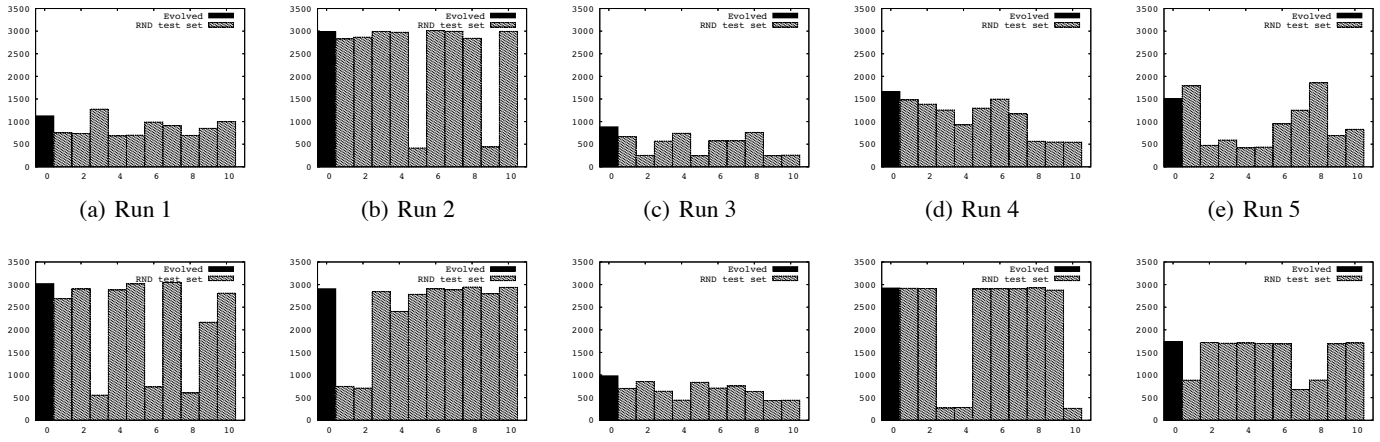
The number of available cell types was set to thirteen including the empty cell type. Available cell types was based on Sippers universal non-uniform CA (Sipper, 1997) and threshold elements (Beiu et al., 2003). Table 1 provides the set of available cell types, together with their functional LUT definition and graphical symbol. For signal directions and LUT addresses refer to Figure 2. The first single cell which the multicellular organism develops from was defined to be of type 5 (NAND).

Table 1: Definition of cell types and their functionality

Cell type	LUT hex	Function name	Graphical representation
0	$0xFFFF0000$	$no\ change\ Emty$	
1	$0x66666666$	$XOR_d\ W \oplus S$	
2	$0x3D3D3D3D$	$XOR_c\ E \oplus S$	
3	$0x0FF00FF0$	$XOR_b\ N \oplus E$	
4	$0x55AA55AA$	$XOR_a\ W \oplus N$	
5	$0x55FF55FF$	$NAND\ \overline{W \bullet N}$	
6	$0xFF00FF00$	$\downarrow SouthPropagation$	
7	$0xCCCCCCCC$	$\uparrow NorthPropagation$	
8	$0xF0F0F0F0$	$\leftarrow EastPropagation$	
9	$0xAAAAAAAA$	$\rightarrow WestPropagation$	
10	$0xE8808000$	$T \geq 4$	
11	$0xFEE8E880$	$T \geq 3$	
12	$0xFFFEFEE8$	$T \geq 2$	

The evolutionary algorithm chosen was a Genetic Algorithm (GA), a modified version of a GA found in Spears (1991). The GA's crossover operator was modified such that a gene was undisturbed and a variable number of crossover points was implemented. The genome size was set to consist of 32 rules and the population size was set to 16. The initial population consisted of random generated valid rules. However, invalid rules may arise through the application of genetic operators. Crossover rate was set to 0.5 and the mutation rate for each gene was set to 0.0017. The GA was set to terminate after 100 000 generations.

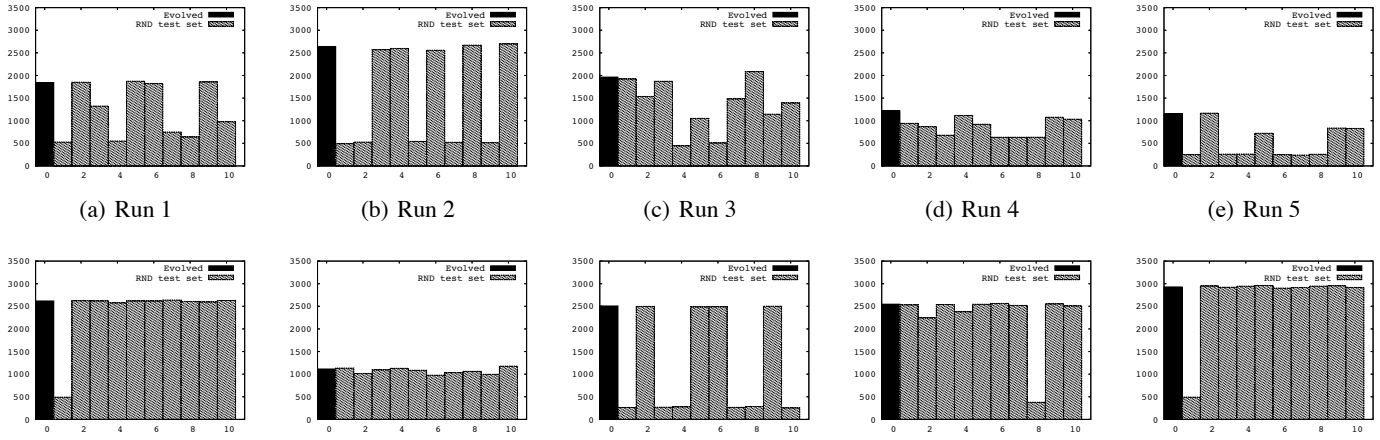
The fitness considers how well an organism function in a set of environments. The application is a sequential counter where counting is based on the state information of the entire cellular space and the sequential operation of the functional components of the cells. The application thus places a requirement on the tuning of the development genome (by evolution) and the emerging phenotype (by development) for such sequential digital circuit behaviour. A counting sequence is defined in the cellular array as the number of logical "1"s in the cellular array increasing by one for each state step. The goal being to achieve counting behaviour in all environments applied, i.e. the same functionality. In this case, a life-time fitness evaluation was used. This is similar to those performed in Tufte and Haddow (2007a) where



(a) Run 1 (b) Run 2 (c) Run 3 (d) Run 4 (e) Run 5

(f) Run 6 (g) Run 7 (h) Run 8 (i) Run 9 (j) Run 10

Figure 3: Evolved in a set of initial random environment. Exposed to random environments. No phenotypic plasticity



(a) Run 1 (b) Run 2 (c) Run 3 (d) Run 4 (e) Run 5

(f) Run 6 (g) Run 7 (h) Run 8 (i) Run 9 (j) Run 10

Figure 4: Evolved in a set of initial random environment. Exposed to random environments. Phenotypic plasticity introduced

different environments were used but here environment may also affects the developing phenotype directly.

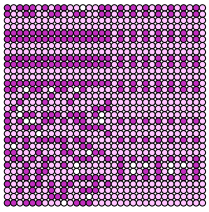
In all experiments the final fitness score was based on the organisms counting behaviour throughout its life time. The development process was apportioned 100 development steps. Each development step was set to include 100 state steps. The maximum size of the organism was set to 1024 cells in an array of 32 by 32 cells.

The experiments were executed on a cPCI machine including a PC running the GA. The development process and functional behaviour of the cellular array was executed on an FPGA (Tufte and Haddow, 2005).

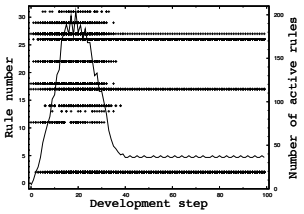
Experiment one: no Phenotypic Plasticity

The first experimental results are taken from Tufte and Haddow (2007a). In this work no environmental information was included in the gene regulation. The work compared the robustness of evolved organisms developed in environments with different degree of environmental fluctuations. The span of environments ranged from a single environment for all organisms to set of environments as used herein. The goal of the experiment was to evolve genomes that could develop into organisms that survived in different environments. This was archived by exposing the evolving organisms to different environments. The presented results are for comparison with results obtained by a development model that can include environmental information in the gene regulation.

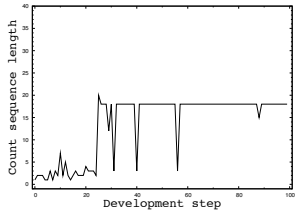
In Figure 3 the results of the experiment in Tufte and Haddow (2007a) are shown. The plots show the results for each of the ten runs. The fitness score of the respective evolved genome is plotted in black in



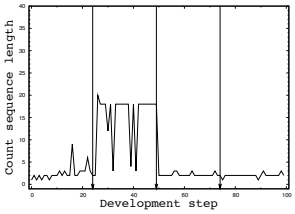
(a) Phenotype at DS 100.



(b) Gene activity.



(c) Behaviour with no environmental fluctuations.



(d) Behaviour with environmental fluctuations.

Figure 5: Comparing behaviour of a developing organism with and with out applying environmental changes. No environmental information in the gene regulation.

each run plot. The grey bars show the performance of the genome if re-developed in ten new randomly generated environments.

These genomes have not specialized to a given environment and their behaviour are quite similar for most of the environments the genomes was re-developed in. Some runs, i.e. *run1* and *run8*, show a short counting sequence. However, the deviations from the evolved results are low.

Experiment two: Including Phenotypic Plasticity

In this experiment environmental information was included in the gene regulation. The extension to include environmental information may be illustrated by changing the information available for the development process from the set-up in Figure 1(a) to 1(b).

The results of experiment two are shown in Figure 4. The plots for each run are obtained and presented in the same way as for the previous experiment.

The fluctuation in performance for some runs, e.g. *run5* and *run8*, is product of evolved dependency on specific environmental data. Such dependency can cause poor performing phenotype structures or competing counters cancelling out each other.

In Figure 7 the best evolved genomes, i.e. longest counter performance, of the two experiments are compared to the mean performance of the same genome developed in ten random environments. In addition the mean performance for all experiments developing in a random environment is shown. Comparison of the results for phenotypic plasticity vs. no plasticity shows an improvement for the best genomes including environmental information in the gene regulation when re-developed in a set of new environments. However, the mean of all runs shows an almost identical performance.

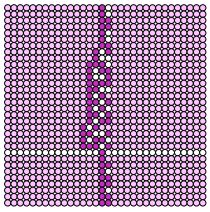
Experiment three: Exposure to Environmental Fluctuations during Development

The third experiments may be an extreme case for environmental fluctuations. The genomes evolved in experiment one and two are re-developed in an environment where fluctuations are enforced during development. External information is applied as an enforced random state to $1/4$ of the cells (empty or within the organism) available. The external changes in cell state are applied at an early stage of development (DS 25), in the middle of the organisms life time (DS 50) and at a late stage of development (DS 75). The cells influenced are defined as an array of 16 x 16 cells in the centre of the cellular array.

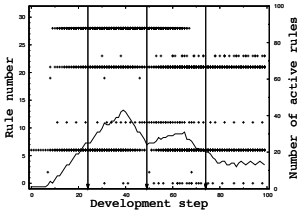
In nature most organisms of a given species develops in a rather uniform environment. The species has evolved within an environment where the species is a result of evolution and possible environmental changes over time. As such, large unpredicted fluctuations on the single individual level is not the main concern. However, if artificial cellular organism for computation are considered with the external information used as data. The external information enforced into the system is on the individual level, i.e. an organism as a computational machine.

Figure 5 presents the result of introducing changes enforced externally to a genome from experiment one (no phenotypic plasticity). The resulting phenotype is shown in Figure 5(a). Since the development model used in this experiment does not take the environmental information into the gene regulation the phenotype structure is equal for all environments. Figure 5(b) illustrates the gene activation for the presented phenotype. The plot presents the gene activation pattern during development together with the number of active rules at each development step. Rule numbers from 0 to 31 are placed on the left Y-axis. The mark (+) in the plot indicates that the rule was activated at the given development step. The right Y-axis shows the number of cells in the organism with an active rule on a given development step. The number of active rule cells is illustrated by the plotted line. The gene activity is constant for all environments.

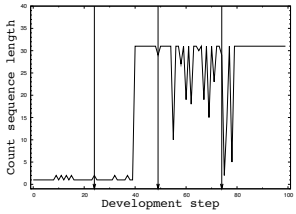
The plot in Figure 5(c) show the counting sequence length achieved at each development step with no enforced environmental changes. As shown the organism develops a fluctuation counting between DS 26 and 58 there after the counting sequence is stabile throughout the life time of the organism. If the genome develops in a changing environment the result is quit different as illustrated in Figure 5(d). The environmental changes are enhanced by the arrow lines. In



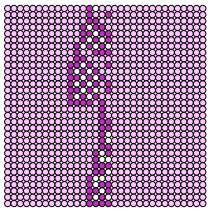
(a) Phenotype at DS 100 with environmental fluctuations.



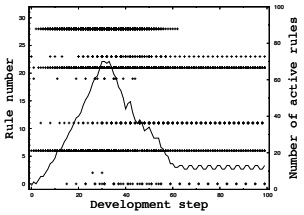
(b) Gene activity with environmental fluctuations.



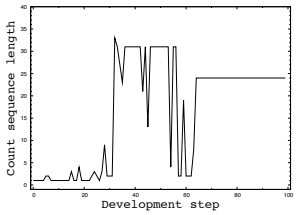
(c) Behaviour with environmental fluctuations.



(d) Phenotype at DS 100 no environmental fluctuations.



(e) Gene activity no environmental fluctuations.



(f) Behaviour no environmental fluctuations.

Figure 6: Comparing gene regulation and behaviour of a developing organism with and with out applying environmental changes. Environmental information included in the gene regulation.

the example the organism tolerate the first shift in the environmental information. However at DS 50 when the second change is enforced the organism's functionality is hardly present as the counting sequence drops. The last environmental change at DS 75 does not cause any major change in behaviour.

The result of applying external information during development for the development model capable of phenotypic plasticity is shown in Figure 6. In contrast to the results presented for no phenotypic plasticity the environment influence on gene regulation results in a possibility for environmental influence on the cellular composition of the phenotype. As such, the resulting phenotype depends on the environment present. Figure 6(a) show the phenotype developed in an environment with fluctuations. In Figure 6(b) the gene activation plot for the phenotype is presented. The enforced environmental fluctuations are illustrated by the arrow lines. The emergent counter sequence is presented in Figure 6(c).

To highlight the presence of environmental information in the gene regulation a candidate phenotype for the same genome developed in an initial random environment, i.e. developed with no enforced fluctuation, are shown in Figure 6(d). The corresponding gene activation plot for the shown phenotype is given in Figure 6(e). Figure 6(f) show the emergence of counting behaviour for the presented phenotype developed in the given random environment.

In contrast to the results presented for the development model with no environmental influence phenotypic plasticity can here be observed by the two different phenotypes arisen from the same genome. The source for the variation in phenotypic structure is the difference in gene activation caused by the extra environmental information. If the gene activity in Figure 6(b) and 6(e) are compared the affect of the fluctuations during development alter the timing of activation of different rules and the number of cells with active rules at different stages of the development of the organism.

The functionality of the organism given in Figure 6(c) and 6(f) show that the counting sequences are not identical but here the fluctuations introduced are not causing permanent damage to the functionality. The enforced changes may cause fluctuations in the counting sequence length but the developing organism achieves a stable behaviour.

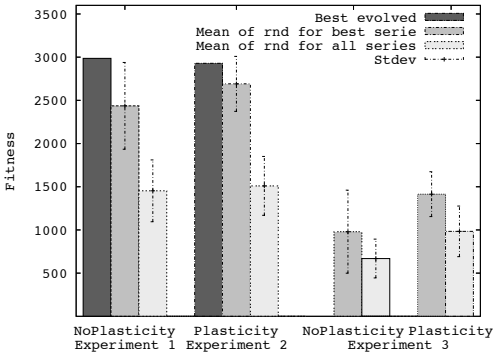


Figure 7: Experiments with possible phenotypic plasticity compared with the development model with no such feature.

In Figure 7 the result of introducing enforced environmen-

tal changes to the sets of best genomes are shown. In experiment three there are no evolved best result. As such, the mean of the best series and the mean of all runs with and without the possibility to exploit phenotypic plasticity are presented.

Conclusion

Including environmental information into the gene regulation mechanisms itself may be a way to achieve organisms that can respond to environmental changes during development. Organisms that can dynamically tune the cellular structure by development in an interplay with the environment. The environmental information is not only influencing the phenotypic structure but also included in the making of the behaviour, i.e. computation, of the artificial organism.

The results show a successful integration of evolution, development and environment toward adaptive organisms. Further, the introduction of additional external environmental information, during development, shows how a developmental system can dynamically respond and adapt. This adaptation is a result of the possibility to create dynamic phenotypes. Such phenotypes change their phenotypic structure as a response to external stimulus, here robust computational behaviour.

In experiment one and two the expansion of the development model to include environmental information found individual genomes that have an increased performance. However the general result of all runs in the experiments are almost identical. As stated, comparing these two results are difficult due to the change in search space and the extended regulation caused by the environmental information. As such, the fact that the inclusion of environmental information results in better individual solutions and that the EA was capable of keeping up the performance with the increased genome size indicate that the environmental information is exploitable.

References

Beiu, V., Yang, L. Quintana, J. M., and Avedillo, M. J. (2003). Vlsi implementations of threshold logic-a comprehensive survey. *IEEE Transactions on Neural Networks*, 14(5):1217–1243.

Bongard, J. C. and Pfeifer, R. (2003). *Morpho-functional Machines: The New Species (Designing Embodied Intelligence)*, chapter Evolving complete agents using artificial ontogeny, pages 237–258. Springer-Verlag.

Federici, D. (2004). Evolving a neurocontroller through a process of embryogeny. In *Simulation of Adaptive Behavior (SAB 2004)*, LNCS, pages 373–384. Springer.

Federici, D. and Downing, K. (2006). Evolution and development of a multi-cellular organism: Scalability, resilience and neutral complexification. *Artificial Life*, 12(3):381–409.

Gordon, T. G. W. and Bentley, P. J. (2005). Development brings scalability to hardware evolution. In *the 2005 NASA/DOD Conference on Evolvable Hardware (EH05)*, pages 272 –279. IEEE.

Haddow, P. C. and Tufte, G. (2000). An evolvable hardware FPGA for adaptive hardware. In *Congress on Evolutionary Computation(CEC00)*, pages 553–560. IEEE.

Hogeweg, P. (2000). Shapes in the shadow: Evolutionary dynamics of morphogenesis. *Artificial Life*, 6(1):85–101.

Lantin, M. and Fracchia, F. (1995). Generalized context-sensative cell systems. In *Proceedings of Information Processing in Cells and Tissues, University of Liverpool*, pages 42–54.

Larsen, E. W. (2004). *Environment, development, and Evolution Toward a Synthesis*, chapter 7 A View of Phenotypic Plasticity from Molecules to Morphogenisis, pages 117–124. MIT-Press.

Miller, J. F. (2003). Evolving developmental programs for adaptation, morphogenesis, and self-repair. In *Seventh European Conference on Artificial Life*, Lecture Notes in Artificial Intelligence, pages 256–265. Springer.

Miller, J. F. (2004). Evolving a self-repairing, self-regulating, french flag organism. In *Genetic and Evolutionary Computation (GECCO 2004)*, Lecture Notes in Computer Science, pages 129–139. Springer.

Quick, T., Dautenhahn, K., Nehaniv, C. L., and Roberts, G. (1999). On bots and bacteria: Ontology independent embodiment. In *ECAL '99: Proceedings of the 5th European Conference on Advances in Artificial Life*, pages 339–343, London, UK. Springer-Verlag.

Rothlauf, F. and Goldberg, D. E. (2003). Redundant representations in evolutionary computation. *Evolutionary Computation*, 11(4):381–415.

Shipman, R., Shackleton, M., Ebner, M., and Watson, R. (2000). Neutral search spaces for artificial evolution: a lesson from life. In *Artificial Life VII, 2000*.

Sipper, M. (1997). *Evolution of Parallel Cellular Machines The Cellular Programming Approach*. Springer-Verlag.

Spears, W. M. (1991). Gac ga archives source code collection webpage. http://www.aic.nrl.navy.mil/galist/src/.

Tufte, G. and Haddow, P. C. (2003). Identification of functionality during development on a virtual sblock fpga. In *Congress on Evolutionary Computation(CEC2003)*, pages 731–738. IEEE.

Tufte, G. and Haddow, P. C. (2005). Towards development on a silicon-based cellular computation machine. *Natural Computation*, 4(4):387–416.

Tufte, G. and Haddow, P. C. (2007a). Achieving environmental tolerance through the initiation and exploitation of external information. In *Congress on Evolutionary Computation(CEC2007)*. IEEE.

Tufte, G. and Haddow, P. C. (2007b). Extending artificial development: Exploiting environmental information for the achievement of phenotypic plasticity. In *7th International Conference on Evolvable Systems (ICES07)*, Lecture Notes in Computer Science, pages 297–308. Springer.

Viswanathan, S. and Pollack, J. (2005). How artificial ontogenies can retard evolution. In *Genetic and Evolutionary Computation (GECCO 2005)*. ACM.

Simulated Trust: Towards robust social learning

Dieter Vanderelst[1], René Ahn[2] and Emilia Barakova[2]

[1]University of Antwerp, Belgium, [2]Einhoven University of Technology, The Netherlands
dieter_vanderelst@emailengine.org

Abstract

Social learning is a potentially powerful learning mechanism to use in artificial multi-agent systems. However, findings about how animals use social learning show that it is also possibly detrimental. By using social learning agents act based on second-hand information that might not be trustworthy. This can lead to the spread of maladaptive behavior throughout populations. Animals employ a number of strategies to selectively use social learning only when appropriate. This suggests that artificial agents could learn more successfully if they are able to strike the appropriate balance between social and individual learning. In this paper, we propose a simple mechanism that regulates the extent to which agents rely on social learning. Our agents can vary the amount of trust they have in others. The trust is not determined by the performance of others but depends exclusively on the agents' own rating of the demonstrations. The effectiveness of this mechanism is examined through a series of simulations. We first show that there are various circumstances under which the performance of multi-agents systems is indeed seriously hampered when agents rely on indiscriminate social learning. We then investigate how agents that incorporate the proposed trust mechanism fare under the same circumstances. Our simulations indicate that the mechanism is quite effective in regulating the extent to which agents rely on social learning. It causes considerable improvements in the learning rate, and can, under some circumstances, even improve the eventual performance of the agents. Finally, some possible extensions of the proposed mechanism are being discussed.

The Ecology Of Social Learning

Throughout the animal kingdom individuals exploit information that has been gathered by others. Animals from invertebrates (reviewed in Leadbeater and Chittka, 2007; Leadbeater et al., 2006; Fiorito, 2001) to great apes and humans (e.g. Tomasello, 1999; Whiten et al., 2007; Bonnie et al., 2006) exhibit forms of social learning[1]. The widespread use of social learning among taxa is caused by its enormous ecological advantages in many circumstances (see for example Kendal et al., 2005; Coolen et al., 2005; Bonnie and Earley, 2007, and references therein). Evolution favored social learning because it might allow individuals to be flexible and adaptive learners while avoiding the dangers associated with individual exploration (Boyd and Richardson, 1988; Zentall, 2006). Ecologists typically stress the fact that individuals benefit from copying behavior from others because it saves them the costs of asocial learning (Laland, 2004). Indeed, Zentall (2006) remarked that the behavior of others has often already been shaped by its consequences and might therefore be assumed to be safe to copy.

Unsurprisingly, social learning comes in many flavors. Various forms of social learning have been identified (Zentall, 2006) and the underlying mechanisms range from fairly simple to utterly complex (Noble and Todd, 2002). However, when studying the dynamics and ecological properties of social learning one can ignore the differences in implementations and consider underlying exchange of information only (Coussi-Korbell and Fragaszy, 1995). This made it possible to evaluate the advantages of social learning in theoretical studies focusing on the game-theoretic aspects.

This line of theoretical research, supported by empirical findings in animal behavior, has shown that the advantage of social learning is by no means universal. Social learning is advantageous only if one takes certain precautions (Laland, 2004; Galef and Laland, 2005). The fundamental problem is that social learning can support the spread, acquisition and the persistence of maladaptive behavior (Giraldeau et al., 2002). This is because social learners re-use information gathered by others but do not collect new information themselves. Therefore, they are implicitly assuming that the information they gather from others is reliable. There are several circumstances under which this assumption does not hold (see Giraldeau et al., 2002; Laland, 2004; Leadbeater and Chittka, 2007, for reviews and references). Second hand information can be, amomg others, incomplete, outdated, biased or utterly wrong.

Instead of animals relying on social learning whenever they can, evidence clearly shows that they are somewhat reluctant to use social information unless there is a good reason to do so (Galef and Laland 2005, see Laland et al. 2005 for a short discussion of a striking example in sticklebacks).

[1]Here, on theoretical grounds, taken to include the use of public information. See Bonnie and Earley (2007) for a discussion.

Animals (including humans, see Koenig and Harris 2005) employ certain selection strategies to control the copying of behavior. This allows them to use social learning in an intelligent fashion, avoiding its potential pitfalls. Because of this, examples in which social learning leads to maladaptive behavior are rather scarce in the literature on animal behavior. The most clear examples of social learning supporting maladaptive behavior are obtained under experimental circumstances where the, usual adequate, strategies fail (e.g. Laland and Williams, 1998; Pongrcza et al., 2003). Such experiments can uncover the strategies adopted by animals.

Laland (2004) found that guppies were induced to take a longer, less efficient, route to a feeding site if others were doing this also. In contrast, single guppies learned quickly to take a shorter route. In the context of the experiment the socially transferred behavior was clearly maladaptive. However, in natural circumstances, choosing the same route as others is a good strategy since it protects against predation by forming shoals. The guppies' strategy to conform leads them to adopt longer routes but this is clearly an advantageous strategy if considered in the ecological context in which it evolved.

Opposed to the conformity bias observed in guppies, some experimental results show a selective use of social learning. Capuchin monkeys do not resort to social learning if the problems they are challenged with (e.g. opening a box) are easy to solve. In contrast, when faced with a difficult task they will copy the behavior of others more frequently (see Laland, 2004, for references and a discussion). Presumably, monkeys are more willing to use the potentially flawed social information if asocial learning is costly. This shows that these animals do not assume a priori that social information is correct and reliable (and thus worthwhile to copy). Instead they adopt a trade-off between learning socially and individually taking into account possible costs and gains. See Laland (2004) for more examples of selective social learning in animals.

While the literature on animals shows relatively few instances of maladaptive social learning under natural circumstances, humans, who rely far more on social learning (Tomasello, 1999) than any other animal, provide many more examples (Boyd and Richerson, 2006). Obvious candidates are the social transfer of tobacco and drug use, reducing fertility and endangering fetus development. But also other, less dramatic, socially transferred behavior could reduce fitness in humans.

Social Learning in Artificial Agents

Recently, different authors have begun to explore the use of social learning as a way of instruction in artificial multi-agent systems (e.g. Acerbi et al., 2007; Pini et al., 2007; Belpaeme et al., 2007; Noble and Franks, 2002; Alissandrakis et al., 2004).

In a multi-agent setting, artificial agents could search concurrently for a solution for a given problem (e.g. how to pick up food). Once a single agent has found a solution, this innovation could be copied by others and could propagate through the population. In this way, social learning could drastically reduce the total number of learning trials needed for a population of artificial agents to solve a problem (Pini et al., 2007). Innovations in groups of animals are known to spread in the same way (e.g. Bonnie et al., 2006; Bonnie and Earley, 2007; Leadbeater and Chittka, 2007).

Though this argument rightfully assumes that social learning has attractive properties, it was also argued above that this is certainly not true in all circumstances. In fact, as said, animal behavior data suggest that social learning should be only engaged in sparsely and with great caution (Laland, 2004; Galef and Laland, 2005; Leadbeater and Chittka, 2007).

Reasoning by analogy, we can hypothesize that a successful learning strategy for artificial agents should strike a careful balance between different types of learning. This suggests that the learning performance of artificial agents can be improved by mechanisms that restrict social learning to circumstances under which it is appropriate.

Experimental Setup

We have investigated the question how agents can balance social and individual learning by simulating a very simple world with a number of agents. The agents in this world have been equipped with a mechanism that regulates the extent to which they rely on social learning. The fundamental risk in social learning is to act on untrustworthy information. Therefore, we equip agents with the possibility to change the level of trust they have in the demonstrations of others. This in turn determines their reliance on social learning.

We investigate the learning behavior of the agents by comparing their performance in simulations for various conditions. In all conditions we consider two populations of agents that have the same cognitive architecture. The first population is born before the second one, and has therefore already acquired some level of experience in the simulated world when the second population is initiated. The experimental conditions modeled differ in two important respects: (1) the protective trust mechanism employed and (2) whether both populations must learn the same task, or different tasks.

The Trust Mechanism

All agents have the same cognitive architecture (schematically represented in figure 1) and operate in a world in which a limited number of percepts (situations) can arise. Agents can respond to each percept using one of limited set of actions. Once this action is performed, the world returns a reward to the agent. The agents learn both individually and socially which action to perform in response to each percept.

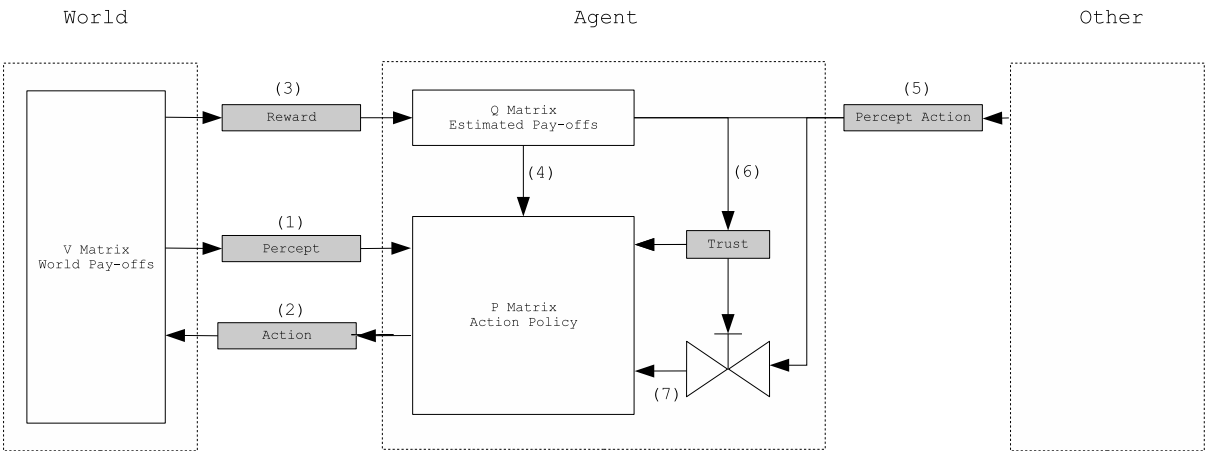


Figure 1: The cognitive architecture of the agents in the simulations and their relationship with the environment.

The behavior of our agents can be captured by a few simple rules. When learning individually this is what happens (the numbers correspond to the ones in figure 1):

- An $Agent$ is confronted (1) with a randomly chosen percept p.
- The $Agent$ chooses (2) an action a with which to respond to the percept p based on its policy **P**. The policy **P** defines the probability of an action a given a percept p.
- The world responds (3) to this action with the appropriate reward as given by the world pay-off function **V** .
- Based on this returned (3) reward, the estimated pay-off Q_{pa} for choosing the action given the percept is adapted.
- The $Agent$ updates (4) its policy **P**, effecting incremental changes to the probabilities for the various actions given the percept p, based on the changed estimates of the pay-offs.

When learning socially, this sequence of events take place:

- An $Agent$ observes (5) what an other agent perceives (percept) and how it reacts (action).
- Based on its own estimated pay-offs **Q** for the given percept, the $Agent$ updates (6) its trust in the observed other.
- The $Agent$ updates (7) its policy **P** for the given percept *dependent* on the trust it has in the other.

So, while learning individually, the agent builds an estimate Q_{pa} of the rewards obtained by executing each of the actions a when confronted with a percept p. This estimate determines its action policy.

During social learning, the agent copies the behavior of other agents whose actions it observes. However, the *extent* to which the behavior of others influences the agents' own policy, depends on the level of trust. The more an agent trusts the other, the more given observations will change its action policy **P**. Therefore, the trust level, which changes over time, regulates the extent to which agents rely on social learning.

Our agents increase the trust they have in others if the perceived behavior is in line with their own estimates of the rewards. If an agent perceives another responding to a percept with an action which itself thinks to be rewarding, the level of trust will rise. So, the more an agent sees others perform according to what it itself thinks is a rewarding policy, the more it will trust and copy them.

Simulations

Methods: Agents & World

In this section we describe the algorithm and settings of the simulations in detail.

The simulated world contains a fixed number of possible percepts. Agents can select one of small number of actions to respond to a given percept.

When an agent is confronted with a certain percept p it performs an action a. Subsequently, it receives a reward from the world. This reward is given a value V_{pa} stored in a matrix **V**. The values V_{pa} characterize the properties of the interaction between the agents and the world. In the present simulations, for any given percept p only one of the values $V_{p.}$ is set to 1 (see table 1 for examples). The others are set to -1. The action a for which $V_{pa} = 1$ determines which action an agent should perform when observing the percept p.

Each artificial agent has the same cognitive architecture (schematically represented in figure 1). These are the three central structures:

- a matrix **P** containing the current action policy,

- a matrix **Q** containing the pay-offs as estimated by the agent,
- a value T reflecting the trust level of the agent in others.

The matrix **P** gives, for each percept p and action a, the chance of an agent choosing this action a when confronted with this percept p (see equation 1). Agents are supposed to learn the optimal policy **P** that goes with the rewards as specified by matrix **V**. Matrix **P** is initialized with random values between 0 and 1 with the constraint that each row must sum to 1.

The matrix **Q** contains an estimate of the matrix **V** that is progressively constructed by the agent over the course of a simulation. This matrix is initialized containing only zeros.

The level of trust T of an agent is given by a value between 0 and 1. At the start of the simulation T is 1 which signifies that initial trust is total[2].

$$\mathbf{P} = \begin{pmatrix} P(a_1|p_1) & \dots & P(a_n|p_1) \\ \dots & \dots & \dots \\ P(a_1|p_m) & \dots & P(a_n|p_m) \end{pmatrix} \quad (1)$$

$$\mathbf{Q} = \begin{pmatrix} Q_{p_1 a_1} & \dots & Q_{p_1 a_n} \\ \dots & \dots & \dots \\ Q_{p_m a_1} & \dots & Q_{p_m a_n} \end{pmatrix} \quad (2)$$

In these simulations time is represented by an integer. At each time tick all agents are updated one by one (in a random order). In each cycle of the model each agent performs a single individual learning trial and may perform several social learning trials. This reflects the assumption that social learning is cheaper than individual learning. In the presented simulations, social learning does not restrict an agent's opportunity to learn individually. This will capture most biological (see Laland, 2004, for a discussion) and artificial situations to a certain extent. So, in our simulations, social learning is modeled as an additional learning method besides individual learning.

At each tick of the model all agents learn individually. Each agent is presented with a random percept. The agent selects one of the possible actions to respond to the percept. The chance $P(a|p)$ is given by the agent's matrix **P**.

After selecting an action p the world returns a reward V_{pa}. Based on this reward, the estimated pay-off Q_{pa} is updated. The update is governed by equation (3). In this equation α_Q is a step size parameter for updating the estimated pay-off matrix **Q**.

$$\Delta Q_{pa} = \alpha_Q (V_{pa} - Q_{pa}) \quad (3)$$

After updating the estimated pay-off matrix **Q**, the policy **P** is updated according to equation (4). The parameter α_I is the individual learning speed. Equation (4) augments the chance of picking action a given a percept p for which the estimated pay-off is currently the largest. Of course, it also decreases the chance of picking any of the other actions. This form of updating action policies is known in the literature on reinforcement learning as pursuit learning (Sutton and Barto, 1998).

$$\begin{cases} for\ a\ = arg\ max_a\ Q_{pa} : \Delta P(a|p) = \alpha_I(1 - P(a|p)), \\ \qquad\qquad \forall a' \neq a : \Delta P(a'|p) = \alpha_I(0 - P(a'|p)). \end{cases} \quad (4)$$

After updating its policy **P**, an agent stores p and a for later consultation by other agents during social learning.

After all agents have learned individually, all agents may perform several social learning trials. Whether they do so or not depends on the specific settings of the simulation (see later).

To learn socially, an agent randomly selects an agent to learn from and consults its latest action and percept. Social learning is modeled as a two-stage process. First, the observing agent updates its trust level. It consults the action a' and the percept p' stored by the observed other. The trust is updated based on the agents **Q** and **P** matrices according to equation (5). In this equation α_T is the step size for updating T. The trust values are constrained to lie between 0 and 1.

$$\Delta T = \begin{cases} -\alpha_T & \text{if } \sum_a [P(a|p) \times Q_{pa}] \leq Q_{p'a'}, \\ \alpha_T & \text{if } \sum_a [P(a|p) \times Q_{pa}] > Q_{p'a'}. \end{cases} \quad (5)$$

Second, after updating the trust level, the observing agent updates its value P_{pa} according to equation (6) with T denoting the trust level the agent has in the observed other. The parameter α_S is the step size governing social learning.

$$\begin{cases} for\ a\ : \Delta P(a|p) = \alpha_S \times T \times (1 - P(a|p)), \\ \forall a' \neq a : \Delta P(a'|p) = \alpha_S \times T \times (0 - P(a'|p)). \end{cases} \quad (6)$$

Selecting an agent to learn from socially is done in the following way. Each agent randomly selects a single agent to learn from and social learning is done as specified above. This is repeated 20 times. An agent can by chance choose an agent to learn from that it has choosen in a previous repetition. However, note that the behavior of this agent might have somewhat changed in the meantime because it has learned socially as well.

As experimenters we evaluate an agents policy by calculating the expected performance E according to equation (7).

$$E_i = \sum_p \sum_a P(a|p) \times V_{pa} \quad (7)$$

Note that in the current simulations no influence of the spatial distribution of the agents was incorporated.

[2]This is by no means essential for the behavior of the model. Results similar to the ones reported in the next section, were obtained by setting T, where appropriate, initially to 0.

	Actions							
	Values 1 ($\mathbf{V}_1$)				Values 2 ($\mathbf{V}_2$)			
	Actions				Actions			
Percepts	1	2	3	4	1	2	3	4
1	**1**	-1	-1	-1	-1	-1	-1	**1**
2	-1	**1**	-1	-1	-1	-1	**1**	-1
3	-1	-1	**1**	-1	-1	**1**	1	-1
4	-1	-1	-1	**1**	**1**	-1	-1	-1

Table 1: The two $\mathbf{V}_{pa}$ matrices in the form of tables used in the reported simulations.

Experimental Simulations

We ran various simulations to explore the properties of the model and to investigate whether and under what circumstances the trust mechanism protects learners against acquiring faulty information.

In these simulations, the agents are required to learn the profitable policies in a world where there are 4 possible percepts with 4 possible actions each. For each percept only one action has a good outcome ($V_{pa} = 1$).

Because we also want to experiment with situations where different populations need to perform different tasks, we need to define two different types of interaction with the world. This is done through two different reward matrices V_{pa} as given in Table 1.

All simulations are run for 200 time ticks. The simulations consist of two stages. First, an initial population (Population 1) of 21 agents is trained. After 50 ticks, another 21 agents (Population 2) are added to the population.

In each simulation Population 1 and 2 can either learn socially ($\alpha_S = 0.1$), individually ($\alpha_I = 0.1, \alpha_Q = 0.1$) or both ($\alpha_S = 0.1, \alpha_I = 0.1, \alpha_Q = 0.1$). If a learning strategy is not being used, the corresponding learning rate α is set to 0. If agents use social learning, they select 20 agents to learn from. This means that these agents have 20 social learning opportunities for each individual learning opportunity.

Simulations also differ with respect to the update of the trust value. Trust values could either be updated ($\alpha_T = 0.1$) or not ($\alpha_T = 0$).

An overview of the settings of the simulations can be found in table 2.

Simulation Results

The results of some of our simulations are plotted in figure 2. Figure 2(a) shows how performance changes over time, while figure 2(b) gives an insight into the development of trust values (where appropriate).

First, we want to demonstrate that our setting is indeed one where social learning can be advantageous. To this end we have simulated a situation where two populations need to perform the same task. Population 1 only learns individually, and population 2 also learns socially. The result are shown in simulation 1 of figure 2(a). As we can see from these results, when Population 2 is introduced into a population of reasonably instructed agents, social learning allows it to quickly catch up with them. Population 2 learns more rapidly than Population 1 by using both individual and social learning and catches up with them in about 20 time ticks.

In Simulation 2 we consider a situation where both populations learn individually and socially. This simulation shows that social learning is not advantageous under all circumstances. At the beginning of the simulation, the performance of Population 1 is actually hampered by the use of social learning. Population 1 learns slower in simulation 2 (with social learning) than in simulation 1 (without social learning). The reason for this is off course that, in simulation 2, the social learning process is also copying erroneous information.

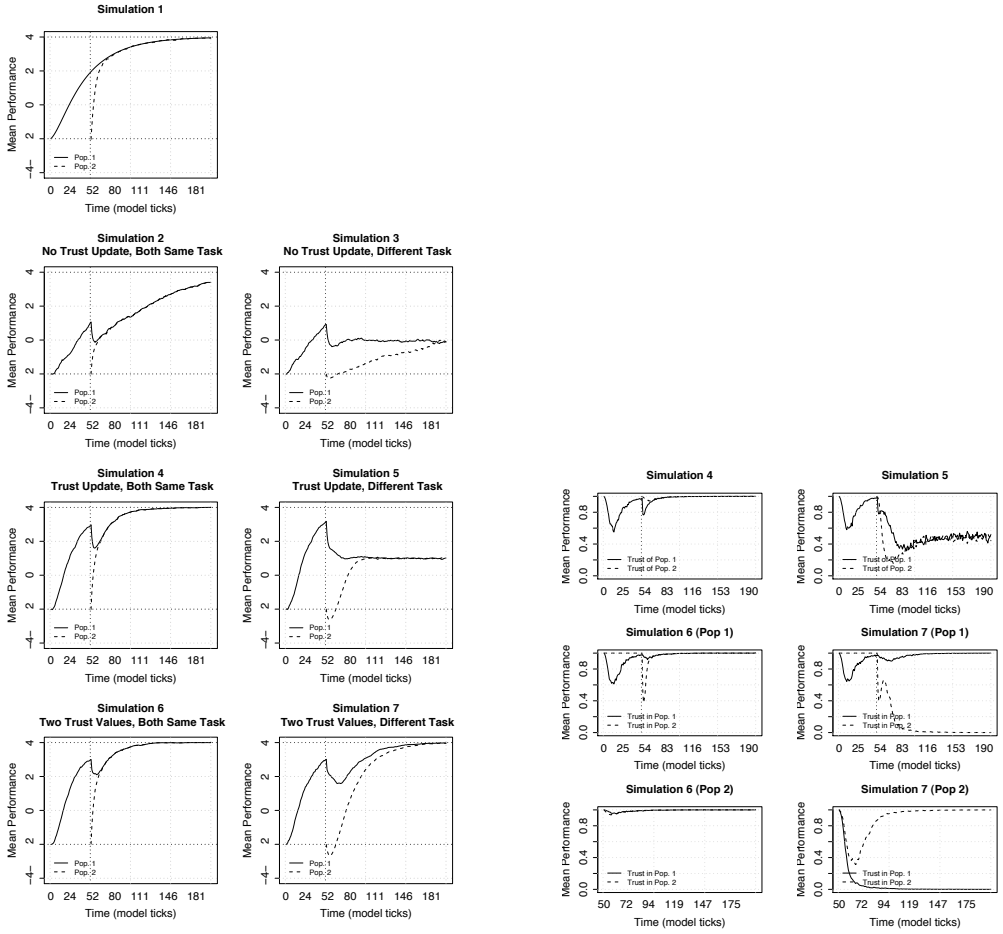
In simulation 3 we considered a situation where population 1 and population 2 have to learn different policies. As is to be expected, here the learning performance is even worse than in simulation 2. After the introduction of Population 2, the performance of Population 2 is actually decreased because it copies the flawed demonstrations of Population 1. Also, in contrast to what happens in simulation 2, Population 1 is now unable to regain its original level of performance because the more population 2 learns, the higher its faulty influence. In the end, the behavior of the two populations converges to a trade-off between the two optimal policies which is optimal for neither of them. The cause for the suboptimal performance in simulations 2 and 3 is that agents copy others even when these are not performing very well or even when they demonstrate a faulty policy. This is exactly what the trust mechanism is supposed to prevent.

The remaining simulations do incorporate different versions of the proposed trust mechanism. In simulation 4, we have copied the situation of simulation 2, but now both populations have a trust mechanism. As can be seen the mechanism is clearly advantageous to both populations.

To better understand what happens here, we plotted the dynamic behavior of the trust values in figure 2(b). Initially, Population 1 is trusting others (of the same population). However, agents quickly discover that demonstrations are not trustworthy. They respond by decreasing their trust level a bit (from 1.0 to 0.6). This allows agents to attain a performance level, through individual learning, at which their demonstrations are accurate enough to be trusted again. After about 20 ticks, the trust levels of the agents start to rise again re-enabling social learning to its full extent. A similar sequence of events is repeated at the introduction of Population 2. The trust levels are reduced to about 0.8 after which they rise again to 1.0. The trust mechanism makes sure that agents perform well enough before they start relying on social learning (again). This causes social learning to be used only if adequate. This significantly increases the learning speed (compare Population 1 in Simulation 1 and 4).

	Population 1				Population 2			
Simulation	Ind.	Soc.	Trust Update	$\mathbf{V}$	Ind.	Soc.	Trust Update	$\mathbf{V}$
Simulation 1	Yes	No	No	$\mathbf{V}_1$	Yes	Yes	No	$\mathbf{V}_1$
Simulation 2	Yes	Yes	No	$\mathbf{V}_1$	Yes	Yes	No	$\mathbf{V}_1$
Simulation 3	Yes	Yes	No	$\mathbf{V}_1$	Yes	Yes	No	$\mathbf{V}_2$
Simulation 4	Yes	Yes	Yes	$\mathbf{V}_1$	Yes	Yes	Yes	$\mathbf{V}_1$
Simulation 5	Yes	Yes	Yes	$\mathbf{V}_1$	Yes	Yes	Yes	$\mathbf{V}_2$
Simulation 6	Yes	Yes	Yes	$\mathbf{V}_1$	Yes	Yes	Yes	$\mathbf{V}_1$
Simulation 7	Yes	Yes	Yes*	$\mathbf{V}_1$	Yes	Yes	Yes*	$\mathbf{V}_2$

Table 2: The parameter settings in the seven simulations. When social or individual learning is used by a population in a given simulation the corresponding learning rate α is set to 0.1. $\mathbf{V}_1$ & $\mathbf{V}_2$ are given in table 1. *: agents store a separate trust value T for each population.



(a) (b)

Figure 2: The results of the simulations across 50 runs. Subfigure (a): The mean performance level in simulations 1-7. The vertical line denotes the time tick at which Population 2 is introduced in the model. The lower horizontal line gives the expected performance of an agent with a randomized **P** matrix (being -2). The upper horizontal line gives the maximum performance an agent can attain (being 4). Subfigure (b): The mean trust level in simulations 4-7. The vertical line denotes the time tick at which Population 2 is introduced in the model.

In simulation 5, we recreate the situation of simulation 3, but now with a trust mechanism. As one can see, the trust mechanism is not capable of completely solving the problems arising in simulation 3: the final performance of the agents in simulation 5 is slightly better than in 3, but still sub-optimal.

In the plot of the trust levels in figure 2(b), it can be seen that the trust levels converge to 0.5. This is caused by having half of the agents demonstrating a faulty policy and half a correct one. The agents cannot improve their performance because they cannot discriminate between trustworthy and untrustworthy agents.

Simulations 6 and 7 are identical to 4 and 5 but for the introduction of a separate trust value for each population. Every agent is equipped with two T values. This means that each agent can have a different level of trust in the members of Population 1 versus the members of Population 2.

Maintaining seperate trust values for the two populations has only a negligible effect in a situation in which both agents have to learn the same task (simulation 4). In simulation 6 the dip in the performance associated with the introduction of Population 2 is somewhat shallower than in simulation 4. Otherwise the results of simulation 4 and 6 are fairly similar. However, being able to discriminate between different types of agents allows the agents in simulation 7 to perform much better than in simulation 5. Now, both populations are able to achieve a perfect score.

Discussion

The results indicate that the proposed trust mechanism is capable of regulating the extent to which agents rely on social learning. Equipping our agents with the mechanism boosts their performance in situations where social learning is potentially disadvantageous (i.e situations in which demonstrations are untrustworthy).

Interestingly, the trust mechanism, as it is proposed in this paper, is a biologically plausible strategy. Humans, but also animals (e.g. Cheney and Seyfarth, 1988), learn more if they trust the source of information (See Carpenter and Call, 2007, for additional references). Koenig and Harris (2005) report experiments in which children from the age of 4 learned the names of novel objects from people who have shown to be trustworthy earlier in the experiment. They do not endorse names supplied by people who earlier misnamed known objects (e.g. naming a ball as a shoe). So, while adapting trust is a strategy that is, as yet, not widely studied in animal behavior, some empirical findings support that it is indeed being used. Further research might discover more instances in which trust is a important factor in human and animal social learning.

It is important to note explicitly that the presented trust mechanism differs from social learning strategies that seek to copy high performant demonstrators. For example, Schlag (1998) proposed that social learning agents (animals) should copy others if they are performing better than they are themselves (*copy-if-better*). However, this requires agents to be able to assess the performance of others, which might not be easy to do (Laland, 2004), especially for artificial agents. The form of trust introduced in the current paper does not require agents to evaluate the performance of others. Instead, agents trust others if they act in the same way as they would given the same percept. Simulation 7 serves as a demonstration of the difference between acting based on trust or the performance of others. In the second phase of the simulation (after tick 50), Population 1 is clearly performing better than Population 2. Nevertheless, Population 2 quickly looses its initial trust in Population 1 and stops copying its behavior. In contrast, Population 2 has more trust in itself. If performance would dictate social learning, all agents should be copying Population 1.

Finally, we think that much of the strength of the proposed mechanism lies in the fact that it can be extended in various interesting ways. We list two of the extensions we consider the most interesting.

First, a fundamental feature of the proposed trust mechanism is that it generalizes over all percept-action pairs. This is to say, an agent that learns to trust another by observing its response to a given percept p, also trusts the others response to all other percepts p'. This behavior is in concordance with the findings in children reported by Koenig and Harris (2005). Indeed, it is hard to see what would be the function of a trust mechanism that does not generalizes across stimuli. In the current simulations, this property of the model is not fully exploited. Generalizing across stimuli might enable agents, just like their biological counterparts, to learn socially about significant but rare stimuli. Some stimuli might not occur frequently enough for agents to learn individually from these instances. However, by observing how other agents, that are judged trustworthy, react to the stimuli, agents could assemble enough learning trials to associate a proper response with these stimuli. One possible extension of the presented work could explore the behavior and the value of the model under such circumstances

Another interesting extension, already hinted at in simulation 7, would be to increase the number of trust values that agents maintain. In the extreme case, an agent could have a trust value associated with each other agent in the population. Trust levels associated with individual agents would enable agents to form trust networks directing the flow of information that is spread through social learning (See Coussi-Korbell and Fragaszy, 1995, for a seminal paper on directed social learning). Also, agents could learn which individuals' behavior is worthwhile to copy (see Dautenhahn and Nehaniv, 2007).

In conclusion, we presented an extendable mechanism that allows agents to regulate their reliance on social learning. The mechanism to boost the performance of agents in multi-agent settings that incorporate social learning. Impor-

tantly, the mechanism does not require agents to be able to judge whether the actions of observed demonstrators have a favorable outcome.

References

Acerbi, A., Marocco, D., and Nolfi, S. (2007). Social facilitation on the development of foraging behaviors in a population of autonomous robots. In Costa, F. A., Rocha, L. M., Costa, E., Harvey, I., and Coutinho, A., editors, *ECAL*, volume 4648 of *Lecture Notes in Computer Science*, pages 625–634. Springer.

Alissandrakis, A., Nehaniv, C., and Dautenhahn, K. (2004). Towards robot cultures? learning to imitate in a robotic arm test-bed with dissimilarly embodied agents. *Interaction Studies*, 5:3–44.

Belpaeme, T., de Boer, B., and Jansen, B. (2007). The dynamic emergence of categories trough imitation. In Dautenhahn, K. and Nehaniv, C. L., editors, *Imitation in Animals and Artifacts*. MIT Press.

Bonnie, K. E. and Earley, R. L. (2007). Expanding the scope for social information use. *Animal Behaviour*, 74(2):171–181.

Bonnie, K. E., Horner, V., Whiten, A., and de Waal, F. B. M. (2006). Spread of arbitrary conventions among chimpanzees: a controlled experiment. *Proceedings in Biological Science*, 274(1608):367–372.

Boyd, R. and Richardson, P. J. (1988). An evolutionary model of social learning: the effect of spatial and temporal variation. In Zentall, R. R. and Galef, B. j., editors, *Social Learning: Psychological and Biological Perspectives*, pages 29–48. Erlbaum, Hillsdale, NJ.

Boyd, R. and Richerson, P. J. (2006). *The Innate Mind: Culture and Cognition*, chapter Culture, Adaptation, and Innateness. Oxford University Press.

Carpenter, M. and Call, J. (2007). The question of what to imitate: inferring goals and intentions from demonstrations. In Dautenhahn, K. and Nehaniv, C. L., editors, *Imitation in Animals and Artifacts*. MIT Press.

Cheney, D. and Seyfarth, R. (1988). Assessment of meaning and the detection of unreliable signals in vervet monkeys. *Animal Behaviour*, 36:477–486.

Coolen, I., Dangles, O., and Casas, J. (2005). Social learning in noncolonial insects? *Current Biolology*, 15(21):1931–1935.

Coussi-Korbell, S. and Fragaszy, D. M. (1995). On the relationship between social dynamics and social learning. *Animal Behaviour*, 50(6):1441–1453.

Dautenhahn, K. and Nehaniv, C. (2007). An agent-based per- spective on imitation. In Dautenhahn, K. and Nehaniv, C. L., editors, *Imitation in Animals and Artifacts*. MIT Press.

Fiorito, G. (2001). Socially guided behaviour in non-insect invertebrates. *Animal Cognition*, 4(2):69–79.

Galef, B. and Laland, K. (2005). Social learning in animals: Empirical studies and theoretical models. *Bioscience*, 55(6):489–499.

Giraldeau, L.-A., Valone, T. J., and Templeton, J. J. (2002). Potential disadvantages of using socially acquired information. *Philosophical Transactions of the Royal Society of London, Series B*, 357(1427):1559–1566.

Kendal, R., Coolen, I., van Bergen, Y., and Laland, K. (2005). Trade-offs in the adaptive use of social and asocial learning. *Advances in the Study of Behavior*, pages 333–379.

Koenig, M. A. and Harris, P. L. (2005). The role of social cognition in early trust. *Trends in Cognitive Sciences*, 9(10):457–459.

Laland, K., Coolen, I., and Kendal, R. (2005). Why not use public information? *Science*, 308:354–355.

Laland, K. N. (2004). Social learning strategies. *Learning & Behavior*, 32(1):4–14.

Laland, K. N. and Williams, K. (1998). Social transmission of maladaptive information in the guppy. *Behavioral Ecology*, 9(5):493–499.

Leadbeater, E. and Chittka, L. (2007). Social learning in insects–from miniature brains to consensus building. *Current Biology*, 17(16):R703–R713.

Leadbeater, E., Raine, N. E., and Chittka, L. (2006). Social learning: ants and the meaning of teaching. *Current Biology*, 16(9):R323–R325.

Noble, J. and Franks, D. W. (2002). Social learning mechanisms compared in a simple environment. In Proceedings of the Eighth International Conference on Artificial Life.

Noble, J. and Todd, P. M. (2002). Imitation or something simpler? modelling simple mechanisms for social information processing. In *Imitation in Animals and Artifacts*. MIT Press, Cambridge, MA.

Pini, G., Tuci, E., and Dorigo, M. (2007). Evolution of social and individual learning in autonomous robots. Ecal Workshop: Social Learning in Embodied Agents.

Pongrcza, P., dm Miklsia, Kubinyia, E., Toplb, J., and Csnyia, V. (2003). Interaction between individual experience and social learning in dogs. *Animal Behavior*, 65(3):595–603.

Schlag, K. H. (1998). Why imitate, and if so, how? a boundedly rational approach to multi-armed bandits. *Journal of Economic Theory*, 78(1):130–156.

Sutton, R. S. and Barto, A. G. (1998). *Reinforcement Learning: An Introduction*. MIT Press, Cambridge.

Tomasello, M. (1999). *The Cultural Origins of Human Cognition*. Harvard University Press, Harvard.

Whiten, A., Spiteri, A., Horner, V., Bonnie, K. E., Lambeth, S. P., Schapiro, S. J., and de Waal, F. B. M. (2007). Transmission of multiple traditions within and between chimpanzee groups. *Current Biology*, 17(12):1038–1043.

Zentall, T. R. (2006). Imitation: definitions, evidence, and mechanisms. *Animal Cognition*, 9(4):335–353.

A study of GasNet spatial embedding in a delayed-response task

Patricia A. Vargas, Ezequiel A. Di Paolo and Phil Husbands

Centre for Computational Neuroscience and Robotics (CCNR)
Department of Informatics
University of Sussex
Falmer, Brighton
BN1 9QH, United Kingdom
{p.vargas, ezequiel, p.husbands}@sussex.ac.uk

Abstract

GasNet artificial neural networks can be used as complex neurocontrollers involving virtual chemical neuromodulation as well as synaptic interaction. The aim of this paper is to further explore the role of space in GasNet models on a delayed-response robot task. Comparative results demonstrate that the use of spatial constraints is not a prerequisite for a good performance of the original model in terms of speed of evolution.

Introduction

Evolutionary robotics allows us to explore complex dynamical neural processes and architectures that connect to interesting issues in neuroscience (Nolfi and Floreano, 2004). The GasNet models can be considered as examples of such complex neurocontrollers involving chemical neuromodulation as well as synaptic interaction (Husbands, 1998). A recently devised non-spatial GasNet model named NSGasNet (Vargas et al., 2007) follows the same principles and had already been successfully applied as a robot controller where the task did not require the controller to have a non-reactive response (Moioli et al., 2008). This work attempts to further explore this novel model in a delayed-response robot task, in addition to compare it with the original GasNet model in terms of evolvability. In essence, our aim is to investigate whether the space embedding present in the original GasNet is the main explanation for its success when applied to more elaborate robot tasks.

We will present a comparison which follows the investigation started by Vargas and collaborators (Vargas et al., 2007). In that work, the NSGasNet model has proven to have higher evolvability with respect to the original model on a central pattern generator task (CPG).

However, it is unclear whether the conclusions obtained for the CPG task will carry over to more complex situations, especially to cases involving an embodied agent. For this reason, we decided to perform a comparative study on a well-researched delayed-response task involving a T-Maze (Husbands, 1998; Jakobi, 1993, 1997; Ulbricht, 1996).

Our results corroborate the fact that the use of spatial embedding is not a prerequisite for better performance either in terms of speed of evolution or in robustness. This might also indicate that the success demonstrated by GasNet models so far (Husbands et al., 1998; McHale and Husbands, 2004; Philippides et al., 2005) are not related to the spatial embedding of nodes but maybe to the temporal dynamics promoted by the gaseous diffusion amongst them.

We will start by briefly describing the original GasNet plus the novel model, together with a summary of the previous results on a CPG task. Thereafter, we will describe our experiment in detail including the respective network architecture and genetic encoding, together with the evolutionary regime. After the results section we will provide a discussion and propose future work.

Non-Spatial GasNet: NSGasNet

Since the introduction in 1943 of the first artificial neuron model proposed by McCulloch and Pitts (McCulloch and Pitts, 1943) most of the subsequent classical artificial neural networks (ANNs) architectures have employed numerical synaptic interaction between their neurons. However, recent findings in neuroscience have suggested the existence of chemical signaling by gases that would play the role of neurotransmitters (Gally et al., 1990). By drawing inspiration from these latest discoveries, the GasNet model was introduced by Husbands (1998) in an attempt to create a novel recurrent artificial neural network, which seeks to combine the electrical and chemical signaling onto a single network.

In the original GasNet model, the classical sigmoided output function $y = tanh(x)$ of each neuron at each time step is modulated by a transfer function parameter k which will define which curve from the family of eleven sigmoids $(x = [-4, 4])$ will be employed during the network's operation. The value of k is controlled by the concentration of diffusing transmitter gas at a node following the network dynamics dictated by the network's equations as described in Husbands (1998).

Almost all GasNet parameters and variables are under evolutionary control. The use of evolutionary computation techniques to evolve ANNs is of fairly recent origin (Signals et al., 1990; Whitley et al., 1990; Yao and Liu, 1997; Yao,

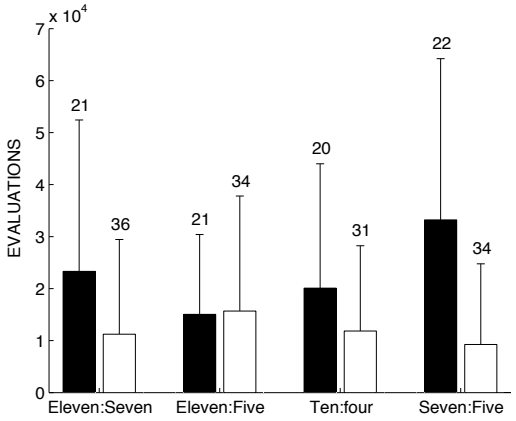


Figure 1: Mean and standard deviations (error bars) of fitness evaluations required to evolve successful networks for each CPG pattern, Eleven-Seven, Eleven-Five, Ten-Four and Seven-Five. Black bar shows original mean data and white bar shows NSGasNet mean data. The numbers above each error bar represent the total number of successfully evolved networks within 50 runs (adapted from (Vargas et al., 2007))

1999). Following this initiative, GasNet models were particularly designed to "evolve" for every task addressed. Hence, the network size, topology and almost all its parameters are under unconstrained evolutionary control.

Normally, depending on the task, the network is composed of a variable number of nodes. Thus, a network is encoded on a variable-sized genotype, where each gene represents a network node. A gene consists of an array of integer variables lying in the range [0, 99] (each variable occupies a gene locus). The decoding from genotype to phenotype obeys simple laws for continuous values and for nominal values (Husbands et al., 1998).

Vargas et al. (2007) introduced a novel spatially unconstrained GasNet named NSGasNet, in which the nodes do not have a location in a Euclidean space. Reminiscent of how the gas neurotransmitter NO normally diffuses once released (Gally et al., 1990; Wood and Garthwaite, 1996, 1994), in the NSGasNet model all emitted gases can spread freely among neurons.

NSGasNet is a discrete time recurrent neural network, which could be fully or partially connected with fixed or variable number of nodes. This full or partial connectivity refers to the synaptic connections. The gaseous connections are defined in terms of sensitivity limits, which impose to each network node a filter that regulates the strength of gas modulation (Vargas et al., 2007). Thus, each node has its set of sensitivity limits lying in the range [0, 1] of which values correspond to each node within the network. Although the NSGasNet has a bias that modulates the concentration of the gas at each node, the rules for how and when the gas is emitted are the same as the original GasNet (Husbands, 1998).

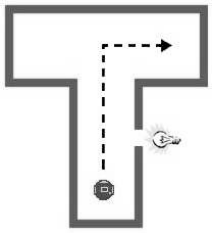


Figure 2: Schematic drawing of the robot and the T-Maze environment with two corridors. The robot is represented by the small circle and it is positioned in the bottom of the first corridor facing north. On the right-hand side there is a beam of light.

In a previous work by (Vargas et al., 2007), this non-spatial model has been successfully applied to a CPG task where the network should evolve to generate a sequence of cyclic output values from the set 0,1. Four patterns were tested and in all of them the NSGasNet was demonstrated to outperform the original spatially constrained GasNet Model in terms of speed of evolution (Figure 1). Some preliminary statistical analysis around mutants was performed to investigate the possible reasons for the best performance hypothesising about the role of the fitness landscape smoothness. A more profound analysis has been carried out in another work using further statistical correlation analysis between both models and the results will be submitted to publication soon. This work on the other hand intends to apply both network models to a more elaborate robot task to further assess the role of space in the performance of GasNet models.

Methods: T-Maze and Evolutionary Regime

T-Maze with light task

The experiment is a delayed response task in which a robot must learn to negotiate a T-Maze turning at the junction in the correct direction after passing a beam of light located in the first corridor either to the left or the right (Figure 2). Therefore, the robot must 'remember' the position of the light in order to successfully accomplish the task. This task, and similar ones, have been used by various researchers to endow artificial agents with minimal memory mechanisms (Husbands, 1998; Jakobi, 1993, 1997; Lanzi, 1998; Ulbricht, 1996; Webb et al., 2003); in this context it is interesting to note that it is still not well understood how biological memory works (Wilson, 1994; De Zeeuw, 2005; Levenson, 2006).

For this task we make use of a dedicated 2D robot simulator (Figure 3) of a Khepera II robot for the evolution of the GasNet models. The Khepera II robot has two wheels and two separate motors, 8 infra-red distance sensors (6 on the

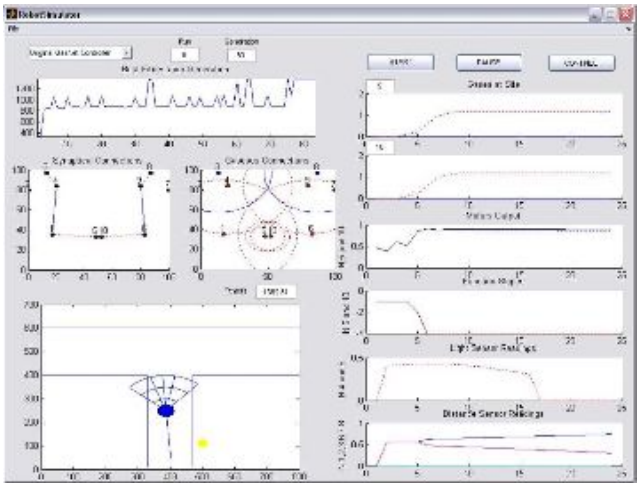


Figure 3: GUI of the robot simulator especially designed for the T-Maze delayed response experiment. On the left, from top to bottom, the interface shows a list to choose the GasNet model of interest (e.g. original or NSGasNet) and the specific run and generation; the best fitness per generation, the network architecture in terms of synaptic and gaseous connections and the robot within its arena. The right side of the interface shows in time from top to bottom: the values of the gas at site for two chosen nodes together with their function slopes, the values of the motor outputs, the values of the light sensor reading, and the values of the distance sensors reading

front and 2 on the rear) and 8 infra-red light sensors (6 on the front and 2 on the rear) .

The robot implemented in our simulator is a simplified model of a Khepera II robot and it was employed to avoid the overloading of graphical encoding in order to speed-up the simulations. It has 5 front distance sensors and 2 almost diametrically opposite light sensors (Figure 4(a)). While implementing the simulator, the two original front-most distance sensors were coupled (Figure 4(b)); hence, both sensor readings have the same value during the simulation. This was due to observations made during the design phase of the simulator where both sensors readings presented the same values most of the time.

Network Architecture, Genetic Encoding and Evolutionary Regime

Both GasNet models, original and NSGasNet, were implemented with a fixed number of nodes (total of 10 nodes). The networks are partially connected in addition to having genetically determined recurrent connections (Figure 5). Nodes 1, 2, 3, 6, 7 and 8 have input from the robot distance sensors S1, S2, S3, S5, S4 and S3, respectively. Nodes 4 and 9 have input from the left (L1) and right (L2) light sensors, respectively (Figure 8). Nodes 5 and 10 are responsible

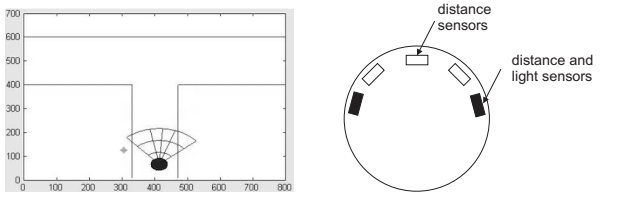


Figure 4: (a) Zoom of the T-Maze arena and the simulated robot (black round shape) localized at the bottom of the first corridor, facing north, and its five distance sensors stressing their range. Distance sensors were numbered from left to right: S1, S2, S3, S4 and S5 and light sensors: L1 - on the left side and L2 - on the right side. The arena is composed of two corridors forming a T-Maze and there is a beam of light (shaded star shape) shining from the left side of the robot. (b) Presents a schematic of the same robot illustrating the disposition of the distance and light sensors.

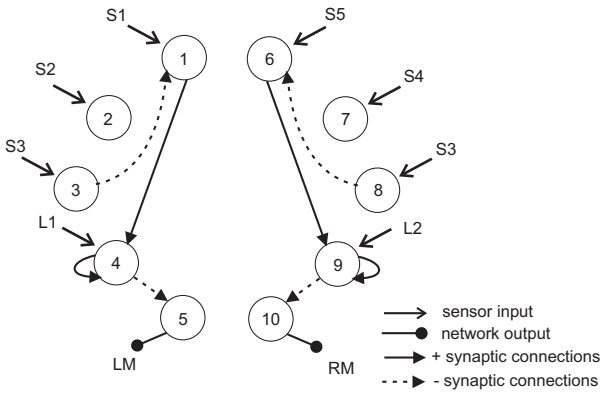


Figure 5: Pictorial example of a symmetrical partially connected ANN for the T-Maze task with ten nodes. The network receives external input from the sensors and supplies output to the motors.

for the output to the robot motors (left (LM) and right (RM) wheels, respectively).

Both networks have a symmetrical architecture meaning that for the genetic encoding we will only have to evolve half of the network. Hence, the original GasNet gene will have 65 parameters for the entire network, i.e. 13 parameters $times$ 5 nodes. Each node is coded as folows: $< gene >=< node >=< x >, < y >, < x1 >, < y1 >, < x2 >, < y2 >, < rec >, < Es >, < Gt >, < s >, < Gr >, < k0 > and < bias >$, where $< x >$ and $< y >$ are the node coordinates on the plane; $< x1 >, < y1 >, < x2 >, < y2 >$ specify the center of two circles on the network plane defining the node spatial electrical connectivity; $< rec >$ is the recurrent status; $< Es >$ is the emitting status; $< Gt >$ is the gas type; $< s >$ is the build up/decay rate; $< Gr >$ is the gas maximum radius of

Parameter	T-Maze task
Mutation rate	8%
Fitness function	$Fitness_{T-Maze} = d1 + d2 + bonus$
Number of runs	40
Maximum number of generations	150
Population size	100
Genotype size	65 (Original) 100 (NSGasNet)
Trials	10
Number of evaluations per trial	[70, 100]

Table 1: Evolutionary regime parameters employed on the T-Maze task.

emission,$< k0 >$ is the transfer function default value and $< bias >$ is the bias value (Husbands, 1998).

The NSGasNet genotype does not have parameters related to node coordinates, spatial electrical connectivity and maximum radius of emission, thus each NSGasNet gene will have 6 parameters for each node plus 10 values for the NSGasNet sensitivity limits (10 nodes), which makes the totality of 80 parameters for the entire network, plus 2 times the maximum number of allowed synaptic connections per node (to include the node number and the synaptic connection weight). For instance, if the maximum allowed number of synaptic connections per node is 2, than the NSGasNet genotype will have $80 + 5(2(2)) = 100$ variables.

The choice of partially interconnected networks for this task follows from previous works (Psujek et al., 2006; Williams and Noble, 2006) and also from the preliminary experiments on the T-Maze task where it was observed that full connectivity produced a negative impact on the evolvability of the networks for this particular task. The fully connected networks were too sensitive to genetic operations and initial conditions (e.g., the starting angle of direction) during the evolutionary process; therefore, a successful controller from one evaluation could hardly repeat its performance on the next fitness evaluation.

We employed a distributed steady-state genetic algorithm as described in (Husbands et al., 1998), who developed the idea from an early work using distributed populations (Hillis, 1990). The current population is updated steadily during the evolutionary process, i.e. each offspring is placed immediately into the current population (Whitley et al., 1990), instead of an entirely new population being generated and replacing the current population at a single time. Offspring were created through mutation operators (no recombination was used) with a probability of (8%) for each gene locus following a Gaussian distribution around its value for non-nominal values and a random value for nominal values. Non-nominal values refer to variables that have continuous values and nominal for discrete values.

In order to gather statistics 40 runs were performed for each model. One evolutionary run is composed of a maximum of 150 generations, or until successful genotypes are produced. Each generation comprises of 100 reproduction events or fitness evaluations.

The robot is tested for ten trials. Each trial is divided into two phases, following Jakobi's experiments set-up (Jakobi, 1997). The fitness value for phase 1 accounts for the distance d1 traveled by the robot in the first corridor ($d1max = 200$) and the fitness for the phase 2 is composed of the distance d2 traveled in the second corridor ($d2max = 180$) plus a bonus if the robot turns to the correct direction. The total fitness is the sum of the fitness at each trial divided by the total number of trials. The only difference from Jakobi's fitness calculation is the bonus value, which is computed as follows during the trials:

- 200 if the robot has turned to the correct side once;
- 500 if the robot has turned an equal number of times to both sides, plus:
  - +200 if the robot has turned four times to one side and four times to the other side
  - +500 if the robot has turned five times to one side and five times to the other side

Therefore the maximum fitness has a value around $1,380$ according to 1. This new bonus scheme was devise for it was observed that the evolution was very sensitive to the bonus criteria which imposes a selection pressure. Possibly this change was due to not implementing Jakobi's minimal simulations schema. Basically, this schema encompasses the addition of a controlled degree of noise and uncertainty during the evolution which will lead the robot to an improved robust behaviour when transferred to the reality. However, in these first robot experiments we are not concerned with the reality gap but with the measure of the evolvability of each GasNet model under noiseless circumstances. Therefore, we do not add noise to our simulations, just the start directional angle of the robot varies from trial to trial.

$$Fitness_{T-Maze} = d1 + d2 + bonus \quad (1)$$

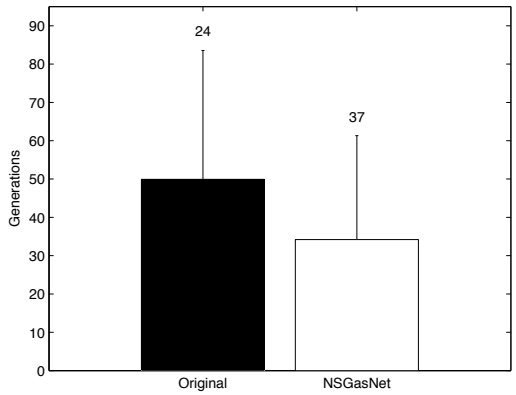


Figure 6: Mean and standard deviations (error bars) of generations required to evolve successful controllers for T-Maze with light task. Black bar shows original mean data and white bar shows NSGasNet mean data. The numbers above each error bar represent the total number of successfully evolved controllers within 40 runs.

Table 1 summarizes the parameter settings implemented within the evolutionary regime.

The successful evolution of a controller is considered if the robot obtains a fitness value that is greater than a threshold of 1,260 over seven subsequent trials. A robot with such fitness value has received the maximum $bonus = 1,000$ for having turned correctly in all 10 trials, plus the minimum distances traveled in both corridors, which when added may vary between [260, 380]. Runs that exceed 150 generations were aborted.

Results

The statistical results over 40 runs for each model are graphically illustrated in Figure 6. Black bar shows original mean data and white bar shows NSGasNet mean data. The numbers above each error bar represent the total number of successfully evolved controllers within 40 runs.

The NSGasNet outperforms the original GasNet model in terms of number of succesful runs. The frequency histograms portrayed at Figure 7 show that both distributions are skewed to the right, thus not symmetric, the difference between the mean and the median tend to spot a similar performance in terms of speed of evolution for the robot task between both models. However, the percentage of successfully evolved networks for the NSGasNet $(37/40 = 92\%)$ is greater than the original $(24/40 = 60\%)$.

Concerning the network architecture, in contrast to the NSGasNet model, the original model evolved less synaptic connections and more gaseous connections (Figure 8).

Many NSGasNet networks and some original ones did not make use of gases in the final evolved solution. When gases

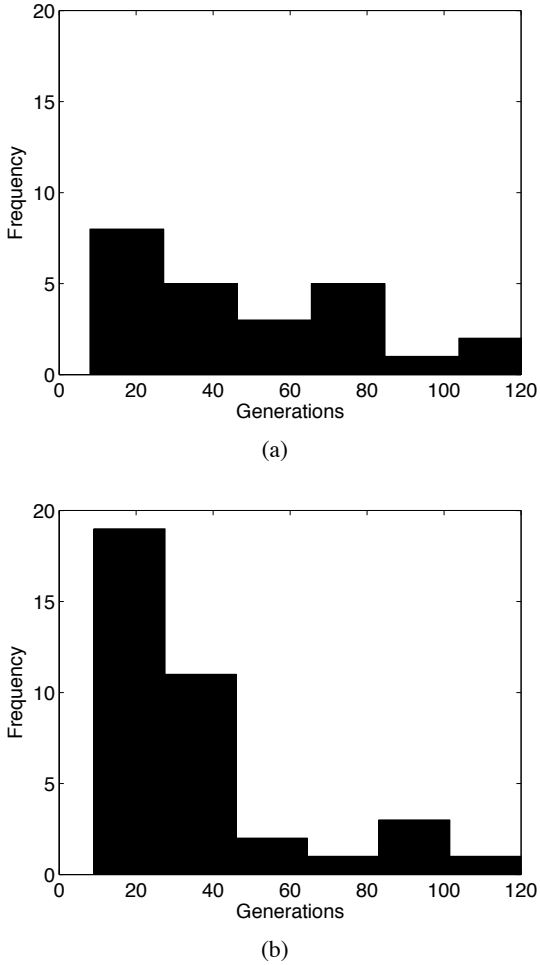


Figure 7: Frequency histograms comparison between the original (a) and the NSGasNet (b) models over the number of generations for the T-Maze task.

were at play, normally the nodes connected to the robot sensor lights had a coupled gaseous connection. Therefore, both nodes were making explicit use of gases to control the dynamics of each other and/or of other network nodes in response to environment changes, e.g. source of light (Figure 9).

An analysis of the behaviour of the robot shows that some of the successfully evolved controllers developed a reactive response to the task. For instance, the robot starts to follow the wall after passing the beam of light and thus, the robot is using the wall as an external memory, instead of creating an internal memory based on its internal state (Braitenberg, 1986; Nolfi, 2002). Naturally, this observation does not invalidate our evolvability results of the GasNet models. It only sheds some light on the potential requisite for an improved way to assess the robot behaviour during evolution, possibly in terms of a more elaborated fitness function.

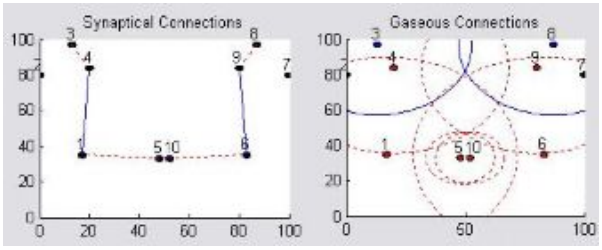


Figure 8: Picture of the simulation of an original GasNet successfully evolved controller, highlighting its synaptic and gaseous connections. There are few synaptic connections and intricate gaseous connections.

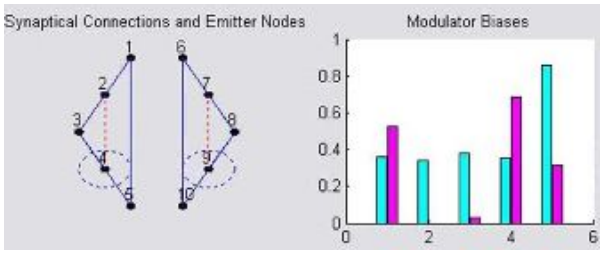


Figure 9: Screenshot of the simulation of a NSGasNet successfully evolved controller, highlighting its synaptic connections on the left and NSGasNet bias values for nodes 1, 2, 3, 4, and 5 on the right (the bars refer to nodes 4 and 9, respectively). Only nodes 4 and 9 are gas emitters. Remember that these are the nodes directly connected to the light sensors.

Discussion

This paper is a further step on the investigation of a novel non-spatial GasNet model (NSGasNet) in an attempt to uncover the role of space within this neural network paradigm. The performance of the original and the NSGasNet model was explored on a memory robot task. Unlike the previous results on a CPG task, the comparison between both models showed little difference in terms of speed of evolution. Although the evolvability values are quite similar they differ in the percentage of evolved controllers meaning that the NSGasNet has a higher success rate. Nonetheless, further analysis should be carried out in order to further assess this better performance.

Additional remarks could be made from the experiments. For instance, the use of partial connection between nodes was adopted in both models for the fully connected networks were too sensitive to genetic operations and initial conditions (e.g. the starting angle of direction) during the evolutionary process. Therefore, a successful controller from one evaluation could hardly repeat its performance on the next fitness evaluation, thus compromising its speed of evolution. One may argue that the problem might be the elevated mutation rate adopted. However, many mutation rates were tested and no improvement was observed. Thus, in order to make a compromise between evolvability and good performance, apart from the partial connection, we adopted 8% for the mutation rate.

It was observed that after evolution, some nodes either had their synaptic weights set to zero or there were no gaseous connections whatsoever. This fact shows the ability of the evolutionary process to find simple solutions to the problem and it also indicates that the introduction of metadynamics could improve the results. Metadynamics in this context means exploring a variety of network's dimensions during the evolutionary process. Therefore, in a future work we envisage using not only partially connected networks, but also exploring the network metadynamics. In our opinion, which is shared by others (Psujek et al., 2006), this coupling might lead to superior results.

According to (Strogatz, 2001) realistic networks have both nontrivial node dynamics and specific but irregular connection topologies. Moreover, highly distributed and non-hierarchical neural circuits had been identified in neuroscience investigations of simple organisms as pointed out by (Altman and Kien, 1990) and stressed by (Beer, 1995). Likewise, an analysis of the resulting network architectures for the T-Maze task has demonstrated a huge variety of topologies of connections (synaptic and gaseous) among the evolved controllers. This enormous variety was also verified by Vargas et al. (2007) for the CPG task. In both cases, it was impossible to identify a predominant pattern of connections and/or of spatial location of the nodes (in the case of the original GasNet model).

Internal state is not a pre-requisite for the agent to perform sophisticated interactions with the environment, as pointed out by (Izquierdo and Di Paolo, 2005; Nolfi, 2002; Stanley and Miikkulainen, 2002; Ziemke and Thieme, 2002). Accordingly, the fact that some of the robots presented a reactive response to the T-Maze task seems to indicate that the chosen task does not require a non-reactive response in order to be successfully accomplished.

In conclusion, the results obtained on this work together with the first investigations presented by (Vargas et al., 2007) seem to indicate that the explicit use of spatial constraints and a spatially embedded diffusion process is not necessary to explain the success of GasNet models. Rather, the interplay between two distinct processes (electrical signals and gas modulation) acting on different timescales, and the multiplicative modulation effect of the gases appear to be the important factors (Philippides et al., 2005).

In order to fully clarify the role of space within GasNet models, future work should include an analysis of the performance of both models in other tasks that require networks with higher dimension.

Acknowledgements

This work has been supported by the Spatially Embedded Complex Systems Engineering (SECSE) project, EPSRC grant no EP/C51632X/1.

References

Altman, J. S. and Kien, J. (1990). Highlighting aplysia's networks. *Trends in Neuroscience*, 13 (3): 81-82.

Beer, R. D. (1995). A dynamical system perspective on agent-environment interaction. *Artificial Intelligence*, 72:173:215.

Braitenberg, V. (1986). *Vehicles, Experiments in Synthetic Psychology*. MIT Press.

De Zeeuw, C. I. (2005). Time and tide in cerebellar memory formation. *Current Opinion on Neurobiology*, 15: 667-674.

Gally, J. A., Montague, P. R., Reeke, G. N., and Edelman, G. M. (1990). The no hypothesis: Possible effects of a short-lived, rapidly diffusible signal in the development and function of the nervous system. In *Proceedings of the National Academy of Sciences of the USA*, pages 87, 3547–3551.

Hillis, W. D. (1990). Co-evolving parasites improve simulated evolution as an optimization procedure. *Physica D*, 42:228–234.

Husbands, P. (1998). Evolving robot behaviours with diffusing gas networks. In *Evolutionary Robotics: First European Workshop, EvoRobot98*, pages 71–86. Springer-Verlag.

Husbands, P., Smith, T., Jakobi, N., and O Shea, M. (1998). Better living through chemistry: Evolving GasNets for robot control. *Connection Science*, 10:185–210.

Izquierdo, E. and Di Paolo, E. (2005). Is an embodied system ever purely reactive? *In M. Capcarrere et al (Eds.) Proceedings of the 8th European Conference on Artificial Life. pages 252-261. Springer-Verlag. LNAI 3630.*

Jakobi, N. (1993). Half-baked, ad-hoc, and noisy: Minimal simulations for evolutionary robotics. *In Husbands, P., and Harvey, I., (Eds.) Proc, Fourth European Conference on Artificial Life, MIT Press. Kaufmann, S. A. (1993). The Origins of Order. Oxford University Press.*, pages 348–357.

Jakobi, N. (1997). Evolutionary robotics and the radical envelope-of-noise hypothesis. *Adapt. Behav.*, 6(2):325–368.

Lanzi, P. L. (1998). Adding memory to xcs. In *Proceedings of the IEEE Conference on Evolutionary Computation (ICEC 98)*, pages 609–614. IEEE Press.

Levenson, J. M. (2006). Epigenetic mechanisms: a common theme in vertebrate and invertebrate memory formation. *Cellular and Molecular Life Sciences*, 63:1009-1016.

McCulloch, W. S. and Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. 265:115–133.

McHale, G. and Husbands, P. (2004). Quadrupedal locomotion: Gasnets, ctrnns and hybrid ctrnn/ pnns compared. In *In J. Pollack, M. Bedau, P. Husbands, T. Ikegami and R. Watson, editors, Alife X Proc. of the 9th Int. Conf. on the Simulation and Synthesis of Living Systems*, pages 106–112. MIT Press.

Moioli, R. C., Vargas, P. A., Zuben, F. J. V., and Husbands, P. (2008). Towards the evolution of an artificial homeostatic system. In *World Conference on Computational Intelligence 2008, WCCI'2008, Hong Kong, China*.

Nolfi, S. (2002). Power and limits of reactive agents. *Neurocomputing*, 42 (1): 119-145.

Nolfi, S. and Floreano, D. (2004). *Evolutionary Robotics: The Biology, Intelligence, and Technology of Self-Organizing Machines*. Bradford Book.

Philippides, A., Husbands, P., Smith, T., and O'Shea, M. (2005). Flexible couplings: Diffusing neuromodulators and adaptive robotics. *Artificial Life*, 11:139–160.

Psujek, S., Ames, J., and Beer, R. D. (2006). Connection and coordination: The interplay between architecture and dynamics in evolved model pattern generators. *Neural Comput.*, 18(3):729–747.

Signals, M., Fogel, S., Fogel, D., and Porto, L. (1990). Evolving neural networks. *Biological Cybernetics, 63*, pages 487–493.

Stanley, K. O. and Miikkulainen, R. (2002). Evolving neural networks through augmenting topologies. *Evolutionary Computation*, 10:99-127.

Strogatz, S. H. (2001). Exploring complex networks. *Nature*, 510:268-276.

Ulbricht, C. (1996). Handling time-warped sequences with neural networks. *In Proceedings of the Fourth International Conference on Simulation of Adaptive Behavior, MIT Press, 1996.*, pages 180–189.

Vargas, P. A., Di Paolo, E. A., and Husbands, P. (2007). Preliminary investigations on the evolvability of a non-spatial GasNet model. In *Proc. of the 9th European Conf. on Artificial life ECAL 2007*, pages 966–975. Springer-Verlag.

Webb, A., Hart, E., Ross, P., and Lawson, A. (2003). Controlling a simulated khepera with an xcs classifier system with memory. In *ECAL*, pages 885–892.

Whitley, D., Starkweather, T., and Bogart, C. (1990). Genetic algorithms and neural networks: Optimizing connections and connectivity. *Parallel Comput.*, 14:347–361.

Williams, H. and Noble, J. (2006). Homeostatic plasticity improves signal propagation in continuous-time recurrent neural networks. *Biosystems*, September 9:252–259.

Wilson, M. (1994). Reactivation of hippocampal ensemble memories during sleep. *Science*, 265.

Wood, J. and Garthwaite, J. (1994). Models of the diffusional spread of nitric oxide: Implications for neural nitric oxide signaling and its pharmacological properties. *Neuropharmacology*, 33:1235–1244.

Wood, J. and Garthwaite, J. (1996). Diffusion of free nitric oxide. *Methods in Enzymology*, 268:31–50.

Yao, X. (1999). Evolving artificial neural networks. *Proc. of the IEEE*, 87:1423–1447.

Yao, X. and Liu, Y. (1997). A new evolutionary system for evolving artificial neural networks. *IEEE Transactions on Neural Networks*, 8(3):694–713.

Ziemke, T. and Thieme, M. (2002). Neuromodulation of reactive sensorimotor mappings as a short-term memory mechanism in delayed response tasks. *Adaptive Behavior*, Vol. 10 No.3/4:175–99.

The Connectivity of NK Landscapes' Basins: A Network Analysis

Sébastien Verel[1], Gabriela Ochoa[2] and Marco Tomassini[3]

[1]University of Nice Sophia-Antipolis / CNRS, Nice, France
[2]University of Nottingham, Nottingham,UK
[3]University of Lausanne, Lausanne, Switzerland
verel@i3s.unice.fr, gxo@cs.nott.ac.uk, Marco.Tomassini@unil.ch

Abstract

We propose a network characterization of combinatorial £tness landscapes by adapting the notion of *inherent networks* proposed for energy surfaces (Doye, 2002). We use the well-known family of NK landscapes as an example. In our case the inherent network is the graph where the vertices represent the local maxima in the landscape, and the edges account for the transition probabilities between their corresponding basins of attraction. We exhaustively extracted such networks on representative small NK landscape instances, and performed a statistical characterization of their properties. We found that most of these network properties can be related to the search dif£culty on the underlying NK landscapes with varying values of K.

Introduction

Local optima are the very feature of a landscape that makes it rugged. Therefore, an understanding of the distribution of local optima is of utmost importance for the understanding of a landscape. Combinatorial landscapes refer to the £nite search spaces generated by important discrete problems such as the traveling salesman problem and many others. A property of some combinatorial landscapes, which has been often observed, is that on average, local optima are much closer to the optimum than are randomly chosen points, and closer to each other than random points would be. In other words, the local optima are not randomly distributed, rather they tend to be clustered in a "central massif" (or "big valley" if we are minimising). This globally convex landscape structure has been observed in the NK family of landscapes (Kauffman, 1993), and in other combinatorial optimization problems, such as the traveling salesman problem (Boese et al., 1994), graph bipartitioning (Merz and Freisleben, 1998), and ¤ow-shop scheduling (Reeves, 1999).

In this study we seek to provide fundamental new insights into the structural organization of the local optima in NK landscapes, particularly into the connectivity of their basins of attraction. Combinatorial landscapes can be seen as a graph whose vertices are the possible con£gurations. If two con£gurations can be transformed into each other by a suitable operator move, then we can trace an edge between them. The resulting graph, with an indication of the £tness at each vertex, is a representation of the given problem £tness landscape. A useful simpli£cation of the graphs for the energy landscapes of atomic clusters was introduced in (Doye, 2002; Doye and Massen, 2005). The idea consists in taking as vertices of the graph not all the possible con£gurations, but only those that correspond to energy minima. For atomic clusters these are well-known, at least for relatively small assemblages. Two minima are considered connected, and thus an edge is traced between them, if the energy barrier separating them is suf£ciently low. In this case there is a transition state, meaning that the system can jump from one minimum to the other by thermal ¤uctuations going through a saddle point in the energy hyper-surface. The values of these activation energies are mostly known experimentally or can be determined by simulation. In this way, a network can be built which is called the "inherent structure" or "inherent network" in (Doye, 2002).

We propose a network characterization of combinatorial £tness landscapes by adapting the notion of *inherent networks* described above. We use the well-known family of NK landscapes as an example because they are a useful tunable benchmark that can provide interesting information for more realistic combinatorial landscapes. In our case the inherent network is the graph where the vertices are all the local maxima and the edges account for transition probabilities between their corresponding basins of attraction. We exhaustively extract such networks on representative small NK landscape instances, and perform a statistical characterization of their properties. Our analysis was inspired, in particular, by the work of Doye (2002); Doye and Massen (2005) on energy landscapes, and in general, by the £eld of complex networks (Newman, 2003). The study of networks has exploded across the academic world since the late 90's. Researchers from the mathematical, biological, and social sciences have made substantial progress on some previously intractable problems, bringing new techniques, reformulating old ideas, and uncovering unexpected connections between seemingly different problems. We aim here at bringing the tools of network analysis for the study of problem

hardness in combinatorial optimization.
The next section describes how combinatorial landscapes are mapped onto networks, and includes the relevant definitions and algorithms used in our study. The empirical network analysis of our selected NK landscape instances is presented next, followed by our conclusions and ideas for future work.

Landscapes as Networks

To model a physical energy landscape as a network, Doye and Massen (2005) needed to decide £rst on a de£nition both of a state of the system and how two states were connected. The states and their connections will then provide the nodes and edges of the network. For systems with continuous degrees of freedom, the author achieved this through the 'inherent structure' mapping. In this mapping each point in con£guration space is associated with the minimum (or 'inherent structure') reached by following a steepest-descent path from that point. This mapping divides the con£guration space into basins of attraction surrounding each minimum on the energy landscape.

Our goal is to adapt this idea to the context of combinatorial optimization. In our case, the nodes of the graph can be straightforwardly de£ned as the local maxima of the landscape. These maxima are obtained exhaustively by running a best-improvement local search algorithm (*HillClimbing*, see Algorithm 1) from every con£guration of the search space. The de£nition of the edges, however, is a much more delicate matter. In our initial attempt Ochoa et al. (2008) we considered that two maxima i and j were connected (with an undirected and unweighted edge), if there exists at least one pair of solutions at Hamming distance one s_i and s_j, one in each basin of attraction (b_i and b_j). We found empirically on small instances of NK landscapes, that such de£nition produced densely connected graphs, with very low (≤ 2) average path length between nodes for all K. Therefore, apart from the already known increase in the number of optima with increasing K, no other network property accounted for the increase in search dif£culty. Furthermore, a single pair of neighbors between adjacent basins, may not realistically account for actual basin transitions occurring when using common heuristic search algorithms. These considerations, motivated us to search for an alternative de£nition of the edges connecting local optima. In particular, we decided to associate weights to the edges that account for the transition probabilities between the basins of attraction of the local optima. More details on the relevant algorithms and formal de£nitions are given below.

De£nitions and Algorithms

De£nition: Fitness landscape.
A landscape is a triplet (S, V, f) where S is a set of potential solutions i.e. a search space, $V : S \longrightarrow 2^S$, a neighborhood structure, is a function that assigns to every $s \in S$ a set of neighbors $V(s)$, and $f : S \longrightarrow R$ is a £tness function that can be pictured as the *height* of the corresponding solutions.

In our study, the search space is composed by binary strings of length N, therefore its size is 2^N. The neighborhood is de£ned by the minimum possible move on a binary search space, that is, the 1-move or bit-¤ip operation. In consequence, for any given string s of length N, the neighborhood size is $|V(s)| = N$.

The $HillClimbing$ algorithm to determine the local optima and therefore de£ne the basins of attraction, is given below:

Algorithm 1 *HillClimbing*

Choose initial solution $s \in S$
repeat
  choose $s^{'} \in V(s)$ such that $f(s^{'}) = max_{x \in V(s)}\ f(x)$
  if $f(s) < f(s^{'})$ **then**
    $s \leftarrow s^{'}$
  end if
until s is a Local optimum

De£nition: Local optimum.
A local optimum is a solution s^* such that $\forall s \in V(s^*)$, $f(s) < f(s^*)$.
The $HillClimbing$ algorithm de£nes a mapping from the search space S to the set of locally optimal solutions S^*.

De£nition: Basin of attraction.
The basin of attraction of a local optimum $i \in S$ is the set $b_i = \{s \in S \mid HillClimbing(s) = i\}$. The size of the basin of attraction of a local optima i is the cardinality of b_i.

De£nition: Edge weight.
Notice that for a non-neutral £tness landscapes, as are NK landscapes, the basins of attraction as de£ned above, produce a partition of the con£guration space S. Therefore, $S = \cup_{i \in S^*} b_i$ and $\forall i \in S\ \forall j \neq i, b_i \cap b_j = \emptyset$
For each solutions s and $s^{'}$, let us de£ne $p(s \rightarrow s^{'})$ as the probability to pass from s to $s^{'}$ with the bit-¤ip operator. In the case of binary strings of size N, and the neighborhood de£ned by the bit-¤ip operation, there are N neighbors for each solution, therefore:
if $s^{'} \in V(s)$, $p(s \rightarrow s^{'}) = \frac{1}{N}$ and
if $s^{'} \notin V(s)$, $p(s \rightarrow s^{'}) = 0$.
We can now de£ne the probability to pass from a solution $s \in S$ to a solution belonging to the basin b_j, as:

$$p(s \rightarrow b_j) = \sum_{s^{'} \in b_j} p(s \rightarrow s^{'})$$

Notice that $p(s \rightarrow b_j) \leq 1$.
Thus, the total probability of going from basin b_i to basin b_j is the average over all $s \in b_i$ of the transition probabilities to solutions $s^{'} \in b_j$:

$$p(b_i \rightarrow b_j) = \frac{1}{\sharp b_i} \sum_{s \in b_i} p(s \rightarrow b_j)$$

$\sharp b_i$ is the size of the basin b_i. We are now prepared to de£ne our 'inherent' network or network of local optima.

De£nition: Local optima network.
The local optima network $G = (S^*, E)$ is the graph where the nodes are the local optima [1], and there is an edge $e_{ij} \in E$ with the weight $w_{ij} = p(b_i \rightarrow b_j)$ between two nodes i and j if $p(b_i \rightarrow b_j) > 0$.
According to our de£nition of weights, $w_{ij} = p(b_i \rightarrow b_j)$ may be different than $w_{ji} = p(b_j \rightarrow b_i)$. Two weights are needed in general, and we have an oriented transition graph.

Empirical Basin and Network Analysis

The NK family of landscapes Kauffman (1993) is a problem-independent model for constructing multimodal landscapes that can gradually be tuned from smooth to rugged. In the model, N refers to the number of (binary) genes in the genotype (i.e. the string length) and K to the number of genes that in¤uence a particular gene (the epistatic interactions). By increasing the value of K from 0 to $N-1$, NK landscapes can be tuned from smooth to rugged. The k variables that form the context of the £tness contribution of gene s_i can be chosen according to different models. The two most widely studied models are the *random neighborhood* model, where the k variables are chosen randomly according to a uniform distribution among the $n-1$ variables other than s_i, and the *adjacent neighborhood* model, in which the k variables that are closest to s_i in a total ordering $s_1, s_2, \ldots, s_n$ (using periodic boundaries). No signi£cant differences between the two models were found in (Kauffman, 1993) in terms of global properties of the respective families of landscapes, such as mean number of local optima or autocorrelation length. Similarly, our preliminary studies on the characteristics of the NK landscape optima networks did not show noticeable differences between the two neighborhood models. Therefore, we conducted our full study on the more general random model.

In order to avoid sampling problems that could bias the results, we used the largest values of N that can still be analysed exhaustively with reasonable computational resources. We thus extracted the local optima networks of landscape instances with $N = 14, 16, 18$, and $K = 2, 4, 6, ..., N-2, N-1$. For each pair of N and K values, 30 randomly generated instances were explored. Therefore, the networks statistics reported below represent the average behaviour of 30 independent instances.

Basins of Attraction

Besides the maxima network, it is useful to describe the associated basins of attraction as these play a key role in search algorithms. Furthermore, some characteristics of the basins can be related to the optima network features. The notion of the basin of attraction of a local maximum has been presented before. We have exhaustively computed the size and number of all the basins of attraction for $N = 16$ and $N = 18$ and for all even K values plus $K = N-1$. In this section, we analyze the basins of attraction from several points of view as it is described below.

Global optimum basin size versus K. In Fig. 1 we plot the average size of the basin corresponding to the global maximum for $N = 16$ and $N = 18$, and all values of K studied. The trend is clear: the basin shrinks very quickly with increasing K. This con£rms that the higher the K value, the more dif£cult for an stochastic search algorithm to locate the basin of attraction of the global optimum

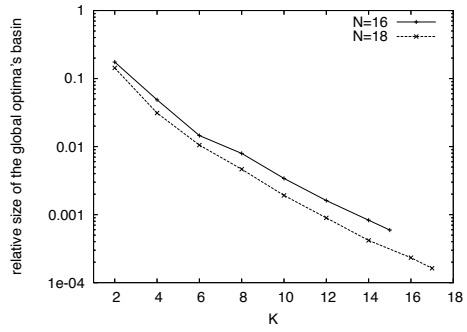


Figure 1: Average of the relative size of the basin corresponding to the global maximum for each K over 30 landscapes.

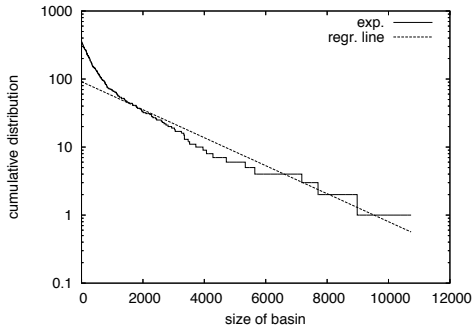


Figure 2: Cumulative distribution of the number of basins of a given size with regression line. A representative landscape with $N = 18$, $K = 4$ is visualized. A lin-log scale is used.

Number of basins of a given size. Fig. 2 shows the cumulative distribution of the number of basins of a given size (with regression line) for a representative instances with

[1] Since each maximum has its associated basin, G also describes the interconnection of basins.

Table 1: Correlation coef£cient ($\bar{\rho}$), and linear regression coef£cients (intercept ($\bar{\alpha}$) and slope ($\bar{\beta}$)) of the relationship between the basin size of optima and the cumulative number of nodes of a given (basin) size (in logarithmic scale: $\log(p(s)) = \alpha + \beta s + \epsilon$). The average and standard deviation values over 30 instances, are shown.

K	$\bar{\rho}$	$\bar{\alpha}$	β
	$N = 16$		
2	$-0.944_{0.0454}$	$2.89_{0.673}$	$-0.0003_{0.0002}$
4	$-0.959_{0.0310}$	$4.19_{0.554}$	$-0.0014_{0.0006}$
6	$-0.967_{0.0280}$	$5.09_{0.504}$	$-0.0036_{0.0010}$
8	$-0.982_{0.0116}$	$5.97_{0.321}$	$-0.0080_{0.0013}$
10	$-0.985_{0.0161}$	$6.74_{0.392}$	$-0.0163_{0.0025}$
12	$-0.990_{0.0088}$	$7.47_{0.346}$	$-0.0304_{0.0042}$
14	$-0.994_{0.0059}$	$8.08_{0.241}$	$-0.0508_{0.0048}$
15	$-0.995_{0.0044}$	$8.37_{0.240}$	$-0.0635_{0.0058}$
	$N = 18$		
2	$-0.959_{0.0257}$	$3.18_{0.696}$	$-0.0001_{0.0001}$
4	$-0.960_{0.0409}$	$4.57_{0.617}$	$-0.0005_{0.0002}$
6	$-0.967_{0.0283}$	$5.50_{0.520}$	$-0.0015_{0.0004}$
8	$-0.977_{0.0238}$	$6.44_{0.485}$	$-0.0037_{0.0007}$
10	$-0.985_{0.0141}$	$7.24_{0.372}$	$-0.0077_{0.0011}$
12	$-0.989_{0.0129}$	$7.98_{0.370}$	$-0.0150_{0.0019}$
14	$-0.993_{0.0072}$	$8.69_{0.276}$	$-0.0272_{0.0024}$
16	$-0.995_{0.0056}$	$9.33_{0.249}$	$-0.0450_{0.0036}$
17	$-0.992_{0.0113}$	$9.49_{0.386}$	$-0.0544_{0.0058}$

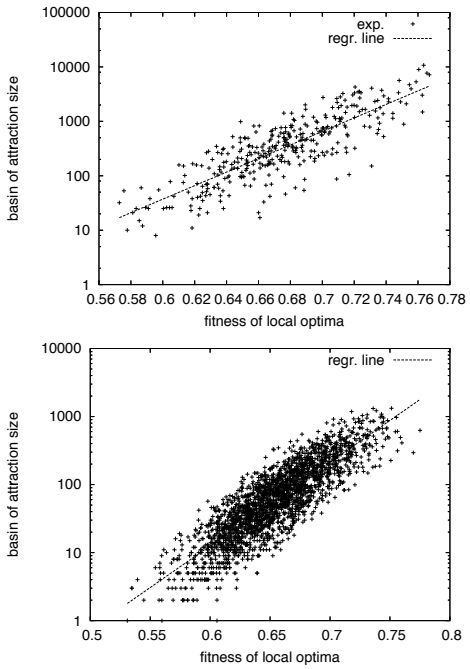


Figure 3: Correlation between the £tness of local optima and their corresponding basin sizes, for two representative instances with $N = 18, K = 4$ (top) and $K = 8$ (bottom).

$N = 18, K = 4$. Table 1 shows the average (of 30 independent landscapes) correlation coef£cients and linear regression coef£cients (intercept ($\bar{\alpha}$) and slope ($\bar{\beta}$)) between the number of nodes and the basin sizes for instances with $N = 16, 18$. Notice that distribution decays exponentially or faster for the lower K and it is closer to exponential for the higher K. This could be relevant to theoretical studies that estimate the size of attraction basins (see for example Garnier and Kallel (2001)). These studies often assume that the basin sizes are uniformly distributed, which is not the case for the NK landscapes studied here. From the slopes $\bar{\beta}$ of the regression lines (table 1) one can see that high values of K give rise to steeper distributions (higher $\bar{\beta}$ values). This indicates that there are fewer basins of large size for large values of K. Basins are thus broader for low values of K, which is consistent with the fact that those landscapes are smoother.

Fitness of local optima versus their basin sizes. The scatter-plots in Fig. 3 illustrate the correlation between the basin sizes of local maxima (in logarithmic scale) and their £tness values. Two representative instances for N = 18 and K = 4, 8 are shown. Notice that there is a clear positive correlation between the £tness values of maxima and their basins' sizes. In other words, the higher the peak the wider tend to be its basin of attraction. Therefore, on average, with a stochastic local search algorithm, the global optimum would be easier to £nd than any other local optimum. This may seem surprising. But, we have to keep in mind that as the number of local optima increases (with increasing K), the global optimum basin is more dif£cult to reach by a stochastic local search algorithm (see Fig. 1). This observation offers a mental picture of NK landscapes: we can consider the landscape as composed of a large number of mountains (each corresponding to a basin of attraction), and those mountains are wider the taller the hilltops. Moreover, the size of a mountain basin grows exponentially with its hight.

General Network Statistics

We now brie¤y describe the statistical measures used for our analysis of maxima networks.

The standard clustering coef£cient (Newman, 2003) does not consider weighted edges. We thus use the *weighted clustering* measure proposed by Barthélemy et al. (2005), which combines the topological information with the weight distribution of the network:

$$c^w(i) = \frac{1}{s_i(k_i - 1)} \sum_{j,h} \frac{w_{ij} + w_{ih}}{2} a_{ij} a_{jh} a_{hi}$$

where $s_i = \sum_{j \neq i} w_{ij}$, $a_{nm} = 1$ if $w_{nm} > 0$, $a_{nm} = 0$ if $w_{nm} = 0$ and $k_i = \sum_{j \neq i} a_{ij}$.

For each triple formed in the neighborhood of the vertex i, $c^w(i)$ counts the weight of the two participating edges of the

vertex i. C^w is de£ned as the weighted clustering coef£cient averaged over all vertices of the network.

The standard topological characterization of networks is obtained by the analysis of the probability distribution $p(k)$ that a randomly chosen vertex has degree k. For our weighted networks, a characterization of weights is obtained by the *connectivity and weight distribution* $p(w)$ that any given edge has weight w.

In our study, for each node i, the sum of weights from the node i is equal to 1. So, an important measure is the weight w_{ii} of self-connecting edges (remaining in the same node). We have the relation: $w_{ii} + s_i = 1$. The vertex *strength*, s_i, is de£ned as $s_i = \sum_{j \in \mathcal{V}(i) - \{i\}} w_{ij}$, where the sum is over the set $\mathcal{V}(i) - \{i\}$ of neighbors of i (Barthélemy et al., 2005). The strength of a node is a generalization of the node's connectivity giving information about the number and importance of the edges.

Another network measure we report here is *disparity* (Barthélemy et al., 2005) $Y_2(i)$, which measures how heterogeneous is the contributions of the edges of node i to the total weight (strength):

$$Y_2(i) = \sum_{j \neq i} \left(\frac{w_{ij}}{s_i} \right)^2$$

The disparity could be averaged over the node with the same degree k. If all weights are nearby of s_i/k, the disparity for nodes of degree k is nearby $1/k$.

Finally, in order to compute the average distance (shortest path) between two nodes on the optima network of a given landscape, we considered the expected number of bit-¤ip mutations to pass from one basin to the other. This expected number can be computed by considering the inverse of the transition probabilities between basins. In other words, if we attach to the edges the inverse of the transition probabilities, this value would represent the average number of random mutations to pass from one basin to the other. More formally, the distance (expected number of bit-¤ip mutations) between two nodes is de£ned by $d_{ij} = 1/w_{ij}$ where $w_{ij} = p(b_i \rightarrow b_j)$. Now, we can de£ne the length of a path between two nodes as being the sum of these distances along the edges that connect the respective basins.

Detailed Study of Network Features

In this section we study in more depth some network features which can be related to stochastic local search dif£culty on the underlying £tness landscapes. Table 2 reports the average (over 30 independent instances for each N and K) of the network properties described. $\bar{n}_v$ and $\bar{n}_e$ are, respectively, the mean number of vertices and the mean number of edges of the graph for a given K rounded to the next integer. $\bar{C}^w$ is the mean weighted clustering coef£cient. $\bar{Y}$ is the mean disparity, and $\bar{d}$ is the mean path length.

Clustering Coef£cients. The fourth column of table 2 lists the average values of the weighted clustering coef£cients for all N and K. It is apparent that the clustering coef£cients decrease regularly with increasing K for all N. For the standard unweighed clustering, this would mean that the larger K is, the less likely that two maxima which are connected to a third one are themselves connected. Taking weights, i.e. transition probabilities into account this means that either there are fewer transitions between neighboring basins for high K, and/or the transitions are less likely to occur. This con£rms from a network point of view the common knowledge that search dif£culty increases with K.

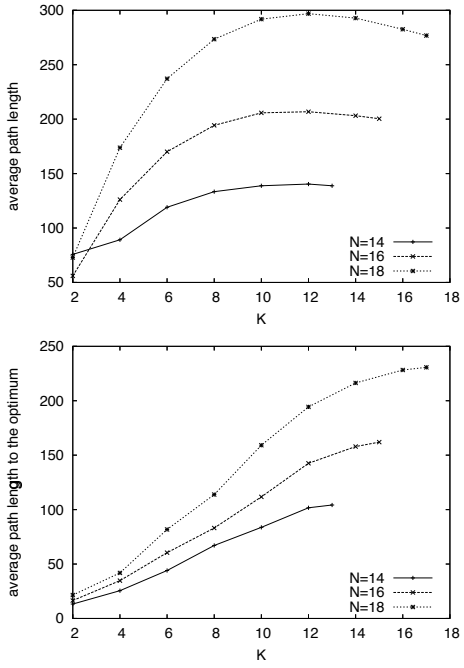


Figure 4: Average distance (shortest path) between nodes (top), and average path length to the optimum from all the other basins (bottom).

Shortest Path to the Global Optimum. The average shortest path lengths $\bar{d}$ are listed in the sixth column of table 2. Fig. 4 (top) is a graphical illustration of the average shortest path length between optima for all the studied NK landscapes. Notice that the shortest path increases with N, this is to be expected since the number of optima increases exponentially with N. More interestingly, for a given N the shortest path increases with K, up to $K = 10$, and then it stagnates and even decreases slightly for the $N = 18$. This is consistent with the well known fact that the search dif£culty in NK landscapes increases with K. However, some paths are more relevant from the point of view of a stochastic local search algorithm following a trajectory over the maxima network. In order to better illustrate the relationship of this network property with the search dif£culty by heuristic

Table 2: NK landscapes network properties. Values are averages over 30 random instances, standard deviations are shown as subscripts. n_v and n_e represent the number of vertexes and edges (rounded to the next integer), $\bar{C}^w$, the mean weighted clustering coef£cient. $\bar{Y}$ represent the mean disparity coef£cient, $\bar{d}$ the mean path length (see text for de£nitions).

K	$\bar{n}_v$	$\bar{n}_e$	C^w	Y	d
$N = 14$					
2	14_{6}	200_{131}	$0.98_{0.0153}$	$0.367_{0.0934}$	76_{194}
4	70_{10}	3163_{766}	$0.92_{0.0139}$	$0.148_{0.0101}$	89_{6}
6	184_{15}	12327_{1238}	$0.79_{0.0149}$	$0.093_{0.0031}$	119_{3}
8	350_{22}	25828_{1801}	$0.66_{0.0153}$	$0.070_{0.0020}$	133_{2}
10	585_{22}	41686_{1488}	$0.54_{0.0091}$	$0.058_{0.0010}$	139_{1}
12	896_{22}	57420_{1012}	$0.46_{0.0048}$	$0.052_{0.0006}$	140_{1}
13	1085_{20}	65287_{955}	$0.42_{0.0045}$	$0.050_{0.0006}$	139_{1}
$N = 16$					
2	33_{15}	516_{358}	$0.96_{0.0245}$	$0.326_{0.0579}$	56_{14}
4	178_{33}	9129_{2930}	$0.92_{0.0171}$	$0.137_{0.0111}$	126_{8}
6	460_{29}	41791_{4690}	$0.79_{0.0154}$	$0.084_{0.0028}$	170_{3}
8	890_{33}	93384_{4394}	$0.65_{0.0102}$	$0.062_{0.0011}$	194_{2}
10	$1,470_{34}$	162139_{4592}	$0.53_{0.0070}$	$0.050_{0.0006}$	206_{1}
12	$2,254_{32}$	227912_{2670}	$0.44_{0.0031}$	$0.043_{0.0003}$	207_{1}
14	$3,264_{29}$	290732_{2056}	$0.38_{0.0022}$	$0.040_{0.0003}$	203_{1}
15	$3,868_{33}$	321203_{2061}	$0.35_{0.0022}$	$0.039_{0.0004}$	200_{1}
$N = 18$					
2	50_{25}	1579_{1854}	$0.95_{0.0291}$	$0.307_{0.0630}$	73_{15}
4	330_{72}	26266_{7056}	$0.92_{0.0137}$	$0.127_{0.0081}$	174_{9}
6	994_{73}	146441_{18685}	$0.78_{0.0155}$	$0.076_{0.0044}$	237_{5}
8	$2,093_{70}$	354009_{18722}	$0.64_{0.0097}$	$0.056_{0.0012}$	273_{2}
10	$3,619_{61}$	620521_{20318}	$0.52_{0.0071}$	$0.044_{0.0007}$	292_{1}
12	$5,657_{59}$	899742_{14011}	$0.43_{0.0037}$	$0.038_{0.0003}$	297_{1}
14	$8,352_{60}$	1163640_{11935}	$0.36_{0.0023}$	$0.034_{0.0002}$	293_{1}
16	$11,797_{63}$	1406870_{6622}	$0.32_{0.0012}$	$0.032_{0.0001}$	283_{1}
17	$13,795_{77}$	1524730_{4818}	$0.30_{0.0009}$	$0.032_{0.0001}$	277_{1}

local search algorithms, Fig. 4 (bottom) shows the shortest path length to the global optimum from all the other optima in the landscape. The trend is clear, the path lengths to the optimum increase steadily with increasing K.

Weight Distribution Here we report on the weight distributions $p(w)$ of the maxima network edges. Fig. 5 shows the empirical probability distribution function for the cases $N = 16$ and $N = 18$ (logarithmic binning has been used on the x-axis). The case $N = 14$ is similar but is not reported here because it is much more noisy for $K = 2$ and 4 due to the small size of the graphs in these cases (see table 2).

One can see that the weights, i.e. the transition probabilities between neighboring basins are small. The distributions are far from uniform and, for both $N = 16$ and $N = 18$, the low K have longer tails. For high K the decay is faster. This seems to indicate that, on average, the transition probabilities are higher for low K.

Disparity Fig. 6 depicts the disparity coef£cient as de£ned in the previous section for $N = 16, 18$. An interesting observation is that the disparity (i.e. dishomogeneity) in the weights of a node's outcoming links tends to decrease steadily with increasing K. This re¤ects that for high K the transitions to other basins tend to become equally likely, which is another indication that the landscape, and thus its representative maxima network, becomes more random and dif£cult to search.

When K increases, the number of edges increases and the number of edges with a weight over a certain threshold increases too (see £g. 5). Therefore, for small K, each node is connected with a small number of nodes each with a relative high weight. On the other hand, for large K, the weights become more homogeneous in the neighbourhood, that is, for each node, all the neighboring basins are at similar distance.

If we suppose that edges with higher weights are likely to be connected to nodes with larger basins (an intuition that we need to con£rm in future work). Then, as the larger basins tend to have higher £tness (see Fig. 3), the path to higher £tness values would be easier to £nd for lower K than for larger K.

Boundary of basins. Fig. 7 shows the averages, over all the nodes in the network, of the weights w_{ii} (i.e the probabilities of remaining in the same basin after a bit-¤ip mutation). Notice that the weights w_{ii} are much higher when compared to those w_{ij} with $j \neq i$ (see Fig. 5). In particular, for $K = 2$, 50% of the random bit-¤ip mutations will produce a solution

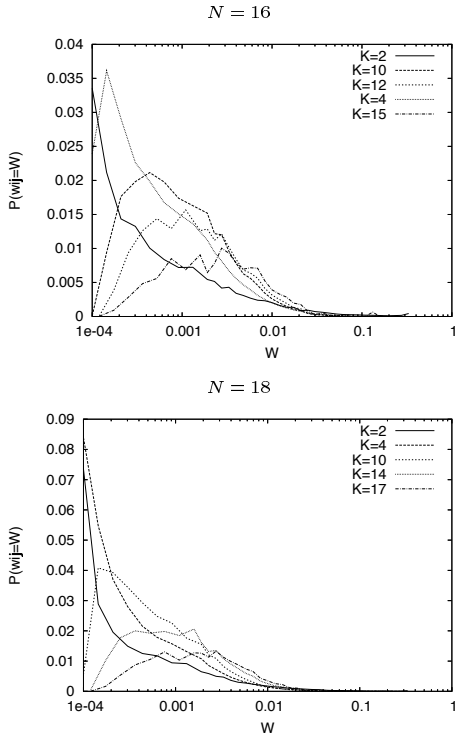


Figure 5: Probability distribution of the network weights w_{ij} with $j \neq i$ in logscale on x-axis. Averages of 30 instances for each N and K are reported.

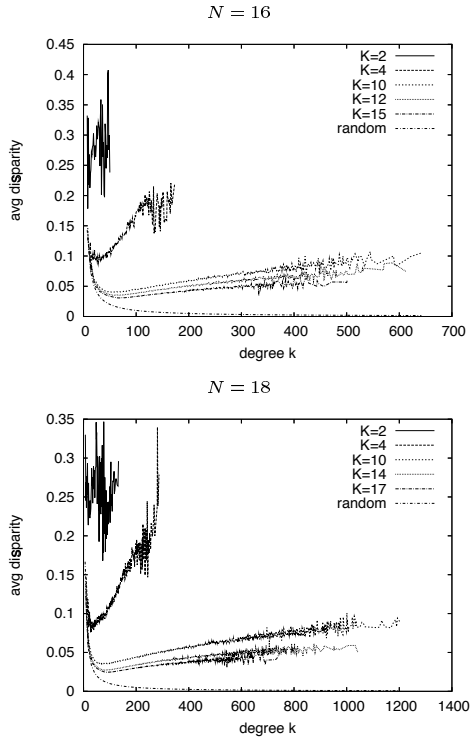


Figure 6: Average disparity, Y_2, of nodes with a given degree k. Average of 30 independent instances for each N and K are reported. The curve $1/k$ is also reported to compare to random case.

within the same basin of attraction. These average probabilities of remaining within the same basin, are above 12% for the higher values of K. Notice that the averages are nearly the same regardless the value of N, but decrease with the epistatic parameter K.

The exploration of new basins with the random bit-¤ip mutation seems to be, therefore, easier for large K than for low K. But, as the number of basins increases, and the £tness correlation between neighboring solutions decreases with increasing K, it becomes harder to £nd the global maxima for large K. This result suggests that the dynamic of stochastic local search algorithms on NK landscapes with large K is different than that with lower values of K, with the former engaging in more random exploration of basins.

The boundary of a basin of attraction can be de£ned as the set of con£gurations within a basin that have at least one neighbor's solution in another basin. Conversely, the interior of a basin is composed by the con£gurations that have all their neighbors in the same basin. Table 3 gives the average number of con£gurations in the interior of basins (this statistic is computed on 30 independent landscapes). Notice that the size of the basins' interior is below 1% (except for $N = 14, K = 2$). Surprisingly, the size of the basins' boundaries is nearly the same as the size of the basins themselves. Therefore, the probability of having a neighboring

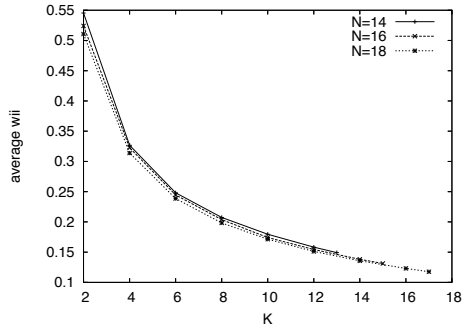


Figure 7: Average weight w_{ii} according to the parameters N and K.

solution in the same basin is high, but nearly all the solutions have a neighbor solution in another basin. Thus, the interior basins seem to be *"hollow"*, a picture which is far from the smooth standard representation of landscapes in 2D with real variables where the basins of attraction are visualized as real mountains.

Conclusions

We have proposed a new characterization of combinatorial £tness landscapes using the family of NK landscapes as an example. We have used an extension of the concept of inherent networks proposed for energy surfaces Doye (2002)

Table 3: Average (on 30 independent landscapes for each N and K) of the mean sizes of the basins interiors.

K	$N = 14$	$N = 16$	$N = 18$
2	$0.0167_{0.02478}$	$0.0050_{0.00798}$	$0.0028_{0.00435}$
4	$0.0025_{0.00065}$	$0.0012_{0.00025}$	$0.0006_{0.00010}$
6	$0.0029_{0.00037}$	$0.0014_{0.00015}$	$0.0007_{0.00009}$
8	$0.0043_{0.00045}$	$0.0022_{0.00012}$	$0.0011_{0.00006}$
10	$0.0055_{0.00041}$	$0.0031_{0.00015}$	$0.0018_{0.00006}$
12	$0.0061_{0.00041}$	$0.0040_{0.00014}$	$0.0025_{0.00007}$
13	$0.0059_{0.00029}$		
14		$0.0045_{0.00012}$	$0.0031_{0.00004}$
15		$0.0044_{0.00014}$	
16			$0.0035_{0.00005}$
17			$0.0034_{0.00006}$

in order to abstract and simplify the landscape description. In our case the inherent network is the graph where the nodes are all the local maxima and the edges accounts for transition probabilities (using the bit-¤ip operator) between the local maxima basins of attraction. We have exhaustively obtained these graphs for $N = \{14, 16, 18\}$, and for all even values of K, plus $K = N - 1$, and conducted a network analysis on them. Our guiding motivation has been to relate the statistical properties of these networks, to the search dif£culty of the underlying combinatorial landscapes when using stochastic local search algorithms (based on the bit-¤ip operator) to optimize them. We have found clear indications of such relationships, in particular:

The clustering coef£cients suggest that, for high values of K, the transition between a given pair of neighboring basins is less likely to occur.

The shortest paths increase with N and, for a given N, they clearly increase with higher K.

The weight distributions indicate that, on average, the transition probabilities are higher for low K.

The disparity coef£cients re¤ect that for high K the transitions to other basins tend to become equally likely, which is an indication of the randomness of the landscape.

The construction of the maxima networks requires the determination of the basins of attraction of the corresponding landscapes. We have thus also described the nature of the basins, and found that the size of the basin corresponding to the global maximum becomes smaller with increasing K. The distribution of the basin sizes is approximately exponential for all N and K, but the basin sizes are larger for low K, another indirect indication of the increasing randomness and dif£culty of the landscapes when K becomes large. Furthermore, there is a strong positive correlation between the basin size of maxima and their degrees. Finally, we found that the size of the basins boundaries is roughly the same as the size of basins themselves. Therefore, nearly all the con£gurations in a given basin have a neighbor solution in another basin. This observation suggests a different landscape picture than the smooth standard representation of 2D landscapes where the basins of attraction are visualized as hilltops.

This study is our £rst attempt towards a topological and statistical characterization of combinatorial landscapes, from the point of view of complex networks analysis. Much remains to be done. The results should be con£rmed for larger instances of NK landscapes. This will require good sampling techniques, or theoretical studies since exhaustive sampling becomes quickly impractical. Other landscape types should also be examined, such as those containing neutrality, which are very common in real-world applications. Finally, the landscape statistical characterization is only a step towards implementing good methods for searching it. We thus hope that our results will help in designing or estimating ef£cient search techniques and operators.

Acknowledgements. Thanks to Professor Edmund Burke for useful advice and support. This work was partially funded by EPSRC grant number EP/D061571/1.

References

Barthélemy, M., Barrat, A., Pastor-Satorras, R., and Vespignani, A. (2005). Characterization and modeling of weighted networks. *Physica A*, 346:34–43.

Boese, K. D., Kahng, A. B., and Muddu, S. (1994). A new adaptive multi-start technique for combinatorial global optimizations. *Operations Research Letters*, 16:101–113.

Doye, J. P. K. (2002). The network topology of a potential energy landscape: a static scale-free network. *Phys. Rev. Lett.*, 88:238701.

Doye, J. P. K. and Massen, C. P. (2005). Characterizing the network topology of the energy landscapes of atomic clusters. *J. Chem. Phys.*, 122:084105.

Garnier, J. and Kallel, L. (2001). Ef£ciency of local search with multiple local optima. *SIAM Journal on Discrete Mathematics*, 15(1):122–141.

Kauffman, S. A. (1993). *The Origins of Order*. Oxford University Press, New York.

Merz, P. and Freisleben, B. (1998). Memetic algorithms and the £tness landscape of the graph bi-partitioning problem. In *Parallel Problem Solving from Nature V*, volume 1498 of *Lecture Notes in Computer Science*, pages 765–774. Springer-Verlag.

Newman, M. E. J. (2003). The structure and function of complex networks. *SIAM Review*, 45:167–256.

Ochoa, G., Tomassini, M., and Vérel, S. (2008). A study of NK landscapes and basins and local optima networks. In *Genetic and Evolutionary Computation Conference, GECCO 2008, Proceedings*. ACM. to appear.

Reeves, C. R. (1999). Landscapes, operators and heuristic search. *Annals of Operations Research*, 86:473–490.

Adaptive Growth Processes: A Model Inspired by Pask's Ear

Nathaniel Virgo[1] and Inman Harvey

University of Sussex, UK
[1]nathanielvirgo@gmail.com

Abstract

We introduce the notion of an "adaptive growth process" in order to explain an experimental result from the 1950s in which a complex mechanism capable of distinguishing between two sounds emerges from a homogeneous chemical solution. We present a very simple computational model which exhibits an adaptive growth process. Adaptive growth processes could have practical applications in adaptive control systems and may also play a role in biological development.

keywords: self-organisation, complex systems, reinforcement learning, Gordon Pask

Introduction

In the late 1950s Pask (1958; 1960; 1961) was able to construct a device whereby a complex functional structure would emerge within a physical medium (a solution of ferrous sulphate), simply by increasing its supply of electrical current according to its performance at a given task. The task Pask set the device was to react to sound. The structure it produced resembled an ear, with an array of resonating metal threads which was able to distinguish between two different sounds. He was also able to get the same device to respond to changes in a magnetic field.

Unfortunately Pask did not record all the details of his setup and the experiment has never been repeated. Nothing quite like it has been achieved before or since. However, if the result can be repeated and generalised its potential would be enormous. It would mean that complex structures with functional components could be grown *in situ* without the need for an evolutionary population. For instance, one could imagine Pask's ear being used as a sensor in an adaptive robot. If the robot found itself in a situation where magnetic fields were relevant it could find itself adapting its ear to respond to them. If Pask's result is sufficiently general one could go further and imagine a neural controller that is able to grow adaptively, creating new neural structures in response to novel challenges. It is clear from Pask (1960) that Pask thought along similar lines.

In this paper we elaborate on the possibilities that this simple experiment opens up. We present our ideas on the general principles behind the result, defining a general notion of an *adaptive growth process*. We back this up with a simplified computational model of an adaptive growth process which, although it does not perform as impressive a task as Pask's ear, works in what we believe is a similar way.

Our system resembles a model of ants leaving pheromone trails to reach a target, except that we do not selectively reinforce the trail of a successful ant. Instead the reward for reaching the target is applied to the system as a whole, in the form of an increase in the rate at which ants enter the system. This stems from a difference in focus between our model and typical ant or swarm based models. Our model is intended to illustrate a plausible mechanism behind the success of Pask's experiment, in which individual threads were not reinforced and in which the solution consisted of a network of many interacting threads.

We also explore the possible biological implications of this kind of emergent adaptive structure.

Pask's experiments

The electrochemical experiment in which the ear was produced is mentioned in Pask (1958; 1960; 1961) but is always presented as an example to back up a philosophical point rather than as an experimental result in itself and it is difficult to decipher the exact experimental conditions which were used. There is also the eyewitness account of Stafford Beer described in Bird and Di Paolo (2008), although it was given many years after the event and does not give a complete description. Further descriptions of Pask's experiment and the history behind it can also be found in Bird and Di Paolo (2008) and Carriani (2007). There is a photograph of the resulting "ear" structure in Pask (1958), which is reproduced in Cariani (2007).

Although the details are obscure the basic idea is clear. Pask was experimenting with passing an electric current through a solution of ferrous sulphate. This causes a thin metal wire to be deposited along the path of maximum current. Since these threads have a very low resistance they affect the electric field surrounding them and thus the growth of further threads. Such systems were a popular way to

model neural growth at the time.

By varying the current flow through an array of electrodes Pask was able to affect the growth of the threads. For instance, when activating one negative and two positive electrodes a wire forms that starts at the negative electrode and branches toward the positive ones. If one of the positive electrodes is switched off the wire moves so that both branches point towards the remaining positive electrode, with the branching point remaining stable. If part of the wire is then removed the gap gradually moves toward the positive electrode, with the wire ahead of the gap being dissolved by the acidity of the solution but the wire behind growing back in its place. The branching pattern is reproduced by the new wire. Details of this can be found in Pask (1960).

Pask then took his electrochemical system and subjected it to sound. At this point the details become less clear but the system was rewarded in some way by an increase in available current whenever it responded to the sound, presumably by forming connections between a particular set of the electrodes. After about half a day a structure was formed which was able to perform this task, and Pask then went on to train it to distinguish between two tones, one about an octave above the other.

The interesting thing is that this structure is fairly complex, with functionally differentiated parts that are not specified by the experimenter. In Pask's words, "the ear, by the way, looks rather like an ear. It is a gap in the thread structure in which you have fibrils which resonate with the excitation frequency" (Pask (1960)). This solution is truly system-level. It is not determined simply by the fittest individual thread but by a combination of many threads playing several different roles. Some act as vibrating fibrils while others are part of the supporting structure. Presumably the fibrils can become further specialised to vibrate when exposed to sound in a particular frequency range. This spontaneous division of labour is not a feature of most learning algorithms.

Pask's Ear as a Dissipative Structure

The system of metallic threads that forms in Pask's ear is a dissipative structure. That is to say, it is a kind of structure that exists as the result of an externally imposed flow of energy through a system. The threads can only form and persist when an electrical current is passed through the medium, and if the current stops the acidity of the ferrous sulphate solution will gradually dissolve them. The structure maintains its form through a balance of creative and destructive processes.

This seems to us to lie at the heart of its operation. The system is subject to continual fluctuations, in which filaments are lengthened or shortened, or new ones grown and old ones destroyed. But the threads are in competition with each other: a given amount of electrical current can only support a certain amount of filament, so any new structure which increases the amount of metal thread can only become stable and be maintained if it causes a sufficient increase in the current flow.

Adaptive Growth Processes

We use the term *adaptive growth process* to refer to a system which operates in this way, where structures compete for some resource and those that contribute towards maintaining an increased supply of that resource tend to be more stable over time.

This definition may be refined at a later date but some general preconditions for an adaptive growth process are that there is a substrate and a resource whose availability depends in some way on the state of the system. The substrate is such that without any inflow of the resource it will decay to a homogeneous state, but that an inflow of resource enables structures to persist in the system. The dynamics of the system must be such that these structures compete for the resource and are continually subject to fluctuations, resulting in structures that contribute to an increased resource flow becoming more likely to persist over time than those that do not.

Of course, this does not happen in all circumstances in all possible substrates. This research is a first step towards identifying the specific requirements for an adaptive growth process to take place. Some insights gained from our model can be found in the discussion section.

One interesting feature of adaptive growth processes is that they can in some circumstances become more complex: Pask's ear presumably starts with a fairly simple network of threads and ends up with a relatively complex arrangement of resonating fibres. This increase in complexity occurs when an increase in resource flow can be achieved by a more complex structure. Since an increase in complexity will usually require an increase in resource use (to maintain a larger amount of metal thread, for instance) the structure will not generally become more complex than it needs to be.

Relationship to Reinforcement Learning and the Credit Assignment Problem

Reinforcement learning is a general term for a learning algorithm which can learn a task (usually a classification task) by being given a "reward signal" whenever it behaves in the desired way. Pask's ear can be seen as an example of this, with the supply of electric current acting as the reward signal. In general the supply of resource to an adaptive growth process acts as a reward signal.

From a reinforcement learning point of view, an interesting feature of Pask's ear and adaptive growth processes in general is that they avoid the so-called Credit Assignment problem. This is simply the problem of deciding which part of the system to reward for a successful behaviour. Since behaviour results from the dynamics of the system as a whole it can be hard or impossible to know which parts contribute

to a given behaviour and which detract from it. Pask's ear solves this problem in a very simple way: the reward is applied to the whole system in the form of an increase in resource availability. Since the system's parts are in competition with each other it is only the ones which contribute to this increased resource availability that will remain stable in the long run.

Relationship to Evolution by Natural Selection

Our proposed mechanism for adaptive growth bares some resemblance to the process of natural selection. In both cases a system is subject to small random variations which are more likely to persist if they increase a certain target property of the system (its fitness to its environment in the case of natural selection, or the rate of resource input from the environment in the case of an an adaptive growth process). In both cases it is the behaviour of the system as a whole, rather than its individual components, that determines which variations are selected.

The primary difference is that in natural selection the selection takes place between a number of similar systems, whereas in an adaptive growth process there is only one system and the selection occurs over time. Or, if one prefers to think of a system like Pask's ear as being a population of threads then the selection takes place according to a population-level property rather than to individual fitness; but again there is only one population.

Both processes could be simultaneously relevant in living systems. Adaptive growth processes within individual organisms could be honed by natural selection operating on a longer time scale, for instance.

Implications for Biological Development

All the structures that occur within living organisms are also maintained by a flow of energy (ultimately provided by the organism's food) and will decay if that energy flow ceases. Perhaps some of the complexity of biological structures is formed and maintained by what we have termed adaptive growth processes. If this is the case then research into these principles could vastly increase our understanding of biological growth and development processes. Pask saw this potential, writing in Pask (1960) that "the natural history of this network [of metallic threads] presents an over-all appearance akin to that of a developing embryo or that of certain ecological systems."

It seems quite possible to us that nature would take advantage of this "design for free" whenever possible. For instance, an organism's genes would not need to specify every component of an ear, but simply to arrange for circumstances in which an ear will emerge in response to stimuli from the environment and a modulated supply of energy. Of course there will also be a strong element of genetic design in nature, but an element of adaptive growth could perhaps explain why organs can atrophy or fail to develop properly when not used. Conceivably it could also be a factor in the enormous plasticity of the brain.

One could also imagine adaptive growth processes occurring at a larger scale in the development of an ecosystem. The idea that ecosystems act so as to increase or maximise the flow of resources through them (or some related quantity) is an old one in ecology, dating back to Lotka (1922) but has fallen out of favour in recent decades due to the lack of a convincing explanation. Perhaps the study of adaptive growth processes could help to illuminate this issue.

An Adaptive Growth Process in a Model

Rather than try to simulate the complex physics of electrochemical deposition we have developed a very simple computational model in which we have tried to capture the conditions required for adaptive growth to take place: there is a balance between creative and destructive processes and structures compete for a resource whose supply is globally changed in response to the system's performance at a task.

In Pask's ear metallic threads are deposited by a flow of electrons between electrodes. We have abstracted this to something which resembles a system of pheromone trails laid down by ant-like entities which move across a grid. We shall use the metaphor of ants and pheromones rather than electrons and filaments to describe our model in order to emphasise that it is not meant to be a physical simulation of Pask's electrochemical experiment. However, we consider the pheromone trails and the rate of arrival of ants to be analogous to the metallic threads in Pask's ear and to the electric current respectively.

It is important to note that, unlike most ant-based optimisation algorithms we do not selectively reinforce the trail left by a successful ant. Instead the 'reward' is applied to the system as a whole in the form of an increase in the rate at which ants arrive.

Specification of the Model

The system is divided into a grid 50 cells wide by 500 high. Each cell contains a floating point number representing the amount of pheromone present. These are initialised to zero.

The following two processes are then repeated a large number of times: first an ant enters at the top of the grid and moves towards the bottom, tending to follow any pheromone trails that are present according to the rules given below, and laying down a pheromone trail of its own. This is followed by a period of time during which all pheromone trails decay. This period of time, and hence the amount of pheromone decay that takes place, is adjusted according to a simple reward function. Note that the two time scales are separated: the ants are assumed to travel instantaneously as far as the pheromone decay time scale is concerned.

Each ant enters the system at a uniformly random position on the top row, and moves toward the bottom one row at a time according to the following algorithm:

1. Look at the amount of pheromone in the cell directly below and the cells to either side of it, three cells in total (the edges wrap around).
2. Add the value $\delta = 0.1$ to each of these three numbers (this is to prevent very weak trails from having a strong effect) and normalise them to get a probability distribution.
3. Move to one of the three cells below according to the computed probability distribution
4. Add the value 1.0 to the amount of pheromone in the previously occupied cell

After an ant has reached the bottom there is a period of simulated time in which the pheromone in each cell decays exponentially. This amounts to multiplying each cell in the grid by the same number between zero and one. The length of this time, and thus the amount of decay, depends on the reward function, which is described below. In this way the rate of build up of trails can be controlled by modulating the time that elapses between each ant entering the system.

In addition to this, before each ant enters the system the value $d = 0.01$ is subtracted from the amount of pheromone in each cell (we set the value to zero if it becomes negative). This has a proportionally greater effect on weaker trails, and means that they effectively decay more rapidly when the rate at which ants enter the system is high.

The Reward Function

The task that we set our system is substantially simpler than Pask's. We want the ants to arrive at the bottom within a specific range of columns, 19 to 31 inclusive (twelve columns out of the 50 in the system). When an ant hits the target we increase the rate at which ants enter the system. Since each ant lays down the same amount of pheromone (500 units in total) the rate at which ants arrive is proportional to the amount of pheromone added to the system, and therefore limits the total strength of trails that can be maintained in the system.

The details of the scheme used for the presented results are as follows:

1. Let the score for iteration i be $S_i = 1$ if the ant arrives at the bottom of the grid within the target interval, 0 if it misses.
2. This value is smoothed out in time slightly using a leaky integrator: Let the reward value $R_i = R_{i-1} + (S_i - R_{i-1})/\lambda$. We give the parameter λ the value 2.0 and let $R_0 = 0$.
3. Ants are assumed to arrive at a rate $99R + 1$. This is represented by multiplying each pheromone value by $1 - 1/(495R_i + 5)$ (an approximation to $e^{-5(99R_i-1)}$) to represent a constant decay of pheromone during the variable time period between ants arriving.

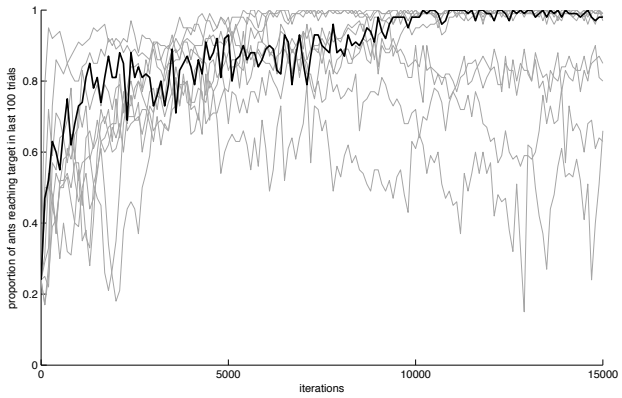


Figure 1: Increase in accuracy over time for ten independent runs of the model. The black line corresponds to the run shown in more detail in figure 2. Each data point represents the proportion of ants which hit the target over a period in which 100 ants are released. The expected probability for an ant to hit the target in the absence of any pheromone trails is 0.24, so the first data point is set to this value for each run.

Experimental Results

Figure 1 shows the proportion of ants hitting the target over time for ten independent runs of the experiment. Four of these systems do not converge on a good solution before the end of the run at iteration 15000 but six of them perform well, converging to a stable state in which very few ants miss the target.

Figure 2 shows the positions of the trails over time for one such run. One can see that the system converges fairly rapidly to a state in which there is a fairly strong trail leading to the target, which has a catchment area that catches almost all the ants entering the system. However, all trails are rewarded for this, not just the ones that lead to the target. This allows several "parasite" trails to persist, which benefit from the ants reaching the target but do not contribute towards it. These are less stable than the one which reaches the target and are eventually out-competed by it.

The four systems which do not converge to a good solution have stronger, more established parasite trails. Since more established trails fluctuate more slowly it can take a very long time for these to decay. Note that although these replicates have not converged on a perfect solution, all but one of them consistently do better than the 25% that would be expected in the absence of any reinforcement effect: they have attained solutions which work but are not perfect. The possibility of getting stuck on 'local' optima is perhaps another thing that adaptive processes have in common with evolution by natural selection.

Discussion

It is important to be clear about the relationship of our model to the subjects under discussion, namely Pask's experiment

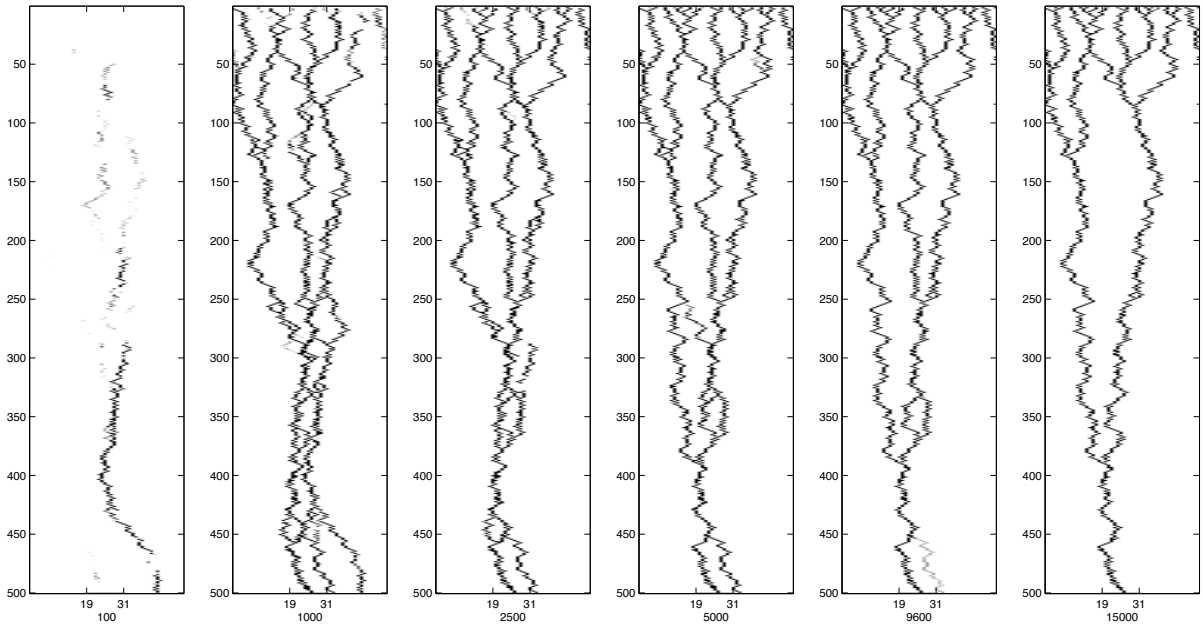


Figure 2: Snapshots of the pheromone trails after 100, 1000, 2500, 5000, 9600 and 15000 ants have passed through the system. The target is the marked interval between columns 19 and 31 at the bottom of the grid. After 100 iterations there is only a weak trail which does not lead to the target. By iteration 1000 a stable trail to the target has been formed which fans out at the top into a large catchment area, but it also supports a number of 'parasite' trails which do not hit the target. These gradually disappear, with the last fading out at around iteration 9600. After this almost all the ants hit the target (see figure 1). The system then changes very little until the end of the run at iteration 15000.

and adaptive growth processes. Our claim is that our model and Pask's device operate according to a common principle, the adaptive growth mechanism that we have described. Our model is intended as a simple instantiation of an adaptive growth process rather than as a direct model of Pask's ear, although loosely speaking the pheromone trails can be seen as a metaphor for the metallic threads in Pask's ear, with the rate of input of ants taking the place of electric current.

Comparison to Ant-Colony Methods

Our model is not intended to compete with ant colony optimisation methods as it does not have the same purpose, but since it bares some similarity to them it is worth discussing the how our system differs and the reasons for taking our approach.

It might appear that our system has several disadvantages compared to a more traditional ant-based system in which a successful trail is rewarded. We can see from figure 1 that convergence to a solution is slow and not guaranteed to occur. However, the purpose of our model is explanatory. The threads in Pask's experiments were not selectively reinforced. Instead he rewarded the whole system with an increase in current flow, and our claim is that our model captures the way in which this directed the system's growth towards a solution. Moreover, the solution found by Pask's electrochemical device does not consist of a single thread; it is a complex solution which requires the co-operation of multiple threads performing a variety of tasks. Rewarding "successful" threads in this context would not make sense, since it is not in general possible to determine which threads are contributing to the solution and which detract from it (see the discussion of the credit assignment problem above).

In our system the task is not to create a single path leading from the top of the grid to the target area, but to create a network of threads which funnels ants from all positions on the top of the grid towards the target area. There is some similarity between this and the system-level nature of the solution found by Pask's device. Our hope is that with a better understanding of adaptive growth processes it will be possible to design systems, either *in silico* like our model or in physical substrates, which can solve more complex tasks, forming solutions which equal or surpass the sophistication that Pask was able to achieve.

Implications for Adaptive Growth Processes

It seems reasonable to call the growth process in our model adaptive because it shares with Pask's ear the property that structures which contribute towards performing the task (and thus increasing resource availability) are more stable and out-compete structures which do not help perform the task.

Our computational experiment thus demonstrates that adaptive growth is a general phenomenon, rather than something which only occurs in the specific electrochemical environment of Pask's experiment.

However, adaptive growth does not occur in all possible substrates and the parameters and reward function of our model had to be chosen from the right ranges in order for the phenomenon to occur. For instance, the subtraction of d from the pheromone level in each cell for each ant that enters the system seems to be important for adaptive growth to occur. Without it stable structures that achieve the funnelling task do form, but they spontaneously collapse much more readily, reforming again soon after. Perhaps this is because parasitic structures can grow too easily, diminishing the resource supply and destabilising the whole system. It appears in this case as if the system performs a random walk between more and less stable structures, spending a high proportion of its time in a state containing a stable structure simply because those states change more slowly. But the subtraction of d on each iteration seems to provide a ratchet effect, making transitions from more to less stable structures very unlikely.

Another factor which seems important is that the growth process is fast but that the decay process is slower for more established structures. If the two processes took place on the same time scale then it would be harder for structures to persist on long time scales. The difference in time scales between new and old structures means that it's very unlikely for the whole structure to unravel by chance. If the resource availability drops it is more likely that a new structure will be dissolved than an old one, giving a trial-and-error quality to the growth process.

It also seems important that the reward function is modulated on a similar time scale to the fluctuations in the structure. This is the reasoning behind applying a leaky integrator to our reward function. A good substrate for an adaptive growth process might therefore be one which exhibits fluctuations over a wide range of time scales.

There is clearly a long way to go before we have a full understanding of why Pask's ear works, and of adaptive growth processes in general. Our example is simple and illustrative but we anticipate that more elegant and successful examples will be found which are capable of adapting to more involved tasks, leading to the possibility of practical application.

Conclusion

We have introduced the notion of an adaptive growth process in order to explain the results of Pask's electrochemical experiment. This allows us to see the enormous potential of his result: in terms of practical applications, adaptive growth processes could be used to produce control systems that can adapt to new tasks and even adjust their level of complexity when necessary. The idea may also be important for our understanding of biology since adaptive growth processes may play a role in development.

We have presented an illustrative computational model in which an adaptive growth process occurs, demonstrating that adaptive growth is a general phenomenon and paving the way for a better understanding of the circumstances under which adaptive growth can occur.

References

Lotka, A.J.: Contribution to the energetics of evolution. Proc. Nat. Acad. Sci. USA **8** (1922) 147–151

Pask, G.: Physical Analogues to the Growth of a Concept. Mechanisation of Thought Processes: Proceedings of a Symposium Held at the National Physical Laboratory. London: H.M.S.O. (1958) 879–922

Pask, G.: The Natural History of Networks. Yovits, Cameron (eds), Self-Organising Systems. New York: Pergamon Press (1960) 232–263

Pask, G.: A Proposed Evolutionary Model. Van Foerster, Zopf (eds), Principles of Self-Organisation. Pergamon Press (1961) 229–254

Cariani, P.: To Evolve an Ear: Epistemological Implications of Gordon Pask's Electrochemical Devices. Systems Research **10(3)** (1993) 19–33

Bird, J., and Di Paolo, E. A.: Gordon Pask and his maverick machines. P. Husbands, M. Wheeler, O. Holland (eds), The Mechanization of Mind in History, Cambridge, MA: MIT Press (2008)

The Spatiality of Swarms — Quantitative Analysis of Dynamic Interaction Networks

Sebastian von Mammen and Christian Jacob

University of Calgary, Alberta, Canada
{s.vonmammen,cjacob}@uclagary.ca

Abstract

Many mathematical models, which try to capture emergent phenomena, are based on state transitions that depend on neighborhood relationships. Cellular Automata (CA) and Random Boolean Networks (RBN) are examples of such models, where connectivity patterns determine the flow of signals among interconnected units. Whereas neighborhoods in CA and RBNs remain static, the focus of our investigations are artificial swarms that act in three-dimensional space, where neighborhood relationships among the swarming agents change over time. In fact, it is through the dynamically changing neighbors that determine a swarm system's overall behavior. In this paper we explore neighborhood dynamics of swarms and ask the question how each agents' time-dependent perception of its neighbors relates to specific flocking formations. We give examples of 'neighborhood functions' for choreographed swarming behaviors, such as line and figure-eight formations. We also evolve control parameters for swarm agents such that they approximate specific neighborhood functions that trigger switching and oscillations.

Introduction

Complex systems in nature usually comprise large numbers of interacting units, as for instance immune system cells that swarm in our bodies to fight off pathogens and remove damaged cells (Litman et al. (2005)). However, it already takes great effort to create and analyze stochastic models of only a few interacting units (Oilek and Klein (1979)).

Numerical experiments have been playing an increasingly important role in the investigation of complex systems (Neelamkavil (1994)). In order to build numerical models of complex systems, it is necessary to identify those features of natural systems that are crucial for the emergence of the phenomena of interest (Dasgupta (2006)). In particular, complex patterns that appear in natural systems, form in space and unfold over time, have been reproduced in models built from large sets of computational units that change their states in accordance with their local neighborhoods. Cellular Automata (Wolfram (1984)) and Random Boolean Networks (Kauffman (1995)) are examples of such models, both of which will be outlined in the subsequent section on related work.

Like in natural swarms — such as bird flocks or fish schools —, the neighborhoods of artificial swarm individuals change constantly and depend on preceding interactions. Therefore, artificial swarms represent a model of complex phenomena that embraces dynamical neighborhood relationships. We explain this idea in detail in the third section.

In order to capture the formation of neighborhood relations in swarms, we measure the numbers of neighbors for every individual at each simulated time step, within a particular neighborhood radius. We show examples of neighborhood evolutions and discuss these through swarms that exhibit specific flocking formations.

Subsequently, we demonstrate that switching and oscillating neighborhood formations can be achieved in homogeneous swarm systems whose flight is solely regulated by a linearly scaled acceleration of the individuals. We conclude this paper with a summary and an outlook on future work.

Related Work

In *Cellular Automata* (CA) the processing units (*cells*) are organized in a lattice structure and are set to an initial state that is changed in accordance with a set of rules that consider the states of all neighboring cells. CAs were primarily developed to model phenomena of self-reproduction based on the interplay among a large number of finite state machines (von Neumann and Burks (1966)). Not only was this goal finally achieved (Gardner (1970)), but CA also became a general model for complex systems based on neighborhood-dependent state changes (Wolfram (2002)). According to Wolfram (1984), patterns emerging from binary cellular automata can be classified into four different categories:

1. spatially homogeneous state;
2. sequence of simple stable or periodic structures;
3. chaotic aperiodic behaviour;
4. complicated localized structures, some propagating.

The transition from one such class or phase to another is of considerable interest for various models of natural phenomena such as the spreading of infectious diseases (del Rey

et al. (2006)). We will explore similar transitions for swarming agents in 3D space, for which their neighbors change dynamically over time.

Whereas in CA cells have a fixed spatial arrangement, *Random Boolean Networks* or 'RBNs' (Kauffman (1995)) abstract from the notion of space. Here, each cell can be connected to any other one, forming an information propagating network. As in CAs, the configuration of all cells defines the global system state.

Albert et al. conducted experiments to create and analyze random RBN with the same distributions of degrees of connectivity as those shown in gene regulatory networks (Albert (2004)), and other scale-free networks (Hidalgo and Barabasi (2006)). It has been demonstrated that RBN models show state transition patterns very similar to those of natural networks (Serra et al. (2003)).

Swarms as a Model of Complexity

Once determined, the neighborhood relations between the units of a CA or RBN remain fixed. One may assume, however, that in many natural phenomena different forces draw and push the involved units so that they change their positions, as is observed in bird flocks, fish schools, ant colonies, cell development. Thereby, of course, the neighborhoods of the units do not remain static.

Exactly this idea is captured in the 'swarm metaphor'. Large sets of swarm individuals, or agents, attract and repel each other. The neighborly influence felt by one individual determines its action for the next time step. A swarm agent changes its velocity and position, thereby gaining a new neighborhood perspective and, at the same time, altering its neighbors' perspectives. Consequently, a feedback loop of actions and reactions emerges. Unlike in CA and RBN, state changes directly impact neighborhood bonds. We argue that this feedback loop between agents and their changing neighbor arrangements is the key feature to model spatially organized systems, since locality plays a crucial role for any effective interaction.

Reynolds (1987) presented a computational model to simulate flocking formations as seen in birds or herds of animals. Here, a set of agents, called *boids*, perceive and react to each other in three-dimensional space. A formal description of Reynolds' flocking model was provided by Kwong and Jacob (2003) who interactively evolved different flight formations of boids. In particular, a boid perceives its neighbors within a viewing cone of length l and angle α (Figure 1). A boid reacts to its n neighbors by an alignment urge $\vec{v}_a$, by attraction towards and repulsion from the geometric center of the neighbors $\vec{v}_c$, and, if they get too close, by a separation urge $\vec{v}_s$. Fluctuations are introduced into the flight pattern by adding a weighted random unit-vector $\vec{v}_r$ to the acceleration. A boid's velocity and acceleration are limited by two values: v_{max} and a_{max}, respectively. Additionally, a world center $\vec{w}$ is provided that determines the swarm's

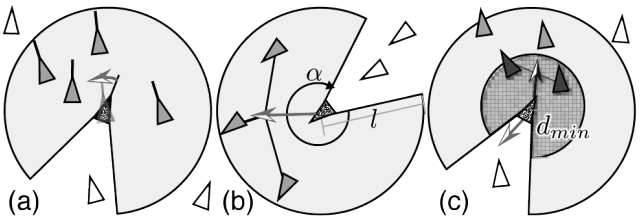


Figure 1: The three basic flocking urges alignment (a), cohesion (b) and separation (c) are depicted as they would influence (grey arrows with b/w head) the central agents (pixelized). White agents are out of scope, grey ones are within the neighborhood vicinity and dark grey ones are close enough to trigger separation. The diagrams are adapted from Craig Reynolds' website *http://www.red3d.com/cwr/boids/*.

flight destination. In order to compute boid i's acceleration $\vec{a}_i$ and resulting velocity $\vec{v}_i$ and position $\vec{p}_i$ one has to determine its set of neighbors, N_i. All those agents in $S_i \subseteq N_i$ whose distance to i is smaller than d_{min} assume a special role by contributing to i's separation urge. Once its set of neighbors is determined, the acceleration vector of a boid results from the following weighted sum of urges.

$$\vec{a}_i = c_a\vec{v}_a + c_c\vec{v}_c + c_s\vec{v}_s + c_w\vec{v}_w + c_r\vec{v}_r \quad (1)$$

$$\vec{v}_a = \frac{1}{|N_i|}\sum_{j\in N_i}\vec{v}_j \quad (2)$$

$$\vec{v}_c = \frac{1}{|N_i|}\sum_{j\in N_i}\vec{p}_j \quad (3)$$

$$\vec{v}_s = \frac{1}{|S_i|}\sum_{j\in S_i}\vec{p}_j \quad (4)$$

$$\vec{v}_w = \vec{w} - \vec{p}_i \quad (5)$$

Neighborhood relations depend on the sight, the orientation and the position of the seeing individual, on the position of the potentially perceived individual, as well as on time.[1]

We want to point out that the emerging causal chain of boid interactions can be expressed as follows (Figure 2): The actions of a swarm agent i change its state which influences all those agents that are seeing i. At the same time i's new state results in the perception of a certain set of neighbors. These neighbors influence i's actions and the feedback loop starts all over again.

It is worthwhile noting that the system state and the neighborhood configuration are inseparable in the outlined swarm model. As a consequence, the observation of alterations of neighborhoods can be utilized to describe the system dynamics. Therefore, we measure the numbers of perceived neighbors $n(t) = |N_i(t)|$ of each swarm agent i at any given

[1] All vectors $\vec{a}_i$, $\vec{v}_a$, $\vec{v}_c$, $\vec{v}_s$, $\vec{v}_w$ and sets S_i and N_i are time-dependent, but we will not denote the time variable explicitly.

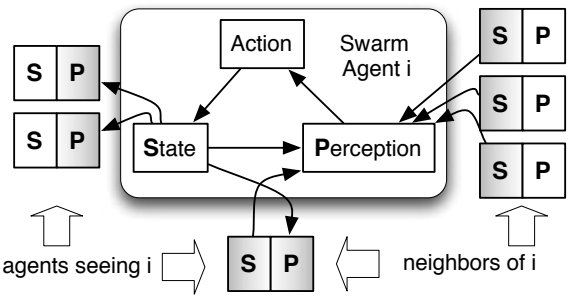


Figure 2: The slim arrows in the upper box show the direction of influence between perception, action and state of a swarm agent i. The S-P tuples stand for the state and perception modules of other agents that interact with i.

point in time t. We characterize a single state of the whole swarm by the average neighbor value of all M swarm individuals. That is, we define the time-dependent neighborhood function for a swarm with M agents as

$$\bar{n}(t) = \frac{1}{M} \sum_{i=0}^{N} n_i(t). \quad (6)$$

Finally, we suggest the evolution of $\bar{n}(t)$ over the course of time to analyze and describe the dynamics of the (swarm) system. Based on this approach we investigate various flocking formations of boid swarms in the next section.

Analysis of Flock Formations

Kwong and Jacob (2003) have shown that diverse flocking behaviors of boids can be evolved with different parameter sets for Equations 1 to 5. We utilize four sets of flocking parameters from this work (Table 1) to analyze 'choreographic' line formations and figure-eight formations based on $\bar{n}(t)$. Two different swarm configurations are provided for each formation type. The following analysis links several phases of swarm interactions and the occurrences of desired formations to the development of the neighborhood function $\bar{n}(t)$. The presented results are all produced by 50 swarm agents with a perception radius $l = 3.5$ and viewing angle $\alpha = 2.0$. As above, we normalize all $\bar{n}(t)$ values by the number of active swarm agents.

Line Formations

Figure 3 shows the development of $\bar{n}(t)$ with the line formation parameters in row (i) of Table 1. The graph shows the average number of neighbors perceived by each agent over time. The plot can be partitioned into five distinct phases. In phase I, the average neighborhood perception $\bar{n}$ is rising rapidly. Mainly the urge towards the world center $\vec{w} = (0, 0, 0)^T$ accelerates the initially stationary agents towards each other (Figure 4 (a) to (c)), ending up much closer than before (Figure 4 (d)). In fact, this phase was already described by Reynolds (1987).

Line formations

	c_a	c_c	c_s	c_w	c_r	a_{max}	v_{max}	d_{min}
(i)	7	8	5	5	5	38	13	0.14
(ii)	7	8	4	10	4	40	9	0.01

Figure-eight formations

	c_a	c_c	c_s	c_w	c_r	a_{max}	v_{max}	d_{min}
(i)	3	10	1	5	2	38	6	0.01
(ii)	5	10	2	12	1	35	6	0.34

Table 1: Evolved parameter sets for 'choreographically' flocking swarms (Kwong and Jacob (2003)).

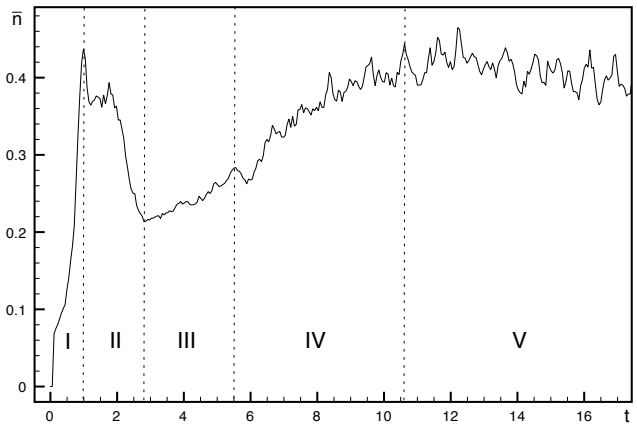


Figure 3: Developments of the average neighborhood perception $\bar{n}(t)$ matches several phases of agent interactions and flock formations.

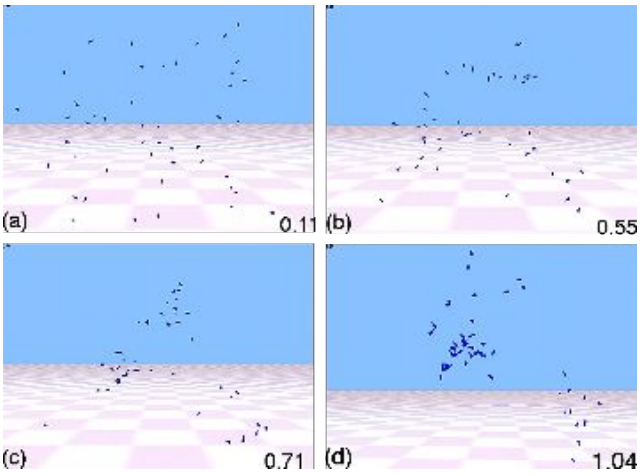


Figure 4: In phase I initially stationary agents are drawn together by the urge towards the world center.

During cluster formation the agents gain momentum bypassing many other agents. This leads to the decreasing average neighborhood perception in Phase II. As a result, several smaller flocks emerge after these two initial phases (Figure 5). The cohesion urge is now strong enough to keep subgroups of agents together that gather in the same vicinities (Figure 5 (a)). The alignment urge transforms these subgroups into flocks that exhibit increasingly homogeneous flight patterns (Figure 5 (b)).

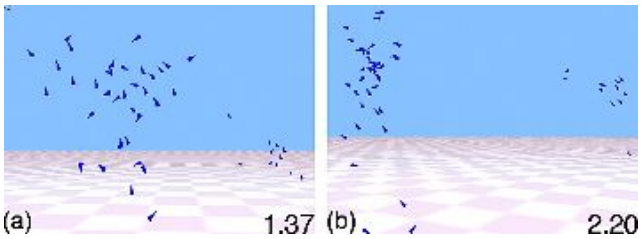


Figure 5: In phase II subgroups align as separate flock formations.

In phase III of Figure 3, a line formation emerges (Figure 6 (a)), yielding relatively small values of $\bar{n}$. Steadily, the agents are drawn closer to each other and $\bar{n}(t)$ increases accordingly. In phase IV a dense agglomeration of agents emerges at the head of the line formation (Figure 6 (b)). Eventually, the line formation is destroyed and substituted by a tight cluster formation (Figure 6(c)). After reaching phase V, the flock remains in a quasi steady state that is subject to only minor fluctuations (Figure 6(d)).

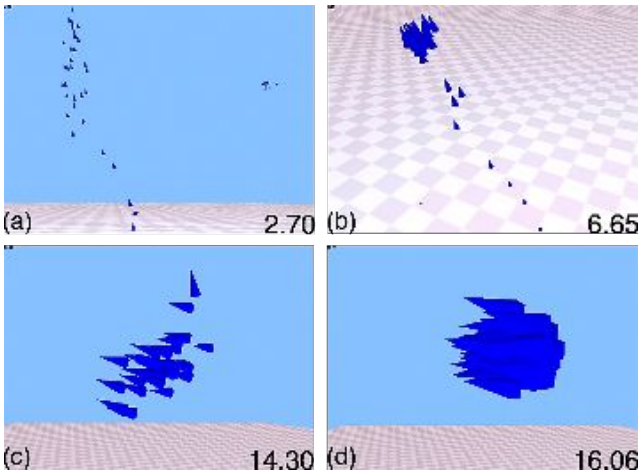


Figure 6: (a) Phase III: agents of single flocks follow each other in a line formation. (b) Phase IV: agents gather into dense clusters at the heads of the line formations. (c) and (d) Phase V: a tight cluster has formed that is robust enough against sporadic attempts of separation.

Changing the simulation to the line formation (ii) parameters in Table 1 results in increased randomness of the agents' acceleration. The swarm looses its tight, cohesive constraints and thereby allows for the sporadic escape of agents. Figure 7 shows the corresponding neighborhood function. The neighbors of a fleeing agent may try to catch up and break out of the cluster as well. Consequently, the swarm's flight is dominated by tight cluster formations but is frequently interrupted by line formations (Figure 8). Another consequence is that single agents or even whole flocks can leave the parent flock, so that eventually all agents are dispersed and unable to interact.

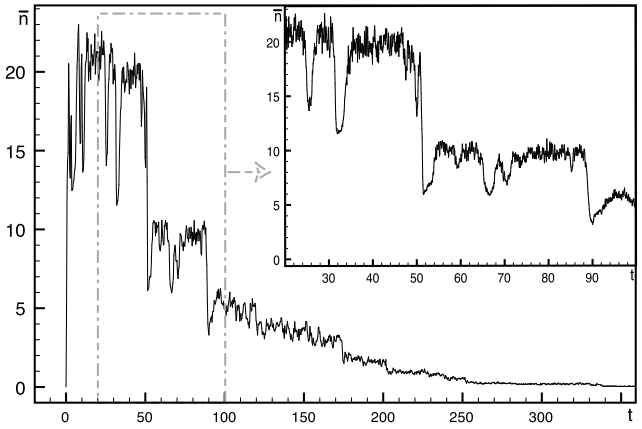


Figure 7: In the simulation of line formation (ii) of Table 1 agents break out of tightly formed clusters and take the lead of long line formations. During such events $\bar{n}$ drops temporarily (e.g. at $t = 25$ and $t = 32$). Frequently the line formations break up (as in Figure 6) and the parting flocks do not interact anymore ($t = 50$). As a consequence, $\bar{n}(t)$ reaches a value of zero at about $t = 400$.

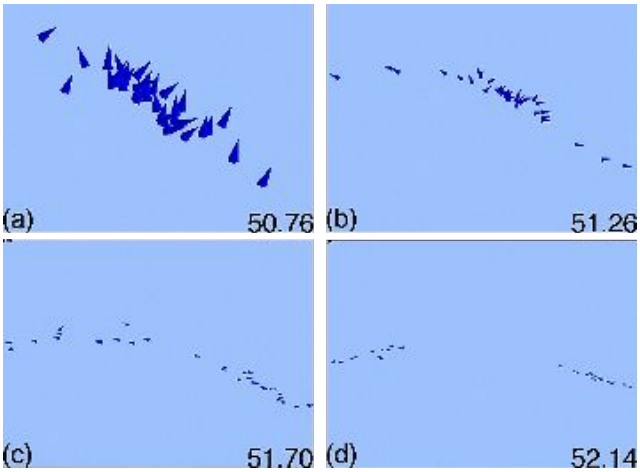


Figure 8: An agent cluster is breaking up into two line formations, one urging upwards, one towards the floor.

Figure-Eight Formations

In analogy to the discussed line formation examples, we investigate two boid configurations that exhibit figure-eight flight patterns with respect to $\bar{n}(t)$. As we can see in Figure 9, parameter configuration (i) from Table 1 reaches a steady state, whereas with setting (ii) agents repeatedly wander through different phases to eventually spread all agents far enough from each other to prevent further interaction — exactly as in line formation (ii) and in Figure 7.

Configuration (i) rapidly swings into a figure-eight formation traced by small clusters of six to ten agents (Figure 10 (b)). Here, the swarm constantly traverses through a limit cycle of global states as indicated by the fast oscillating values of $\bar{n}$ (Figure 9 (i)). In comparison to the line formation experiments, the oscillation of $\bar{n}$ is characteristic for figure-eight formations. Furthermore, for configuration (i) the oscillation reached at about $t = 20$ marks a quasi steady state.

In figure-eight configuration (ii) the neighborhood perception converges towards zero, and the subgroups of swarm agents may break away during intermediary line formations. This is similar to the second line formation experiment.

Line formations, such as illustrated in Figure 11(a) are reflected by the steep drops of $\bar{n}(t)$ in Figure 9(ii). In general, line formations fold back quickly into figure-eight formations, as is shown in Figure 11(b).

In contrast to the line experiments, the difference between a configuration that quickly results in a stable equilibrium and a swarm that exhibits long periods of drastic changes cannot be directly inferred from the according flocking parameters in Table 1. We assume that in complex figure-eight flight patterns it becomes more difficult to identify a parameter, such as the random weight c_r in Eq. 1, as crucial for spontaneous behaviors.

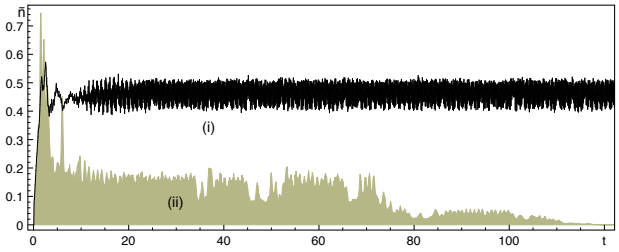


Figure 9: The development of $\bar{n}$ of figure-eight formations (i) and (ii) based on the parameter sets in Table 1.

Reverse Engineering of $\bar{n}(t)$

The examples in the previous section demonstrate how measuring the neighborhood dynamics over time can help to describe and analyze swarming behaviors. Now, we utilize this association to approximate neighborhood dynamics as

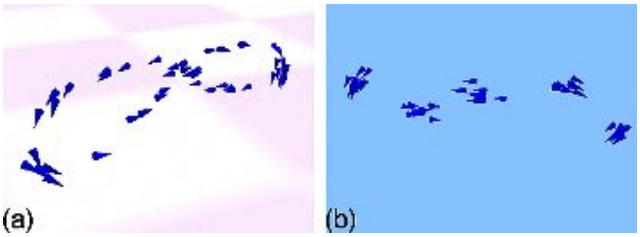


Figure 10: (a) Agents in figure-eight formation. (b) Tight agent flocks (of six to ten agents) in figure-eight formation.

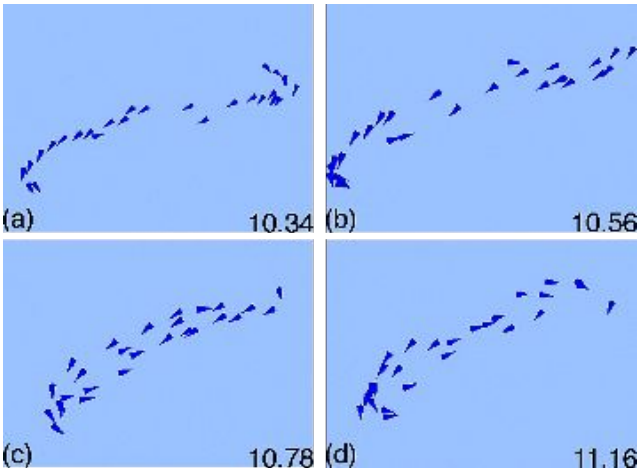


Figure 11: A line formation is about to collapse into a figure-eight pattern.

they might contribute to the coordination of naturally occurring phenomena, such as biological switches and clocks or timers.

The neighborhood value $n_i(t)$ of a single agent may change rapidly, remain fixed or oscillate. It is also easy to discover a whole flock of agents entering an equilibrium of a specific average neighbor value $\bar{n}(t)$. The results in the previous section also show that even a whole swarm can change dynamically, or, put differently, can follow specific evolutions of $\bar{n}(t)$.

If it was not for its contextual evolution, the average neighborhood $\bar{n}(t)$ would mainly characterize the concurrent spread, or density, of boid flocks, thereby corresponding much to the common view on molecular concentrations. In fact, neighborhood fluctuations indicate changes in the structure of swarms. Immediately, the question arises which patterns of movement could one expect when looking at evolutions of $\bar{n}(t)$ that correspond to the development of molecular concentrations in biological measurements.[2]

Even though the expression of genes happens stochas-

[2]Such measured molecular concentrations may come from microarray experiments that approximately capture the number of (reporter) proteins over time.

tically, the levels of expression can differ greatly which promotes the idea of a genetic switch (Jacob and Burleigh (2004)). By the approximation of a step function for $\bar{n}(t)$, we intend to show that even a homogeneous swarm could exhibit bi-stable switching behavior. Oscillations occur in natural systems as timers, such as circadian clocks. A a second option, we therefore explore which swarm behaviors can be evolved that follow a sinusoidal neighborhood function. For both endeavors we utilize a genetic algorithm that operates on populations of swarms as described in the following paragraphs.

Evolutionary Experiments

A homogeneous boid swarm, consisting of agents that share the same control parameters, are represented by a genotype vector $\vec{b} = (c_a, c_c, c_s, c_w, c_r, v_{max}, a_{max}, l, r)^T$. We also want to modify the starting positions, initial accelerations and initial velocities of the swarm agents: $init_0, init_1, ..., init_N$. The extended swarm genotype is therefore $\bar{g} = (\vec{b}, init_0, init_1, ..., init_N)$.

In the following experiments we provide a desired target function $x(t)$ for $\bar{n}$ and reward its approximation with a fitness value $f = 1/\left(\sum_{t=1}^{40} |\bar{n}(t) - x(t)|\right)$. We rely on the genetic operators of fitness proportionate selection, incremental mutation and multi-point crossover on all numeric values. A population counts 30 swarms, with each swarm consisting of 30 agents. The genetic algorithm was run up to 300 generations.

Step Function

As the computation of the genotype was limited to 40 simulated seconds, we decided to trigger the switch at about half of the overall time-frame. A difference of 0.5 units in a system with values $\bar{n} \in [0; 1)$ denotes an obvious leap, whereas $\bar{n} = 0.25$ is large enough to allow for further swarm interactions (as opposed to $\bar{n} = 0.0$ that rules out the possibility for local interactions).

$$x(t) = \begin{cases} 0.25 & t < 22 \\ 0.75 & t \geq 22 \end{cases}$$

Figure 12 displays the step function approximation of a boid configuration that appears after 200 generations of the outlined genetic algorithm. In Table 2 we list the parameters of the best evolved swarm genotype, which reveals two surprising values: $c_s = 1.0$ and $v_{max} = 0.0$. In fact, the velocity of the swarm individuals is greater than zero — the integration step of the simulation increases the velocity in accordance with the provided acceleration $\vec{a}$ that is limited to $a_{max} = 12.15$. The limitation of $v_{max} = 0.0$ means that the agent is stopped after each iteration, resulting in a very small velocity value, yet ensuring the orientation and alignment according to its flocking urges. As a consequence, starting from their initial positions, the agents are slowly converging towards each other — nicely timed with the target function. The large weight for the separation urge $c_s = 1.0$ prevents the agents from getting too close and exceeding the target value of $x = 0.75$. The swarm is trained to approximate the step function for 40 simulated seconds. For that reason, the flight parameters are delicately balanced within this time frame. In the given example the swarm reaches an equilibrium shortly afterwards.

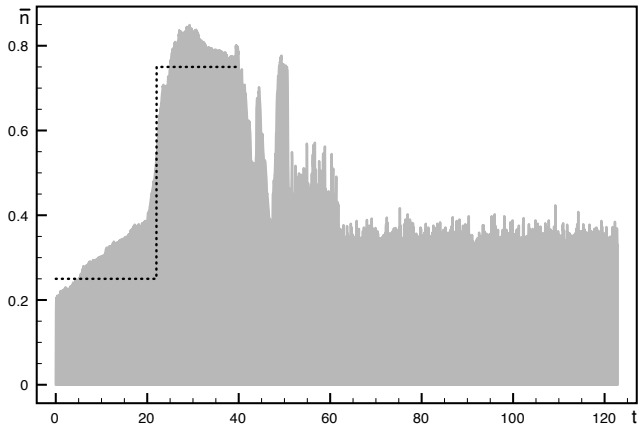


Figure 12: A neighborhood function $\bar{n}(t)$ of a boid configuration, bred by an evolutionary algorithm, approximates a step function as $\bar{x}(t)$. Afterwards, outside the evaluation window the swarm drops into an equilibrium with $\bar{n} \in [0.35; 0.45]$.

Boid configuration for $\bar{n}$ step function approximation

c_a	c_c	c_s	c_w	c_r	a_{max}	v_{max}	d_{min}
0.37	0.86	1.0	0.44	0.48	12.15	0.0	4.89

Table 2: Evolved swarm parameters that result in the neighborhood function of Figure 12, implementing a switch in $\bar{n}(t)$ ($\alpha = 2.09, l = 9.32$).

Sine Function

Two periods of a sine function are provided as target function $x(t)$ for time frame of 40 simulated seconds. As in the step function approximation, $\bar{n}$ is not forced to drop below 0.25 to guarantee a minimal space for interactions.

$$x(t) = sin(4\pi * t/40.0) * 0.25 + 0.5$$

Figure 13 shows the neighborhood function $\bar{n}(t)$ for the evolved swarm configuration as listed in Table 3. Eventually, at $t = 1244$ in Figure 14, the oscillation ends; this is when the agents form a tight cluster orbiting around the world center $\vec{w}$.

Since complex interactions render it difficult to identify certain flocking patterns, we activated motion blurring to

better capture the pattern formations of the swarm. We determined that the oscillation happens as the biggest flock repeatedly expands (Figure 15) and contracts (Figure 16). Leaps from a plateau to a local maximum, as seen at $t = 100$ in Figure 13, occur when formerly separated flocks rejoin (Figure 17). In Figure 18 several screenshots with activated motion blur illustrate intermediary flight formations.

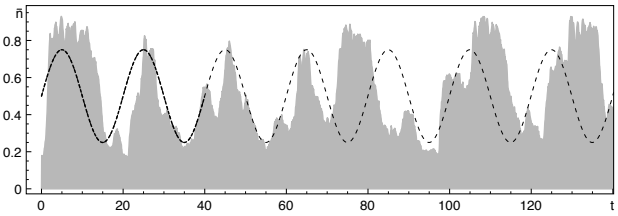


Figure 13: A boid configuration bred by an evolutionary algorithm approximates two periods of a sinusoidal neighborhood function. As shown, the oscillations sustain even afterwards.

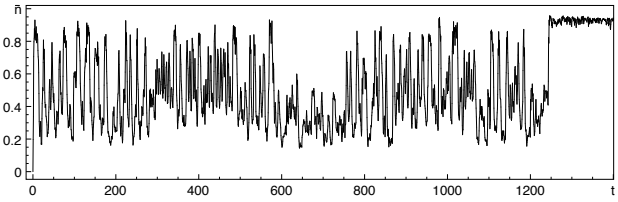


Figure 14: After about 1200 simulated seconds the oscillating swarm transitions into a steady state.

Boid configuration for $\bar{n}$ sine approximation

c_a	c_c	c_s	c_w	c_r	a_{max}	v_{max}	d_{min}
0.76	0.95	0.53	0.36	0.76	12.15	7.16	4.12

Table 3: Evolved swarm parameters that result in the neighborhood function $\bar{n}(t)$ of Figure 13. The corresponding swarms display oscillating behaviors ($\alpha = 2.64$, $l = 7.86$).

Summary and Future Work

Agent states and neighborhood relations are inseparable in swarm systems. Therefore, the dynamics of a swarm can be measured as the fluctuations in perceived neighbors. Based on this approach we are able to characterize the dynamics of boid swarms that exhibit various flocking formations. Hereby, we also identify phases and phase transitions of the boid system, including limit cycles and stead states.

Vice-versa, we evolve boid configurations to approximate characteristic neighborhood functions. Here we show that

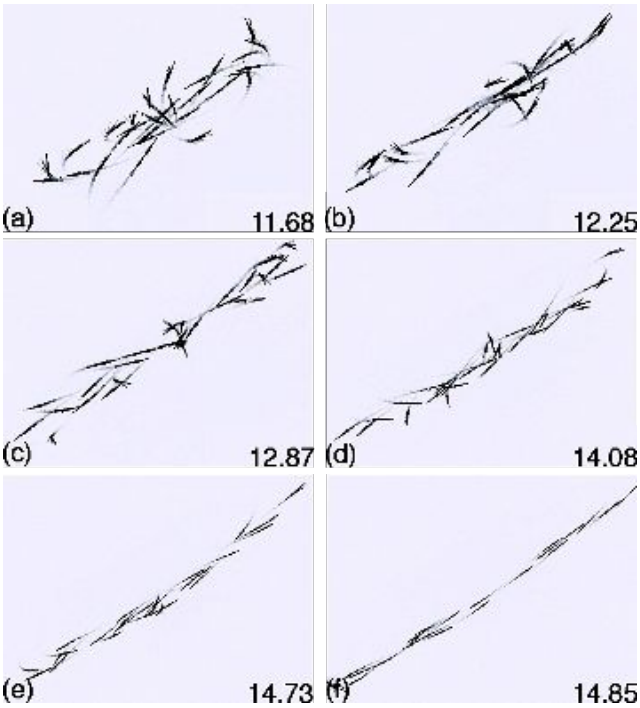


Figure 15: The flock extends in two directions.

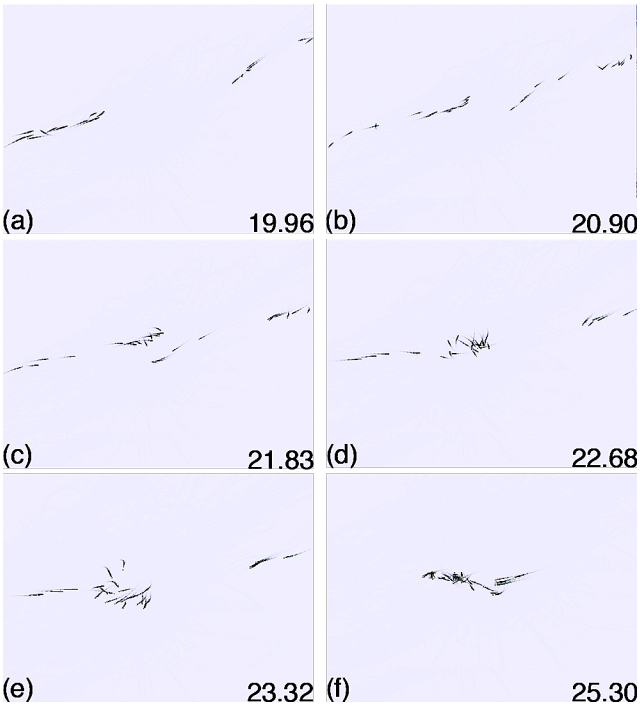


Figure 16: The previously extended flock from Figure 15 contracts again.

non-linear and oscillating neighborhood developments can emerge in spatially organized homogeneous swarms that

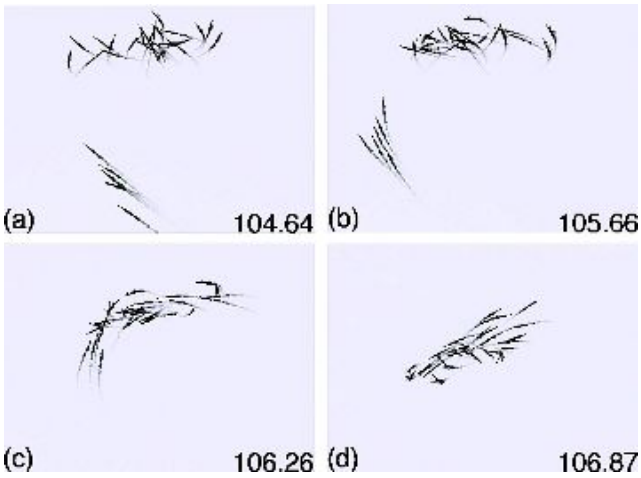


Figure 17: A second flock approaches and joins the other one (continued from Figure 16).

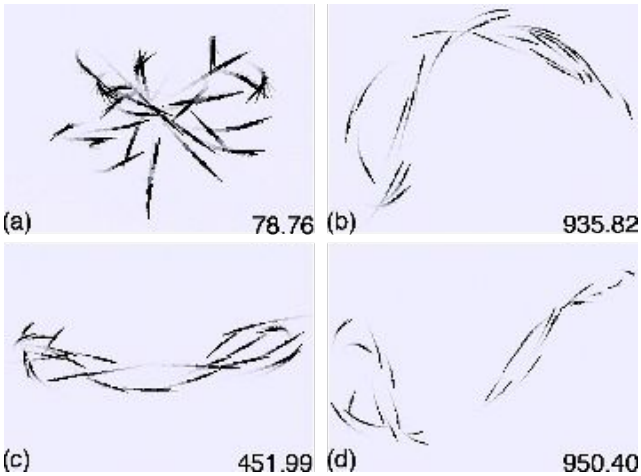


Figure 18: Motion blurring renders some of the more complex flight patterns identifiable: (a) Spherical formation, (b) a U-bent figure-eight, (c) and (d) extended figure-eights.

solely rely on linearly scaled flight acceleration based on repulsion and attraction.

As in studies of complex pattern formations in two dimensional CA (Wolfram (1984)), experimental data suggests that (1) Varying initial conditions and noise influence the evolution of boid flocking patterns locally. The general characteristics of the emerging patterns, however, are mainly based on the flocking parameters of the swarm. (2) Different swarm configurations can lead to very different pattern formations. (3) Chaotic behavior — unexpected, chance-based phase transitions — can occur in systems that initially show orderly, periodic patterns.

For further investigation, we suggest the creation of an abstract swarm model. It has to maintain the link between state and neighborhood, but should be reducible to any high-dimensional space. It would be desirable to find operators that comply with the amalgamation of time and space, or respectively structure and state, without realization of physics. With a generalized set of operators that change states and neighborhood relations concurrently, a systematic classification of swarm dynamics might be possible. Thereby, swarms could become an important model for the dynamics of complex systems in general.

References

Albert, R. (2004). Boolean modeling of genetic regulatory networks. *Complex Networks*, pages 459–481.

Dasgupta, D. (Nov. 2006). Advances in artificial immune systems. *Computational Intelligence Magazine, IEEE*, 1(4):40–49.

del Rey, A., White, S., and Sánchez, G. (2006). A model based on cellular automata to simulate epidemic diseases. *Cellular Automata*, pages 304–310.

Gardner, M. (1970). Mathematical games: The fantastic combinations of john conway's new solitaire game "life". *Scientific American*, 223:120–123.

Hidalgo, C. A. and Barabasi, A.-L. (2006). Scale-free networks. *Scholarpedia: The free peer reviewed encyclopedia*.

Jacob, C. and Burleigh, I. (2004). Biomolecular swarms - an agent-based model of the lactose operon. *Natural Computing: an international journal*, 3(4):361–376.

Kauffman, S. (1995). *At Home in the Universe: The Search for the Laws of Self-Organization and Complexity*. Oxford University Press.

Kwong, H. and Jacob, C. (2003). Evolutionary exploration of dynamic swarm behaviour. In *Congress on Evolutionary Computation*, Canberra, Australia. IEEE Press.

Litman, G. W., Cannon, J. P., and Dishaw, L. J. (2005). Reconstructing immune phylogeny: new perspectives. *Nat Rev Immunol*, 5(11):866–879.

Neelamkavil, F. (1994). *Computer Simulation and Modelling*. John Wiley and Sons.

Oilek, M. and Klein, P. (1979). Stochastic model of the immune response. *Modelling and Optimization of Complex System*, pages 15–25.

Reynolds, C. W. (1987). Flocks, herds, and schools: A distributed behavioral model. In *SIGGRAPH '87 Conference Proceedings*, volume 4, pages 25–34.

Serra, R., Villani, M., and Agostini, L. (2003). On the dynamics of scale-free boolean networks. *Neural Nets*, pages 43–49.

von Neumann, J. and Burks, A. W. (1966). *Theory of self-reproducing automata*. University of Illinois Press, Urbana and London.

Wolfram, S. (1984). Cellular automata as models of complexity. *Nature*, 311:419–424.

Wolfram, S. (2002). *A new kind of science*. Wolfram Media Inc., Champaign, Ilinois, US, United States.

Optimal Robot Recharging Strategies For Time Discounted Labour

Jens Wawerla and Richard T. Vaughan
Autonomy Lab, Simon Fraser University
Burnaby, BC V5A 1S6, Canada
{jwawerla, vaughan}@sfu.ca

Abstract

Energy is defined as the potential to perform work: every system that does some work must possess the required energy in advance. An interesting class of systems, including animals and recharging robots, has to actively choose when to obtain energy and when to dissipate energy in work. If working and collecting energy are mutually exclusive, as is common in many animal and robot scenarios, the system faces an essential two-phase action selection problem: (i) how much energy should be accumulated before starting work; (ii) at what remaining energy level should the agent switch back to feeding/recharging? This paper presents an abstract general model of a energy-managing agent that does time-discounted work. Analyzing the model, we find solutions to both questions that optimise's the value of the work done. This result is validated empirically by simulated robot experiments that agree closely with the model.

Introduction

Since the early years of robotics, e.g. Walter (1963), roboticists have been challenged by the need to supply robots with energy. For mobile robots that carry their own limited energy store, the key question is "when should the robot refuel/recharge?". A standard approach in the literature and in commercial robots is to set a fixed threshold and refuel whenever the robot's energy supply drops below this threshold (Silverman et al. (2002), Silverman et al. (2003)). The simplicity of this approach is appealing, but it may not be the optimal strategy. For example Wawerla and Vaughan (2007) showed that, in a realistic surveying task, an adaptive threshold produces a higher overall work rate than a static threshold, and a rate maximizing approach outperforms both by a large margin. The biologically-inspired rate maximizing method performs well, but it has some important limitations discussed below. This paper provides a more sophisticated model for rational recharging robots.

Robots, by their name[1] and nature, are supposed to perform some form of labour, e.g. space exploration, entertainment, rescue missions, clean-up, assembly, etc. Most tasks require execution in a timely manner, though some more then others. For example, we may be willing to wait for a day or two for the latest geological observations from Mars, while waiting the same time for the rescue of trapped miners or ordnance disposal might not be acceptable. The standard method to model the decreasing value of work over time, and thereby encourage timely execution, is to discount by some factor β in discrete timesteps the reward given in exchange for investment (Varian (1992)), where investment here is energy dissipated in labour. The inverse of discount $(1/\beta)$ is the familiar interest rate, of savings accounts and credit cards.

The laws of physics dictate that energy cannot be transferred instantaneously, or in other words, refuelling takes time and this time cannot be spent working. If a robot spends an hour refuelling, it starts to work one hour later and since the reward is discounted over time, it receives a smaller payment then if it would have started working immediately. But the initial charging period is strictly required, as no work can be done without previously obtaining energy. This conflict between the mutually exclusive tasks of refuelling and working raises two interesting questions:

Q1 How much energy should be accumulated before starting work?

Q2 At what remaining energy level should the agent switch back to obtaining energy?

Most real-world robot systems avoid these questions by maintaining a permanent connection to an energy source, e.g. industrial robotic manipulators wired into the mains power grid, or solar powered robots which are capable of gathering energy while performing some task at the same time. This paper addresses the more interesting class of machine, including animals and mobile robots, that must obtain and store energy prior to working. The Q1,Q2 action selection problem must be solved by every animal and long-lived robot in some way or another. Further we are only considering rational agents. Any introductory textbook on decision making (e.g. Stuart and Peter (2003)) defines an agent to be rational if it always selects that action, i.e. an answer to Q1 and Q2, that returns the highest expected utility. Here we

[1] Webster: Czech, from *robota*: compulsory labour

assume that utility is proportional to the reward obtained by working, which is discounted over time.

After considering related work, we analyze the problem in terms of a simplified abstract model which, when parametrized to approximate a particular robot system, predicts the optimal answers to Q1 and Q2. We validate the model by comparing its predictions with data empirically obtained from a simulated robot.

To the best of our knowledge, this is the first proposed solution to this general robotics problem.

Related Work

The literature on robotic energy management has many aspects ranging from docking mechanisms, energy-efficient path-planning, to fuel types. The most relevant aspect to this work is on action-selection. Perhaps the most standard and simple way to determine when it is time for the robot to recharge is to set a fixed threshold. This can either be a threshold directly on the energy supply as in Silverman et al. (2002) or on time elapsed since last charging as in Austin et al. (2001). The latter is usually easier to implement but less accurate, because one has to have some model of the energy supply. However, Wawerla and Vaughan (2007) showed a fixed threshold policy can be improved upon. While it is true that maximizing the energy intake rate maximizes potential work rate, it does not optimize with respect to when the work is done. It also *assumes* that recharging is always valuable. This is often true, but not always, as we show below. Also notable is that all of the above papers refuel the robot to maximum capacity at each opportunity, and do not consider that this may not be the best policy.

Litus et al. (2007) consider the problem of energy efficient rendezvous as an action selection problem, and so investigate the where, but not the when and how long, to refuel. Birk (1996) had robots living in a closed ecosystem learn to 'survive'. Here robots learned to choose between recharging and fighting off competitors. Birk's agents' value function of 'survivability' is different to that considered here. The rational robot and its owner are interested in gaining maximum reward by working at the robot's task, and are indifferent to the lifespan of the robot. This is a key difference between the purpose of robots and animals.

Although intended as a wake-up call for psychology research, Toda's Fungus Eater thought experiment (Toda (1982)) has been influential in the robotics literature. The survival quest of a mining robot on a distant planet contributed significantly to ideas of embodiment and whole agents (Pfeifer (1996)), but the action selection problem presented has yet to be solved in more then the trivial way of a fixed threshold policy.

Spier and McFarland (1997) and McFarland and Spier (1997) investigate work - refuel cycles, or as they call it 'basic cycles', and show a simple rule, based on cue and deficit,

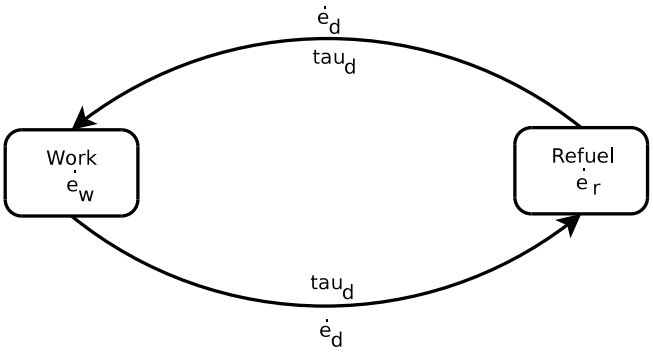


Figure 1: Refuel and time discounted labour model, see text for details

can solve a two resource problem. The cue-deficit policy is inherently reactive and thus fails to cope with the cost of switching between behaviours. Lacking any form of look ahead or planning, so it is difficult to see how it would handle discounted labour situations.

From McFarland's work, it is a small step into the vast literature on behavioural ecology, from which Houston and McNamara (1999), Stephens and Krebs (1986) and Stephens et al. (2007) are strongly recommended starting points. Due to the biological background of these publications, the analogy to labour and reward in a robotic case is not obvious. The majority of this work uses dynamic programming (DP) as a means of evaluating models of animal behaviour. An exception that does not rely on DP is Hedenström (2003), who investigates the bio-mechanics of land based animals to derive models of optimal fuel load during migration. The model optimizes for migration time, does not map directly into discounted labour or the cyclic work-charge lifestyle of the long-lived robot.

It is known that animals prefer a small, immediate reward to a large delayed reward. So animals seem to do some form of time discounting. According to Kacelnik and Bateson (1996) the reason seems to be that animals in general are risk averse. From a robotics point of view, one major issue with the descriptive models of behavioural ecology is that they lack the 'how does the animal actually do it' prescriptive description and hence do not translate readily into robot controllers. Work that tries to bridge the gap between ecology and robotics is Seth (2007). Here Seth uses ALife methods to evolve controllers that obey Herrnstein's matching law (roughly: relative rate of response matches relative rate of reward), which again is in the domain of rate maximization.

The Model

In this section we describe the behavioural model used in this work. In order to keep the analysis tractable we choose an abstract, slightly simplified model. The world is mod-

elled as two distinct spatially separated sites: a work site and a refuelling site. Moving between sites has a non-zero cost (Figure 1). The robot has a energy storage of $E(t)$, where $0 \leq E(t) \leq E_{max}$. If the energy supply drops to zero anywhere but at the refuelling site, the robot looses is ability to move or work and can gain no more reward. The robot can be in one of four states:

- **refuelling** with a refuelling rate of $\dot{e_r}$, to do so the robot has to be at the refuelling site.
- **transitioning from the refuelling site to the work site**, the duration of this transition is τ_d and the robot has to spend energy at a rate of $\dot{e_d}$, so the transitions cost in term of energy is $\tau_d \dot{e_d}$
- **working**, which gets the robot a reward of $R = \int_{t_0}^{t_0+\tau_w} \beta^t dt$, where t_0 is it time when the robot starts to work and τ_w is the duration the robot works for. Therefore the reward the robot earns by working is discounted with a discount factor $0 < \beta < 1$. While working the robot spends energy with a rate of $\dot{e_w}$. In other words, the robot turns energy into work and therefore reward. In case where the robot performs several work sessions, the reward is accumulated and only the overall reward is at interest to the owner of the robot. As with refuelling, work can only be performed at the work site.
- **transitioning from the work site to the refuelling site**, the duration of this transition is τ_d and the robot has to spend energy at a rate of $\dot{e_d}$

The robot's goal is to achieve as much reward as possible. To do so, it has to make two decisions, (1) when to stop refuelling and resume work and (2) when to stop working and refuel. We mostly refer to the action of accumulating energy as *refuelling* and not as *recharging* because we want to emphasis the general nature of our model.

It is worth pointing out that in a real world scenario all important variables, namely the energy rates, could be known in advance or are easily measured by the robot. Here we assume these variables to be constant, though in an actual implementation we would use averages as approximations. It would also be feasible to do some form of piece-wise linear approximation of the energy rates. The discount factor can also be assumed to be known, since this factor is task dependent and, hence, is set by the owner or designer of the robot or by some external market. As we show below, even if all else is fixed, the robot owner can use the discount factor as a control variable that can be tweaked to fine tune the robot's behaviour. Everything else is predefined by the tasks, the robot's construction or the environment.

In order to improve readability, we need to introduce some additional notation. $k_1 = \frac{\dot{e_r}}{\dot{e_w}}$ is the ratio of the energy rate while refuelling to the rate while working. Similarly, $k_2 = \frac{\dot{e_d}}{\dot{e_w}}$ is the energy rate while transitioning to the energy rate while working. $k_3 = \frac{\dot{e_d}}{\dot{e_r}} = \frac{k_2}{k_1}$ is the ratio of the energy rate while in transition and the energy rate refuelling. τ_r is the time spent refuelling during one refuel-work cycle. The amount of work the robot can perform is limited by the energy supply the robot has, so we express the potential work duration as a function of refuelling and transitioning time where $\tau_w = \tau_r k_1 - 2\tau_d k_2$, which is basically the amount of time the robot can work for, given the amount of energy the robot got from refuelling minus the energy the robot has to spend to travel to the work site and back to the charging station. We also introduce the period of time $T = \tau_r + 2\tau_d + \tau_w = \tau_r(1+k_1) + 2\tau_d(1-k_2)$ as the length of one refuel-work cycle.

When to stop working

Let $e(t)$ be the energy in the robot's storage at time t. At what energy level $e(t) < \epsilon_{w \to r}$ should the robot stop working and transition to the refuelling site? Since the value of work is time discounted, work that the robot performs now is always more valuable then the same amount of work performed later. This creates an inherent opportunity cost in transitioning from the work site to the refuelling site because it takes time and costs energy $\tau_d \dot{e_w}$ that cannot be spent working. This implies that the robot needs to work as long as possible now and not later. Hence the only two economically rational transitioning thresholds are:

- $\epsilon_{w \to r} = \tau_d \dot{e_d}$
  The robot stops working when it has just enough energy left to make it to the refuelling station. The robot will spend the maximum amount of energy, and therefore time, working, ensuring the highest reward before refuelling. Comparatively, should a higher transitional threshold be used, the robot would stop working earlier and refuel earlier, but discounting results in a smaller reward. Should the transitioning threshold be smaller, the robot would have insufficient energy to reach the refuelling station. In this case, the robot cannot gain any further reward because it runs out of fuel between the work and refuel sites.
- $\epsilon_{w \to r} = 0$
  The robot spends all of its energy working and terminates its functionality while doing so. At first glance this option seems counter intuitive, but one can imagine highly discounted labour situations, such as rescue missions, where the energy that would otherwise be spent on approaching a refuelling site is better spent on the task at hand. This might also be a rational option if the transition cost is very high, e.g. NASA's Viking Mars lander took a lot of energy to Mars in the form of a small nuclear reactor, rather than returning to Earth periodically for fresh batteries (the recent Mars rovers employ solar cells to recharge their batteries originally charged on Earth).

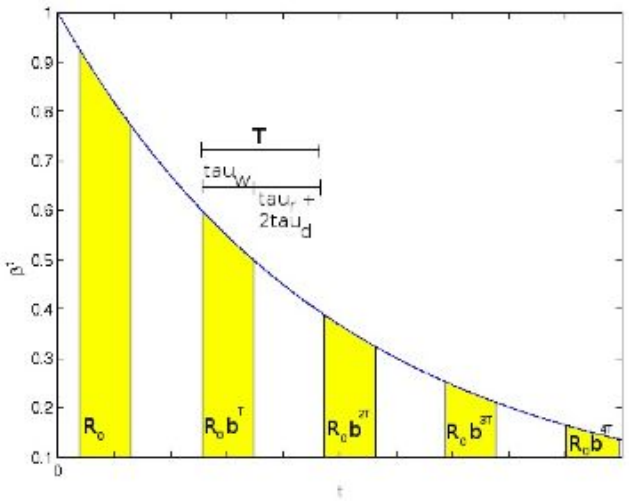


Figure 2: General discounting in refuel - work cycles. The shaded areas are periods in which the robot works and thus earns a reward. The white areas correspond with time in which the robot does not obtain any rewards because it either travels or refuels

Suicide or live forever?

Using our simple model, we can determine whether a robot in a given scenario should terminate while working or continue indefinitely with the work-refuel cycle. Let

$$R_0 = \int_{\tau_r+\tau_d}^{\tau_r+\tau_d+\tau_w} \beta^t dt = \beta^{\tau_r+\tau_d} \frac{b^{\tau_w}-1}{\ln(\beta)} \qquad (1)$$

be the reward obtained from spending $E_{max} - \epsilon_{w\to r}$ energy or τ_w time during the first working period (see figure 2). In this figure the shaded areas correspond to time in which the robot performs work and thus obtains a reward proportional to the size of the shaded area. Later work periods are discounted more strongly and hence provide a smaller reward. White areas correspond to times in which no reward is earned because the robot either travels between the work and refuelling site or it refuels. The size of this area is proportional to the opportunity cost, that is, reward that, in principal, could have been obtained if the time had been spent working.

Let T be the duration of one full work-refuel cycle, that is working - transition - refuel - transition, or $T = \tau_w + \tau_d + \tau_r + \tau_d$. Therefore, the reward gained in the next cycle is the initial reward R_0 discounted by T and becomes $\beta^\tau R_0$. Subsequent rewards are again discounted by T and so the reward for the third cycle is $\beta^{2T} R_0$. The sum of all rewards if working infinitely, that is choosing $\epsilon_{w\to r} = \tau_d k_2$, is

$$R_\infty = R_0 \sum_{i=0}^{\infty} \beta^{iT} = R_0 \frac{1}{1-\beta^T} \qquad (2)$$

In practice no system will last forever, so this analysis is slightly biased towards infinite life histories.

If the robot chooses $\epsilon_{w\to r} = 0$ it gains the initial reward R_0 plus a one time bonus of

$$R_+ = \int_{\tau_r+\tau_d+\tau_w}^{\tau_r+\tau_d+\tau_w+\tau_d k_2} \beta^t dt = \beta^{\tau_r+\tau_d+\tau_w} \frac{\beta^{\tau_d k_2}-1}{\ln(\beta)} \qquad (3)$$

by spending the energy required for transitioning on working. The reward gained over the live time of the robot (which is fairly short) is $R_{rip} = R_0 + R_+$.

So the answer to Q2 is that the rational robot selects that threshold $\epsilon_{w\to r}$ that achieves the higher overall reward, so it picks

$$\epsilon_{w\to r} = \begin{cases} 0 & : \quad R_{rip} \geq R_\infty \\ \tau_d \dot{e_d} & : \quad R_{rip} < R_\infty \end{cases} \qquad (4)$$

Since the discount function $\int \beta^t dt$ belongs to the class of memory-less functions, we only have to calculate eq. 4 once, in other words if it is the best option to refuel after the first work cycle it is always the best option to do so and vice versa.

How much energy to store

We have shown how to determine a threshold for transitioning from work to refuelling. In this section we will analyze when to stop refuelling and resume work, or phrased differently, how much energy to accumulate before starting to work. Energy and time are interchangeable elements, provided that we know the rate at which energy is spent and gained. Since discounting is done in the time domain, our analysis equates energy with time for simplicity. Based on this, we can ask the time equivalent of Q1: 'how long should the robot refuel for?' We call this refuelling duration τ_r. To be rational, the robot must refuel long enough to gain enough energy to make the trip to the work site and back, that is $2\dot{e_d}\tau_d$, otherwise it would have to turn around before reaching the work site and thus will not gain any reward. Refuelling after the storage tank is full is time wasted that would better be spent obtaining a reward. Therefore the refuelling time is limited to $2\tau_d k_3 \leq \tau_r \leq E_{max}\dot{e_r}^{-1}$. In the following we assume, without loss of generality, the robot starts at the refuelling site with an empty fuel tank. Assuming differently will just result in shift of the analysis by a constant factor, but will not change the overall conclusions.

Acyclic tasks

First we examine situations in which the robot has to refuel for a task that has to be done only once, that is the robot refuels, performs the task, and returns to the refuelling site. Depending on the time spent refuelling the robot obtains the following reward during the upcoming work period.

$$R(\tau_r) = \int_{\tau_r+\tau_d}^{\tau_r+\tau_d+\tau_w} \beta^t dt = \beta^{\tau_r+\tau_d} \int_0^{\tau_r k_1 - 2\tau_d k_2} \beta^t dt \qquad (5)$$

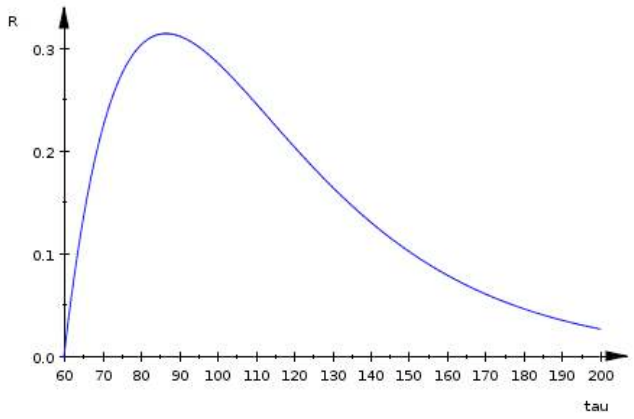


Figure 3: Reward depending on refuelling time with an example configuration $k_1 = 0.5, k_2 = 0.5, \beta = 0.97, \tau_d = 85s$

Next we need to find the $\hat{\tau_r}$ that maximizes $R(\tau_r)$, which is

$$\hat{\tau_r} = argmax(R(\tau_r)) = \frac{\log_\beta\left(\frac{1}{k_1+1}\right) + 2\tau_d k_2}{k_1} \qquad (6)$$

Figure 3 shows an example reward function (eq. 5) depending on the refuelling duration τ_r. Using eq. 6 we calculate that for this particular configuration the reward is maximized when the robot refuels for $\hat{\tau_r} = 86.6234...$. If the fuel tank is filled before that time, the best the robot can do is return to work. This will give it the highest reward achievable, but the designer should keep in mind that there might exist a class of robots with a larger fuel tank that will achieve a higher reward. Note that if the robot stops refuelling at $\hat{\tau_r}$ even if its energy store is not full to capacity, and transitions to working, it earns the highest reward possible. To our knowledge this has not been stated explicitly in the robotics literature before. It is generally assumed that robots should completely recharge at each opportunity, but this is not always the optimal strategy.

Cyclic tasks

In cyclic tasks a robot is required to always return to work after resupplying with energy. Here the analysis is slightly different then in the acyclic case because the refuelling time of the current cycle not only influences the duration and length of the work period of this cycle but of all cycles to come. Hence, we should select a refuelling threshold that maximizes the overall reward. The overall reward is calculated by (see fig. 2)

$$R_\infty(\tau_r) = R_0 \sum_{i=0}^{\infty} \beta^{iT} = \frac{(\beta^{\tau_w} - 1)\beta^{\tau_r + \tau_d}}{(1 - \beta^T)\ln(\beta)} \qquad (7)$$

Unfortunately, it seems impossible to find a closed form solution to $\hat{\tau_r} = argmax(R_\infty(\tau_r))$. However, eq. 7 can easily be evaluated for a given τ_r and so calculating the reward for each of a finite set of values for τ_r and selecting the one that maximizes the reward is quite practical. In any real application the number of τ_r to be tested is limited and possibly rather small, in the order of a few thousand. This is because any real robot will have a finite energy storage and any practical scenario will require only limited sampling due to the resolution of the fuel gauge, the uncertainty in the environment, etc. In the case of our Chatterbox Robot (see below), the battery capacity is 2.8 Ah and the fuel gauge has a resolution of 1mA, resulting in less than 3000 calculations for an exhaustive search.

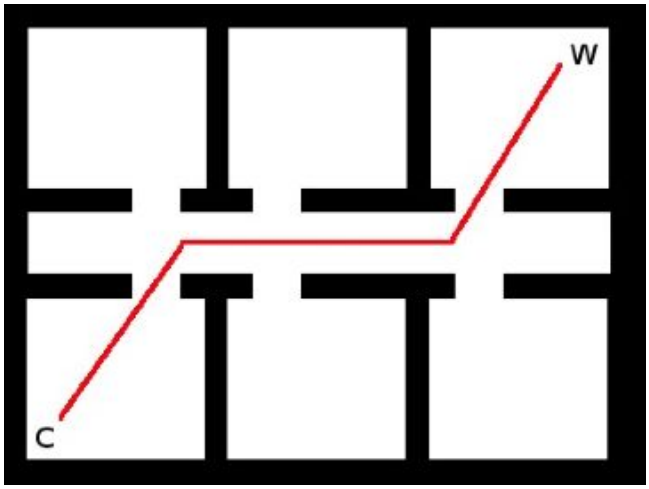


Figure 4: Office like environment with a charging station 'C' and a work site 'W'. The red line is the stylized path the robot travels on.

Experiments

In this section we present experiments to validate the theoretical results described in detail above. All experiments were performed using the robot simulator Stage[2]. The simulated robot uses simulated electrical energy, where we assume charging and discharging to be linear, with constant current for charging I_c, working I_w and driving I_d. We further ignore any effects caused by the docking mechanism, change in battery chemistry or ambient temperature.

In all experiments we roughly model a Chatterbox robot, a robot designed and built at SFU based on an iRobot Create platform. This robot has a battery capacity of approximately 2.8Ah and draws about 2A while driving. We defined an abstract work task which consumes 4A of current. Once at a charging station, the robot docks reliably and recharges with 2A. The world the robot operates in is office-like with one charging station and one work site shown in fig. 4. The obstacle avoidance and navigation controller drives the robot from the charging station to the work site and vice versa in

[2]http://playerstage.sourceforge.net

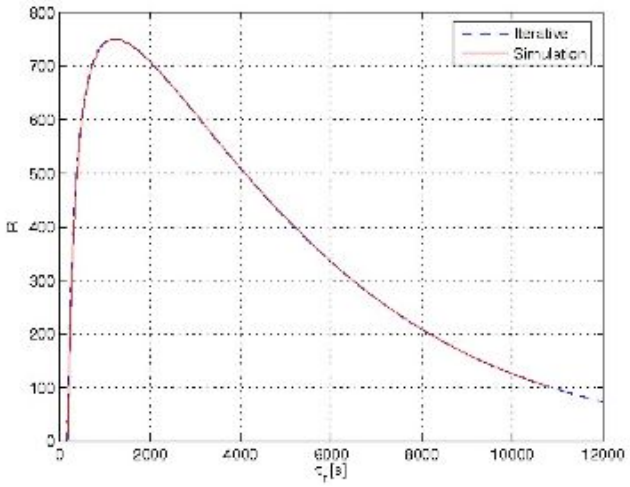


Figure 5: Comparing analytical and simulation results for accumulated reward from a **cyclic** task depending on refuelling time with an example configuration $I_c = 2.0, I_d = 2.0, I_w = 4.0, \beta = 0.9997, \tau_d \approx 85$

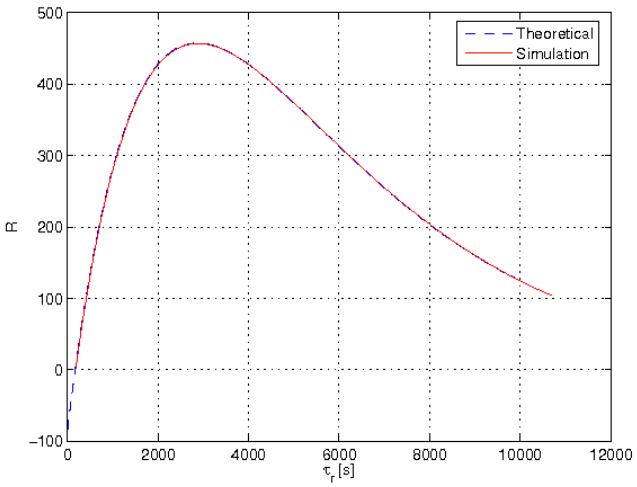


Figure 6: Comparing analytical and simulation results for accumulated reward from an **acyclic** task depending on refuelling time with an example configuration $I_c = 2.0, I_d = 2.0, I_w = 4.0, \beta = 0.9997, \tau_d \approx 85$

approximately $\tau_d = 85s$. Due to naturally occuring noise in the experimental setup the travel time may vary by up to 6 seconds. While working, the robot receives one unit of reward per second, discounted by β. Discounting occurs on a one second basis.

Cyclic Task

The goal of this experiment is to evaluate how closely our analysis from section matches a robot in a simulated environment. In this experiment the robot's task is to recharge for some time τ_r, proceed to the work site, work until the battery energy drops to $\epsilon_{w \to r} = \tau_d I_d$, return to the charging station, and repeat the process. The reward for work is discounted by $\beta = 0.9997$. To find out which τ_r maximizes the reward we varied the threshold for leaving the charging station $\epsilon_{r \to w} = \tau_r I_c$ in each try. A trial lasted for 50000 seconds ($\approx$ 13.8 hours). Figure 5 compares the accumulated reward gained over τ_r from the simulation and from the best solution obtained from the model by iterating over eq. 7. The recharging time that maximizes the reward is predicted by the model to be $\hat{\tau_r} = 1219$ and in the simulation $\hat{\tau_r} = 1170$. The difference comes from the variation in time, and therefore energy, the robot requires to travel between the charging station and the work site. Not only does this change the starting time of work which influences the reward, it also makes it necessary to give the robot a small amount of spare energy to ensure it would not run out of battery. This, in turn, delays charging and thereby influences the reward gained. However, the empirical results agree qualitatively with the values predicted by the model, and the optimal recharging time predicted by the model was within 4% of that observed in the simulation.

Acyclic Task

As before, we perform this experiment in order to compare the theoretical results with a simulation. The setup is the same as in the cyclic task experiment with the difference that the robot only has to perform one charge-work cycle. Figure 6 compares the simulation results to the analytical results. Where the general shape of the curve is similar to that in the cyclic task, it is worth to point out that the maximum reward is gained with a larger charging threshold. This is intuitively correct as the robot has only once chance to obtain a reward. It can be (depending on the discount factor) beneficial to begin work later, but to work for a longer period. For our configuration, the most profitable theoretical charging time is $\hat{\tau_r} = 2872.7$ and the best simulation results were obtained with $\hat{\tau_r} = 2880$. Again the difference between the theoretical and experimental results, barely visible in the plot, are due to imprecision in the robot simulation.

Once or forever

In a further experiment we investigate the circumstances under which it is more profitable, and hence rational, for the robot to fully deplete its energy supply while working and when it is better to choose a perpetual refuelling policy. As in the previous experiments we use a simulated Chatterbox robot with the previously described parameters in the office-like environment. For this scenario, we vary the discount rate between 0.9850 and 0.9999 in 0.0005 increments and run two sets of simulations. In the first, the robot depletes it's energy supply while working, that is, we choose the leave work threshold $\epsilon_{w \to r} = 0$. For the second set, we choose $\epsilon_{r \to w} = \tau_d \dot{e_d}$, a leave work threshold that causes the robot to keep performing work-refuel cycles forever. Since

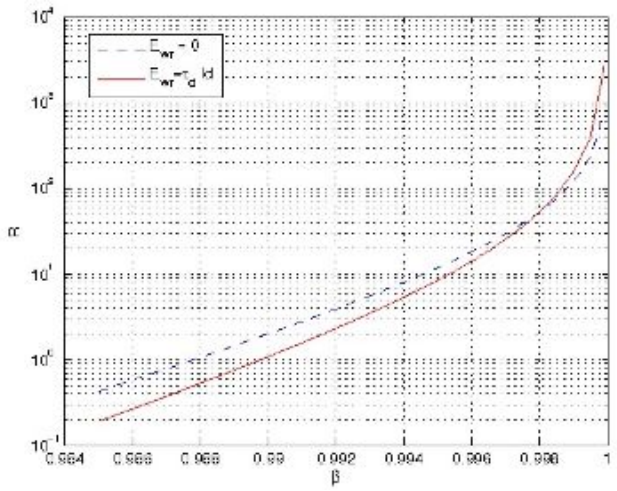


Figure 7: Reward obtained for different discount factors with two leave work thresholds. Configuration $I_c = 2.0, I_d = 2.0, I_w = 4.0, \tau_d \approx 85$

we change the discount rate we have to adapt the leave refuelling site threshold in order for the robot to earn the highest possible reward. For this determine the optimal threshold in the same way as for the previous experiments. Figure 7 depicts the rewards obtained for different discount factors with each policy. As the graph further shows, for higher discounting (smaller discount rate), it is beneficial for the robot to choose a one time work policy. Conversely, for smaller discounting (higher β), it pays to keep working. The theoretical discount rate for switching the policy from one work period to an infinite work refuel cycle is $\beta = 0.9979$, which, as the graph shows, closely resembles the experimental result.

Discussion and Conclusion

We outlined a theoretical analysis of when to refuel and for how long to refuel a robot in situations where the reward for the robot's objective is discounted over time. This discounting is, more often then not, ignored in robotics literature, although it is at the very base of rational behaviour (Stuart and Peter (2003)). We took theoretical results and demonstrated that they apply to a simulated robot. In these simulations we assumed the location of and the distance between work and refuelling station to be known. This is reasonable in the state of the art in mapping and localization, in a wide range of scenarios. We further assumed the average energy spending rates to be constant and known, something achievable in most cases. One assumption made that simplifies a real-world robot scenario is the refuelling rate. Gasoline-powered vehicles which refuel from a standard gas station have a constant refuelling rate, or close to it. However, the charging rate of a battery may depend on many factors including the charging method used, temperature, battery chemistry, and the current capacity of the battery. One useful extension of our model would be to include a realistic chemical battery recharge transfer function.

This paper has presented and analyzed a core action selection problem for autonomous agents such as animals and mobile robots: how much to fuel before working, and when to abandon working and return to fuelling, such that the value of discounted work is maximized. A simple model readily provides answers to these questions and closely predicts the observed behaviour of a robot simulation. While the model is simple, it is very general, and these results suggest that it could be of practical as well as theoretical interest. We propose it as a baseline to build upon.

Acknowledgements

The authors would like to thank Yaroslav Litus, Sarah Brown and Adam Lein for the inspiring discussions that informed this work.

References

Austin, D., Fletcher, L., and Zelinsky, A. (2001). Mobile robotics in the long term - exploring the fourth dimension. In *Proceedings IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 613–618, Wailwa, Hawaii.

Birk, A. (1996). Learning to survive. In *Procs. of the 5th European Workshop on Learning Robots*, pages 1–8, Bari, Italy.

Hedenström, A. (2003). Optimal migration strategies in animals that run: a range equation and its consequences. *Animal Behavior*, 66:631–636.

Houston, A. I. and McNamara, J. M. (1999). *Models of Adaptive Behaviour*. Cambridge University Press.

Kacelnik, A. and Bateson, M. (1996). Risky theories: The effects of variance on foraging decisions. *American Zoologist*, 36:402–434.

Litus, Y., Vaughan, R. T., and Zebrowski, P. (2007). The frugal feeding problem: Energy-efficient, multi-robot, multi-place rendezvous. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 27–32, Roma, Italy.

McFarland, D. and Spier, E. (1997). Basic cycles, utility and opportunism in self-sufficient robots. *Robotics and Autonomous Systems*, 20:179–190.

Pfeifer, R. (1996). Building fungus eaters: Design principles of autonomous agents. In *From Animals to Animats, Cambridge, MA: MIT Press*, volume 4.

Seth, A. K. (2007). The ecology of action selection: Insights from artificial life. *Philos. Trans. R. Soc. B*.

Silverman, M., Jung, B., Nies, D. M., and Sukhatme, G. S. (2003). Staying alive longer: Autonomous robot recharging put to the test. Cres-03-015, Center for Robotics and Embedded Systems, University of Southern California.

Silverman, M., Nies, D. M., Jung, B., and Sukhatme, G. S. (2002). Staying alive: A docking station for autonomous robot recharging. In *IEEE International Conference on Robotics and Automation*, pages 1050–1055, Washington D.C.

Spier, E. and McFarland, D. (1997). Possibly optimal decision-making under self-sufficiency and autonomy. *Journal of Theoretical Biology*, 189(3):317–331.

Stephens, D. W., Brown, J. S., and Ydenberg, R. C. (2007). *Foraging*. University of Chicago Press.

Stephens, D. W. and Krebs, J. R. (1986). *Foraging Theory*. Princton University Press.

Stuart, R. J. and Peter, N. (2003). *Artificial Intelligence: A Modern Approach*. Prentice Hall.

Toda, M. (1982). *Man, Robot and Society*. Martinus Nijhoff Publishing.

Varian, H. R. (1992). *Microeconomic Analysis*. W. W. Norton.

Walter, W. G. (1963). *The Living Brain*. W. W. Norton, New York.

Wawerla, J. and Vaughan, R. T. (2007). Near-optimal mobile robot recharging with the rate-maximizing forager. In *European Conference On Artifical Life (ECAL)*, pages 776–785, Lisbon, Portugal.

Hierarchical Components and Entity-based Modelling in Artificial Life

Matt Webster[1] and Grant Malcolm[1]

[1]Department of Computer Science, University of Liverpool, Liverpool, L69 3BX, UK.
matt@liverpool.ac.uk

Abstract

We present notions of entity and entity-based model that are applicable to artificial life. We illustrate these notions by giving an abstraction of Langton's loops: loop-like structures that reproduce in a cellular automaton (CA). Our abstraction takes as entities persistent configurations of the cellular automaton, and shows how these entities may be combined to form more complex entities. The resulting entity-based model of Langton's loops describes the functionality and interrelationships of these components, abstracting from their actual realisation in a cellular automaton. As well as providing a basis for the study of ecologies of interacting entities in artificial life, our approach provides a useful intermediate level of abstraction that can relate top-down and bottom-up approaches to the study of life-like systems.

Introduction

One of the earliest results in the field of artificial life was given by von Neumann (1966), in which an automaton on a cellular automaton (CA) grid was shown to reproduce itself. Subsequent improvements by Codd (1968), Langton (1986), and others (Sipper, 1998) showed an intriguing approach to the modelling of life that differed from much of mainstream biology. In biology, abstract processes such as reproduction, metabolism and evolution are specified based on observation of existing life forms. We can characterise this as a top-down approach, in which we attempt to understand life and its systems through abstract models and classifications. The opposite approach is widely-adopted in artificial life, in which bottom-up models of life are generated. This difference hs been well-documented (Langton, 1986; Sipper, 1995; Bonabeau and Theraulaz, 1996), and is a key characteristic of much work within the field of artificial life. The question of which approach to use was also explored by Bedau (1998) with the question, 'Does the essence of life involve matter or form?'

The work by Rosen (1991) on the modelling of life has received attention recently in the artificial life community (Chu and Ho, 2006; Louie, 2007; Wolkenhauer, 2007; Chu and Ho, 2007). Rosen argued that reductionistic models of life were not adequate for giving a formal description of the organisation observed, i.e., life cannot be reduced to physical laws of the Universe. Instead, Rosen suggested a high-level description of life based on systems consisting of interacting components. This kind of approach has been adopted elsewhere in the artificial life community, including work by Adams and Lipson (2003) and ourselves (Webster and Malcolm, 2007b,a).

The apparent disparity between top-down and bottom-up approaches to modelling life presents a problem for the field of artificial life. Both approaches have been proven to be valuable, and many interesting results have been obtained in both directions. However, the question of how these differing approaches are related is still an open question. In this paper we present a way of relating these two levels. exemplified through a 'reverse engineering' of Langton's loops, a seminal example of implementing artificial life in a cellular automaton. We take a paradigmatic artificial life reproducer, Langton's loop, and show how its interacting components can be abstracted from the low-level state of the cellular automaton. We show that these components can be tied together using formal constraints that are based on observation of the cellular automaton itself. The result is a high-level formal description of Langton's loop, which abstracts the functionality from the particular implementation details of the loop, and captures the interaction of various components within the reproducing whole. We describe how the high-level abstract model can be fitted to a variety of low-level models through a refinement process.

This relationship between high- and low-level models is inspired by work in the field of software engineering, in which high-level specifications of a software system are refined to produce a low-level implementation. A typical example of this would be the transition from a software design specification in an abstract modelling language (e.g., UML), which is refined to a software specification in a high-level language (e.g., C++), which is then refined further to a low-level language implementation (e.g., using Intel 64 assembly language) during the compilation process. There may be several different low-level language implementations that match the higher-level specifications. Simi-

larly, the high-level component-based specification of Langton's loop can be refined to a number of different low-level specifications, in which the implementation details (such as the transition rule, or number of states of the cells in the cellular automaton) vary, but which all satisify the high-level 'reverse-engineered' component-based description of a reproducing loop. These refinements need not be trivial alternatives, such as re-labelled states, as the high-level component-based specification would essentially describe any reproducing loop satisfying certain constraints. Therefore, this specification could be refined to different cellular automata (with different states, topologies and transition rules), or even non-discrete examples of cellular computation, in which the data-paths of Langton's loop are continuous or noisy communication channels in the vein of those described by information theory (Shannon, 1948).

In the following section, we review the construction of Langton's loops, and present abstract entities that capture the functionality of the various components that build up these loops. We also illustrate how our approach allows for a dynamic system in which entities are created and change relationship between one another.

Langton's Loops

Langton's loops provide a simple example of self-reproduction in cellular automata (Langton, 1984). The question of whether an automaton could reproduce, thereby exhibiting at least one life-like trait, was first studied by von Neumann (1966). He was able to provide a positive answer by exhibiting a *universal constructor*: an automaton which, given as input a description of any automaton, could construct that automaton and copy the input description as that automaton's input. Therefore, given a description of itself, the universal constructor reproduces by building a copy of itself with its own self-description. Von Neumann's universal constructor required cellular automata with 29 states, and was later simplified by Codd (1968). While universality is an interesting feature from the point of view of computability, it is not, as Langton (1984) noted, a necessary feature of reproduction, and indeed it seems unlikely to figure in biological reproduction. Langton's loops were the result of his search for a simpler example of a reproducer, though not so simple as to become trivial. In particular, Langton required that the process of reproduction be 'actively directed' by the reproducer, and that the reproducer store information directing its reproduction that is both interpreted (as instructions) and uninterpreted (as data that is copied or transcribed).

Langton states[1] that 'the idea for this simple self-reproducing configuration came out of a study of the components of Codd's universal constructor', and it is our goal to elucidate in what sense both Langton's and Codd's configurations can be said to have 'components'.

A cellular automaton is an example of a *deterministic* transition system: at a given moment each cell in the grid is in a particular state; its state at the next moment of time is determined by its own state and the states of its neighbours (the 5-cell neighbourhood for Langton's loops) according to a *transition function*, which it is usually convenient to present as a table listing transitions for all possible configurations of a neighbourhood's states. Langton's loops are realised on a cellular automaton where each cell is in one of eight states. Throughout the remainder of this paper we will refer to this state set as $\mathsf{S} = \{0, 1, 2, 3, 4, 5, 6, 7\}$. Each of these states has a particular function: for example, state 0 represents 'blank space', though it also provides direction for instructions such as state 7, which causes a 'data-path' to be extended; these data-paths are the basic structure of Langton's loops, as they were also in Codd's universal constructor. They are formed from two rows of sheath cells with a row of 'core cells' between, as pictured in Figure 1.

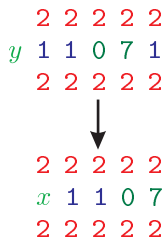


Figure 1: A data-path. The data signal '7 0' travels along the data-path. The state of cell x in the updated state is dependent on the state of cell y. If $y = 1$, then in the updated state $x = 1$.

The function of a data-path is the transmission of a data signal along its length. For example, in Figure 1 the '7 0' signal has been shifted one cell to the right. This process continues as long as the data-path does. data-paths can be 'capped' using a cell in state 2, as seen in Figure 2.



Figure 2: A capped data-path.

The cap allows the data signal to effect the extension of the data-path. This extension process is what allows Langton's loop to extend an 'arm', which loops around to complete the act of reproduction.

Another kind of data-path functionality in Langton's loop is the T-connection, as seen in Figure 3. The data-path sends a signal to a point where two other data-paths are connected. The intersection of the three paths is at a particular cell.

[1] *op. cit.*, p.137

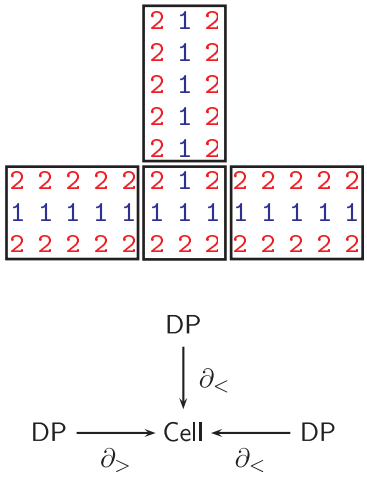


Figure 3: A T-connection. The image at the top shows a T-connection as realised in a cellular automaton; the diagram below describes the same connectivity in the schematic form described in the following subsection.

When the data signal arrives at the cell, it is duplicated and sent out through each of the other two connected data-paths.

Components

We want now to abstract, as far as we can, the functionality described above from its realisation in a particular cellular automaton. Our goal is to be able to identify individual components, such as data-paths, capped data-paths and T-connections, and capture the way they interact.

The most important functional unit is the data-path, consisting of a series of core cells sandwiched between two rows of sheath cells. The function of such a component is to transmit signals: it acts as a queue for the data in the core cells, which move, cell by cell, along the data-path. To capture this, we define a *data-path* to be a transition system (which we shall call DP) whose states are pairs $(n; d)$, where $n > 0$ and d is an n-tuple of states, $d = d_1, \ldots, d_n$ with $d_i \in \mathsf{S}$; and with the following schematic transition rule:

$$(n;\ d_1, \ldots, d_n) \longmapsto (n;\ x, d_1, \ldots, d_{n-1}) \ . \qquad (1)$$

I.e., all the data values in the path move up one space (left to right), and a new value x enters at the start-cell. Note that this is non-deterministic: there are no constraints on what this new value is, beyond $x \in \mathsf{S}$. Thus, we might have both

$$(5;\ 1, 1, 0, 7, 1) \longmapsto (5;\ 1, 1, 1, 0, 7) \ ,$$

as in Figure 1, and

$$(5;\ 1, 1, 0, 7, 1) \longmapsto (5;\ 7, 1, 1, 0, 7) \ .$$

Insofar as a data-path represents an actual sequence of core cells in the cellular automaton, the value that 'enters' the data-path will be determined by the values of the cells in the neighbourhood of the start-cell d_1, but that is exactly what we are abstracting from: functionally, a data-path allows arbitrary signals to be transmitted.

We can *ground* a data-path in a cellular automaton by means of a mapping $h : \mathsf{CA} \to \mathsf{DP}$ of the state-space of the cellular automaton to the state-space of the data-path. That is, if the state-space of CA consists of configurations $\vec{s} \in \mathsf{S}^{pq}$, with pq the number of cells in the grid (which we could allow to be infinite), then such a mapping would project a pq-tuple $\vec{s}$ to an n-tuple $(n;\ s_{i_1}, \ldots, s_{i_n})$. This would allow projecting a configuration to a tuple of states of an arbitrary collection of cells (i.e., they need not be contiguous), which wouldn't capture the intention of picking out a particular data-path in the configuration. However, we impose the condition that the mapping *preserve transitions*, i.e., if $\vec{s} \longmapsto \vec{s'}$ is a transition of the cellular automaton (viewed as a transition system), then $h(s) \longmapsto h(s')$ in DP. This means that the mapping must pick out a tuple of cells that acts as a data-path: we must be able to observe signals moving cell-by-cell along those tuples.

The basic building block of all components, including data-paths, is the individual cell. As a constituent part of the cellular automaton grid, its functionality is dependent on the states of the cells in its neighbourhood, so at an abstract level, its functionality is simply to be in a particular state. We define a *cell* to be a transition system (call it Cell) with state set S and universal transition relation; i.e., $s \longmapsto s'$ for all $s, s' \in \mathsf{S}$. A *grounded cell* is a transition-preserving map $h : \mathsf{CA} \to \mathsf{Cell}$. Note that the requirement that h preserve transitions is trivial, because any state can make a transition to any state in Cell.

This gives us a means of glueing data-paths together. Define $\partial_<, \partial_> : \mathsf{DP} \to \mathsf{Cell}$ by

$$\begin{aligned} \partial_<(n;\ d_1, \ldots, d_n) &= d_1 \\ \partial_>(n;\ d_1, \ldots, d_n) &= d_n \end{aligned}$$

so that $\partial_<$ picks out the 'start cell' of a data-path, and $\partial_>$ picks out the 'end cell'. Again, these maps are trivially transition-preserving.

Now, to join together two data-paths, let a *data-path connection*, $\mathsf{DP} \Rightarrow \mathsf{DP}$, be the transition system whose state-set consists of pairs (p_1, p_2), with each p_i a DP-state such that $\partial_>(p_1) = \partial_<(p_2)$, i.e., p_1 and p_2 communicate by sharing a cell which is both the end-cell of p_1 and the start-cell of p_2. Let the transitions of $\mathsf{DP} \Rightarrow \mathsf{DP}$ be given by

$$\begin{gathered} ((m;\ d_1, \ldots, d_m), (n;\ d_m = e_1, \ldots, e_n)) \longmapsto \\ ((m;\ x, \ldots, d_{m-1}), (n;\ d_{m-1}, d_m = e_1, \ldots, e_{n-1})) \end{gathered}$$

This effectively constrains the non-determinism in (1): the new value that enters the second data-path p_2 is determined by the values at the end of the first data-path p_1. For exam-

ple, we have

$$((3;\ 1,0,7),\ (4;\ 7,1,1,0)) \longmapsto$$
$$((3;\ 7,1,0),\ (4;\ 0,7,1,1))\ .$$

Effectively, we can think of this as one long data-path, in which the above transition could be rewritten as

$$(6;\ 1,0,7,1,1,0) \longmapsto (6;\ 7,1,0,7,1,1)\ .$$

As with the previous components, a data-path connection can be grounded in a cellular automaton by providing transition-preserving morphisms from the cellular automaton to the individual subcomponents: the two data-paths and their shared cell. Note that there is no requirement that grounded connected data-paths be orthogonal.

Transcription of the data within Langton's loops is achieved by branches in the data-paths: as signals move through a T-connection, they are copied along the two branches, as in Figure 3. Abstractly, we have three data-paths p_1, p_2, and p_3, all sharing a single cell at their intersection, which is the end-cell of p_1 and the start-cell of both p_2 and p_3 — a diagrammatic representation is given in the lower half of Figure 3. Thus, a *T-connection* is set up by two data-paths $\partial_<$-connected to the end-cell of a single data-path, and we define $\mathsf{DP} \Rightarrow \mathsf{DP}|\mathsf{DP}$ to be the transition system whose states are triples (p_1, p_2, p_3) with $\partial_>(p_1) = \partial_<(p_2) = \partial_<(p_3)$. The transitions are given by the fact that both (p_1, p_2) and (p_1, p_3) are data-path connections: in brief, $(p_1, p_2, p_3) \longmapsto (p_1', p_2', p_3')$ iff both $(p_1, p_2) \longmapsto (p_1', p_2')$ and $(p_1, p_3) \longmapsto (p_1', p_3')$ as data-path connections. It should be clear that the effect of these definitions is to capture the data-flow and copying of the T-connections pictured in Figure 3. Again, a T-connection is grounded in a cellular automaton by grounding its subcomponents.

Clearly, we can specify a great variety of structures built from data-paths and shared cells in a diagrammatic way along the lines of the T-connection shown in Figure 3. For example, a *loop* is a loop of connected data-paths, as shown in Figure 4. In more detail, we have data-paths p_1, p_2, p_3 and p_4 with structural constraints

$$\begin{aligned}
\partial_>(p_1) &= \partial_<(p_2) \\
\partial_>(p_2) &= \partial_<(p_3) \\
\partial_>(p_3) &= \partial_<(p_4) \\
\partial_>(p_4) &= \partial_<(p_1)\ .
\end{aligned}$$

As with T-connections, the transitions are determined by the fact that each pair p_i, p_{i+1} (mod 4) is a data-path connection. It should be clear that the data in these paths continually circles around the loop. As with the other components described above, a loop is grounded by grounding the subcomponents, as illustrated in Figure 4.

In fact, we have not quite modelled Langton's loops: the essential missing ingredient is the arm that 'interprets' the data in the loop, extending into the grid, turning, and eventually looping back on itself. Let us call such an interpreting arm a *capped data-path*, and provide it with the structure of a transition system, which we shall call CDP. The states of CDP are the states of DP, but we shall use the suggestive notation $(n;\ d\,]$, $n > 0$ and $d = d_1, \ldots, d_n$ a data-path of length n. The transitions of CDP should capture the effects of interpreting the signals in the data-path. For example, the signal '7' extends the path:

$$(n;\ d_1, \ldots, d_{n-1}, 7\,] \longmapsto (n{+}1;\ x, d_1, \ldots, d_{n-1}, 1\,] \quad (2)$$

Note also that this transition allows data to move along the capped data-path, just as for data-paths themselves; in particular, a new value x non-deterministically 'enters' the data-path. As with data-paths, capped data-paths can be $\partial_<$-connected to data-paths, giving a transition system $\mathsf{DP} \Rightarrow \mathsf{CDP}$, and so may also form T-connections. We shall give further transitions in the following subsection; we end this subsection by giving the abstract, functional form of Langton's loops: an *L-loop* is a tuple (p_1, p_2, p_3, p_4, c), where the p_i form a loop of data-paths, and c is a capped data-path $\partial_<$-connected to p_4:

$$\begin{aligned}
\partial_>(p_1) &= \partial_<(p_2) \\
\partial_>(p_2) &= \partial_<(p_3) \\
\partial_>(p_3) &= \partial_<(p_4) \\
\partial_>(p_4) &= \partial_<(p_1) = \partial_<(c)\ .
\end{aligned}$$

With the definitions given above, we could fill an L-loop with '7' signals, and the transitions allow the capped data-path to extend indefinitely.

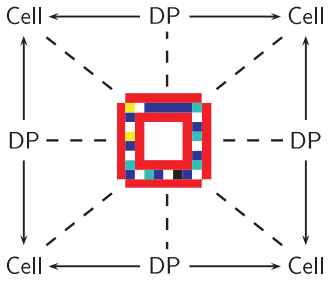


Figure 4: Connections between data-paths and cells for Langton's loop. The outer square denotes the structure of the component data-paths, while the dotted lines indicate a grounding of the subcomponents of the loop in a cellular automaton.

Introduction of New Components

We have seen how we can model components of Langton's loops, building larger components from subcomponents in a diagrammatic, hierarchical way. The configurations we have seen are all static, in the sense that components are

related in a fixed way that never alters. Obviously, since the purpose of Langton's loops is to demonstrate reproduction by the creation of one loop by another, we need to allow for components that are created or destroyed, or even change their structural relationships.

In the case of Langton's loops, all the structural changes are brought about by the actions of the interpreting arm: the capped data-path attached to the loop. We saw in (2) above how we could capture the effect of a '7' signal at the cap of the data-path, namely by extending the length of the path. The other relevant signal for Langton's loop is a '4' signal, which is interpreted as an instruction to create a left-hand turn in the data-path. In fact, we shall simplify things here, as a left-hand turn in Langton's cellular automaton is created by *two* '4' signals. However, this simplification is quite in keeping with our goal of abstracting the functionality of the relevant components. In accordance with this simplification, when a '4' signal reaches the cap of the interpreting arm, the left-hand turn is achieved by turning the capped data-path into a data-path, with a capped data-path $\partial_<$-connected to its end-cell. In our abstract, functional view, two things happen. The first is that the diagram representing the component changes from

$$\mathsf{Cell} \xleftarrow{\partial_<} \mathsf{CDP} \xrightarrow{\partial_>} \mathsf{Cell}$$

to

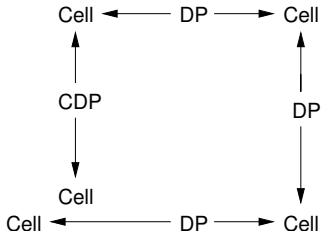
$$\mathsf{Cell} \xleftarrow{\partial_<} \mathsf{DP} \xrightarrow{\partial_>} \mathsf{Cell} \xleftarrow{\partial_<} \mathsf{CDP} \ .$$

The second thing that happens is that the states of the transition system denoted by this diagram makes the corresponding change:

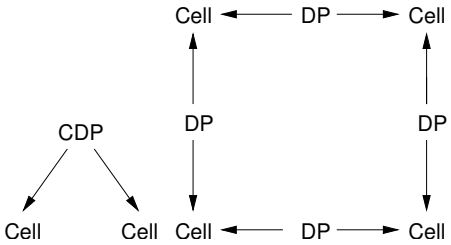
$$(n;\ d_1 \dots d_{n-1}, 4] \longmapsto ((n;\ x, d_1, \dots, d_{n-1}), (1;\ d_{n-1}])$$

This transition turns a capped data-path into a data-path and a new capped data-path; i.e., extending the notation above, the transition system CDP becomes $\mathsf{DP} \Rightarrow \mathsf{CDP}$.

A CA grid imposes constraints on what may happen in components that are grounded in it, but there are purely topological constraints that may be imposed on components, whether or not they are grounded in a cellular automaton. For example, and here we consider functionality rather than topology, we may wish to state that, after turning three times, an interpreting arm is heading back towards itself, and will sever the umbilical cord that ties the parent to the child loop. When it meets its original arm, the loop is closed and the original data-path is split: the first part becomes a capped data-path, while the remainder becomes a part of the child loop. Diagrammatically, the state



becomes



In accordance with this, the states of the relevant components change as follows:

$$((m;\vec{a}), (n;\vec{b}), (m';\vec{c}), (n;\ \vec{d}]) \longmapsto$$
$$(m-m';{}_{m-m'}|\vec{a}], ((m';\vec{a}|_{m-m'}), (n;\vec{b}), (m';\vec{c}), (n;\vec{d}))$$

provided that $m' < m$, and where ${}_p|\vec{a}$ denotes the first p elements of $\vec{a}$, and $\vec{a}|_p$ denotes the 'remainder' of $\vec{a}$ from the pth position on. Here again we simplify matters: in Langton's automaton, the capped data-path belonging to the parent loop bears a '5' signal that will close off the capped data-path; also, the data-path in the child loop contains a '6' signal that will cause a new capped data-path to be created, causing the process of reproduction to begin anew. This extra functionality can be captured straightforwardly, using the above techniques, so we omit the details. The main point is that, even without these details, we have captured reproduction of loops in an abstractly functional, entity-based model.

Entities and Some Technical Details

In the preceding subsections, we have tried to avoid going into too many technical details, preferring to keep the exposition at an intuitive level. From our use of diagrams and transition-preserving mappings, it may be clear that category theory provides a natural setting for our constructions. Indeed, the constructions of transition systems such as $\mathsf{DP} \Rightarrow \mathsf{DP}$ are limit constructions in a category of transition systems and transition-preserving morphisms, as shown in (Malcolm, 2006). This is very much in keeping with the slogan 'Behaviour is Limit' from the Categorical Systems Theory of Goguen (1992). In this view, mappings between transition systems can be seen as constraints: for example, the two mappings in the configuration $\mathsf{DP} \Rightarrow \mathsf{DP}$ state that the end-cell of one data-path is the start-cell of the other, and as noted above, this constrains the non-determinism in (1). The limit of a configuration then consists of all possible behaviours that meet the given constraints; in the example of $\mathsf{DP} \Rightarrow \mathsf{DP}$, the limit gives all possible behaviours where the

'output' of the first data-path is the 'input' of the second'. Thus our approach is quite general in that the limit construction captures all the ways in which components may interact.

In our functional model of Langton's loops, we take an *entity* to be a component, such as a data-path, which is a transition system *in a diagram*, such as the diagram of Figure 3. We might say that an entity is necessarily 'situated': it stands not alone (except in the case of a trivial diagram) but in relationships with other entities, particularly any subcomponents it may have, and may share with other entities. Since complex objects are created by limit constructions from subcomponents, technically, our entities are *sheaves*: see Malcolm (2008) for details. Again, this is in keeping with the slogan 'Objects are Sheaves' of Goguen (1992).

Conclusion

As we described in the introduction, the aim of our work is to create a formal bridge between top-down and bottom-up models of life forms. We have shown how an abstract model of cellular automaton-based artificial life forms can be constructed, based on a formal description of various interacting components of the life form. We presented an entity-based model of Langton's loops that abstracts the functionality of the various components from their actual realisation in a cellular automaton. Interactions between data-paths and cells are formalised in terms of constraints on their transition systems. Our approach allows for entities to be constructed hierarchically from subcomponents, and to interact with each other through shared subcomponents: for example, two data-paths communicate values through a shared cell in the $\mathsf{DP} \Rightarrow \mathsf{DP}$ configuration. We can also model dynamic configurations where entities may be created and adopt varying relationships with other entities. This is useful in modelling reproduction, as it is often the case that the reproduction process involves creating various components of the offspring. We gave an example of the introduction of new components for Langton's loops, in the case where a data-path approaches a cell, and then branches off in a new direction. This is a repeating process that results eventually in the creation of a second loop, showing that our abstract component-based model is able to model reproduction.

Therefore, we can see the different entities in the model as different behaviours, which may combine to form more complex behaviours. For example, a data-path is an entity that propagates information; this can be combined with other data-paths, cells and a capped data path in order to form a conglomerate entity capable of reproduction.

We showed how these abstract models can be related ('grounded') to a particular implementation of a loop, through a mapping from the transition system of the former, to the transition system of the latter. This notion of grounding allows an actual realisation of the model in a cellular automaton to be viewed as a refinement of the model. Conceptually, grounding can also be seen as the imposing of constraints on the model by the cellular automaton and its topology: in much the same way, a model grounded in a real-life system would be constrained by the laws of physics.

Related Work

In this paper we have described an approach to the development of entity-based models of artificial life systems. A recent report by Wheeler et al. (2002) highlighted many of the challenges in the area of artificial life modelling.

Entity-based models of reproduction, such as those described earlier, have been used before in artificial life. For example, Adams and Lipson (2003) give a formal universal framework for reproduction based on 'subsystems' within an environment. The subsystems are analogous to the entities or components discussed in this paper. One possible property of a subsystem is reproduction. Since Adams and Lipson do not preclude the possibility of subsystems consisting of other subsystems, we could re-frame the discussion of Langton's loop with each component as a subsystem. The system consisting of these subsystems, i.e., the loop, has the property of reproduction. Another example is the work by Hordijk et al. (1998) on embedded-particle models, which model emergent functionality in evolved cellular automata. The approach is similar to our own in that an abstract description of organised behaviour is formed, although the emphasis is more on abstracting higher-level behaviour from an existing CA configuration than on providing a general means of describing the behaviour of hierarchical systems.

Abstract, high-level models of life-like phenomena have also been explored in biology. Lazebnik (2002) describes the fundamental differences between the language of biology and engineering, and posits that the formalisms of engineering permit a greater understanding of complex systems such as life. Discrete, top-down models of biological processes have also been described, e.g., the approach of Laubenbacher and Mendes (2006) to modelling biochemical networks.

The work in this paper may be seen as an 'intellectual progeny' of our earlier work on formal affordance-based modelling of reproduction, in which we described a reproduction system in terms of a labelled transition system, with entities present in various states, and affordances relating the entities which cooperate in various parts of the reproductive process (Webster and Malcolm, 2007b,a). These affordance-based models are therefore entity-based, and are in this way related to the entity-based models presented here.

Our work is also influenced by the work of Rosen (1991, 1999) on modelling life-like processes. Rosen argues that a reductionistic 'machine metaphor', in which life is seen simply as the result of the underlying physical laws of the Universe, is insufficient to capture the full complexity of life: self-organisation, reproduction and so on. Rosen suggests that a more holistic model of life is needed, and that a sufficient model for life might be obtained at a natural level of

abstraction. For example, if a large conglomeration of cells (e.g., an animal) appears to reproduce, then the natural way to view that process is not in terms of the cells, but in terms of the sub-components of the animal that enable reproduction, e.g., sex organs, nervous system, etc. Therefore, our attempt to model artificial life forms like Langton's loop is in the same vein. Of course, Langton's loop can be described completely by its formal definition; however, it may also be interesting to describe formally the same system at a different level of abstraction, as the aim of abstraction is to provide greater insight than can be obtained through individual case studies.

Future Work

We have presented Langton's loops as a case study in our approach to modelling hierarchical artificial life systems. An obvious question that arises is: what other systems can usefully be captured in this way? There are several candidate systems in artificial life and in biology that seem to have a natural hierarchical structure, and one thrust of future research would be to construct entity-based models of these.

An example from artificial life is the mechanical reproducing robots described by Zykov et al. (2005), in which a number of modular robots interact in order to reproduce a conglomerate multi-robot entity. The robots are constrained to connect at certain points, mirroring the cells at which one or more data-paths may be connected.

Another interesting application of our work would be to model the component behaviour of more complex forms of reproduction, including evolution (Sayama, 1999) and sexual reproduction (Oros and Nehaniv, 2007). These seem particularly feasible examples in that they are based on the same conceptual mechanisms as Langton's loops. In the case of sexual reproduction, there is a link to our earlier work (Webster and Malcolm, 2007b,a) on reproduction modelling based on Gibson's theory of affordances (Gibson, 1979). Sexual reproduction can be seen as a collaborative arrangement, in which each partner affords the other the act of reproduction, and therefore there is a possibility of extending the component-based models presented in this paper to include a notion of affordances.

In the field of biological systems, we would expect to be able to model simple ecologies such as those of bacteriophage viruses and host cells. There are obvious entities or components which we may identify, such as viral RNA, generation of offspring based on proteins, cell walls, and so on. We have given a simplified model of the T4 bacteriophage lifecycle in (Webster and Malcolm, 2008), and we expect the extension of this to a more realistic hierarchical model to be reasonably straightforward. Other interesting biological examples with significant hierarchical structure would include meiosis, and possibly tissues: for example, hepatic tissue has a rich structure that provides a rich functionality (Teutsch, 2004).

A useful feature of our entity-based models is that they are formal. In principle, it should therefore be possible to prove formally that a certain group of entities in a given model is capable of the act of reproduction. One way of doing so would be to construct a 'minimal' model of reproduction, and show that the given model is a refinement of it. A minimal model would be purely schematic, consisting of just two states, the first of which contains just one entity, and which has a transition to the second state, which contains just two entities: notionally, these would be the original entity and its offspring. A refinement of this by another model would relate these two entities to entities in the other model, for example a loop and its offspring loop in the case of Langton's loops. In addition, the transition from the first schematic state to the second would be related to a path in the other model that leads from a state containing the progenitor to a state containing the offspring. In general, a model may refine this minimal model in more than one way, picking out different entities that reproduce: as a ready example from biology, it may be possible to view an organism as reproducing itself, while another view of the same process may see the organism's genetic material as the entity that is reproduced. While the intuitions are clear, the technical details of refinement for our hierarchical, entity-based models still need to be spelt out.

References

Adams, B. and Lipson, H. (2003). A universal framework for self-replication. In *European Conference on Artificial Life (ECAL'03)*, pages 1–9.

Bedau, M. A. (1998). Four puzzles about life. *Artificial Life*, 4:125–140.

Bonabeau, E. W. and Theraulaz, G. (1996). Why do we need artificial life? In Langton, C. G., editor, *Artificial Life: An Overview*. MIT Press.

Chu, D. and Ho, W. K. (2006). A category theoretical argument against the possibility of artificial life: Robert Rosen's central proof revisited. *Artificial Life*, 12:117–134.

Chu, D. and Ho, W. K. (2007). The localization hypothesis and machines. *Artificial Life*, 13:299–302.

Codd, E. (1968). *Cellular Automata*. Academic Press, New York.

Gibson, J. J. (1979). *The Ecological Approach to Visual Perception*. Houghton Mifflin, Boston. ISBN 0395270499.

Goguen, J. A. (1992). Sheaf semantics for concurrent interacting objects. *Mathematical Structures in Computer Science*, 11:159–191.

Hordijk, W., Crutchfield, J. P., and Mitchell, M. (1998). Mechanisms of emergent computation in cellular automata. In *Proceedings of the Fifth International Conference on Parallel Problem Solving From Nature—PPSN V*. New York: Springer.

Langton, C. G. (1984). Self-reproduction in cellular automata. *Physica D: Nonlinear Phenomena*, 10:135–144.

Langton, C. G. (1986). Studying artificial life with cellular automata. *Physica D*, 22:120–149.

Laubenbacher, R. and Mendes, P. (2006). A discrete approach to top-down modeling of biochemical networks. In Kriete, A. and Eils, R., editors, *Computational Systems Biology*. Elsevier Academic Press.

Lazebnik, Y. (2002). Can a biologist fix a radio? — or, what I learned while studying apoptosis. *Cancer Cell*, 2(3):179–182.

Louie, A. H. (2007). A living system must have noncomputable models. *Artificial Life*, 13:293–297.

Malcolm, G. (2006). Sheaves and structures of transition systems. In Futatsugi, K., Jouannaud, J.-P., and Meseguer, J., editors, *Algebra, Meaning and Computation: Essays dedicated to Joseph A. Goguen on the Occasion of his 65th Birthday*, volume 4060 of *Lecture Notes in Computer Science*, pages 405–419. Springer.

Malcolm, G. (2008). Sheaves, objects, and distributed systems. *Electronic Notes in Theoretical Computer Science*. To appear.

Oros, N. and Nehaniv, C. L. (2007). Sexyloop: Self-reproduction, evolution and sex in cellular automata. In *Proceedings of the 2007 IEEE Symposium on Artificial Life (CI-ALife 2007)*.

Rosen, R. (1991). *Life Itself*. Columbia University Press.

Rosen, R. (1999). *Essays on Life Itself*. Columbia University Press. ISBN: 978-0231105118.

Sayama, H. (1999). A new structurally dissolvable self-reproducing loop evolving in a simple cellular automata space. *Artificial Life*, 5:343–365.

Shannon, C. E. (1948). A mathematical theory of communication. *The Bell System Technical Journal*, 27:379–423, 623–656.

Sipper, M. (1995). An introduction to artificial life. *Explorations in Artificial Life (special issue of AI Expert)*, pages 4–8.

Sipper, M. (1998). Fifty years of research on self-replication: An overview. *Artificial Life*, 4:237–257.

Teutsch, H. (2004). Modular design of the liver of the rat. In Malcolm, G., editor, *Multidisciplinary Approaches to Visual Representations and Interpretations*, volume 2 of *Studies in Multidisciplinarity*, pages 63–67, Elsevier.

von Neumann, J. (1966). *Theory of Self-Reproducing Automata*. University of Illinois Press. Edited by Arthur W. Burks.

Webster, M. and Malcolm, G. (2007a). Reproducer classification using the theory of affordances. In *Proceedings of the 2007 IEEE Symposium on Artificial Life (CI-ALife 2007)*, pages 115–122. IEEE Press.

Webster, M. and Malcolm, G. (2007b). Reproducer classification using the theory of affordances: Models and examples. *International Journal of Information Technology and Intelligent Computing*, 2(2).

Webster, M. and Malcolm, G. (2008). Classifying and relating affordance-based models of reproduction. Submitted for publication.

Wheeler, M., Bullock, S., Paolo, E. D., Noble, J., Bedau, M., Husbands, P., Kirby, S., and Seth, A. (2002). The view from elsewhere: Perspectives on ALife modelling. *Artificial Life*, 8:87–100.

Wolkenhauer, O. (2007). Interpreting Rosen. *Artificial Life*, 13:291–292.

Zykov, V., Mytilinaios, E., Adams, B., and Lipson, H. (2005). Self-reproducing machines: A set of modular robot cubes accomplish a feat fundamental to biological systems. *Nature*, 435(7038):163–164.

Investigating Emergence by Coarse Graining Elementary Cellular Automata

Andrew Weeks[1], Fiona Polack[1] and Susan Stepney[1]

[1]Department of Computer Science, University of York, UK, YO10 5DD

Abstract

We extend coarse graining of cellular automata to investigate aspects of emergence. From the total coarse graining approach introduced by Israeli and Goldenfeld, *Coarse-graining of cellular automata, emergence, and the predictability of complex systems*, Phys. Rev. E, 2006, we devise partial coarse graining, and show qualitative differences in the results of total and partial coarse graining. Mutual information is used to show objectively how coarse grainings are related to the identification of emergent structure. We show that some valid coarse grainings have high mutual information, and are thus good at identifying and predicting emergent structures. We also show that the mapping from lower to emergent levels crucially affects the quality emergence.

Introduction

We are interested in observing and modelling complex emergent systems, with the goal of understanding how we could begin to specify and implement engineered emergent systems. Emergence is variously characterised; we start from Ronald et al's definition of emergence: "The language of design L1 and the language of observation L2 are distinct, and the causal link between the elementary interactions programmed in L1 and the behaviors observed in L2 is non-obvious to the observer..." (Ronald et al., 1999). Here, we refer to the local level, of the implementation substrate, as L. The language of observation represents a global, or coarse-grain, level where emergent behaviour is observable that we refer to as the specification, S. After Shalizi (2001), we define emergence in information-theoretic terms, as the greater *predictive efficiency* of descriptions in S over those in L.

In natural complex systems, it is hard to define languages L and S, and to determine accurate mappings between them. Here, the complex emergent systems are elementary cellular automata (ECAs); their language is simple and well-defined, and thus mappings can be identified and analysed.

One perception of an emergent system is that its high level behaviour is independent of the low level behaviour. However, the emergent properties are actually a carefully chosen extract of the low-level behaviour. The observational discontinuity allows us to identify emergent behaviour. Elsewhere (Weeks et al., 2007), we show that coarse graining is a simple form of emergence. If we can state coarse-grained rules, then we can use the coarse level to predict behaviour. Because information is lost in the higher level we cannot predict behaviour correctly in all cases, but the rules should be able to predict some common futures. Here, we explore emergence through coarse graining ECAs and measurement of mutual information between levels.

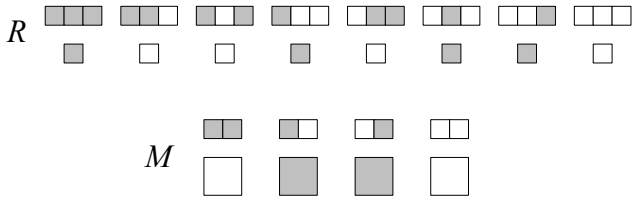


Figure 1: Rules and mappings. Shaded cells represent value 1. R shows how the "name" of an ECA rule is derived: each of the 8 possible ECA initial state is shown in bigendian order; below it, the next state of the central cell is shown; the rule is "named" by reading off the values of the next state. Here, R is the transition table for ECA rule 150 (10010110_2). M represents the coarse grain mapping 0110, with grain $g = 2$.

Coarse Graining ECAs

An ECA is a one-dimensional cellular automaton, with two states and a neighbourhood of three. There are 256 ECA rules, of which 88 are distinct (not just spatial reflections or 0-1 inversions). Rule sets are named by taking the decimal representation of the binary string that represents the outputs of the transition rules from all neighbourhood states taken in bigendian order (figure 1).

The coarse graining of ECAs was investigated by Israeli and Goldenfeld (2006). In a coarse graining at grain g, the values of a contiguous block of g cells at the fine level are projected, or mapped, to the value of a single cell at a coarse level (figure 1).

Israeli and Goldenfeld (2006) require their coarse grainings to be *total*, that is, to satisfy the commutativity condition that running the fine ECA for $n \times g$ time-steps then

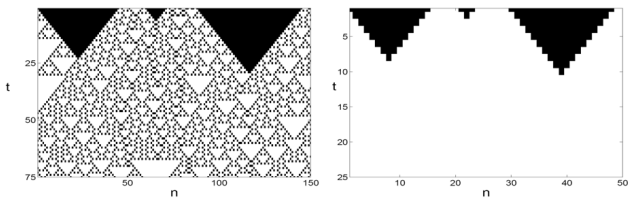


Figure 2: Space-time plot of 75 time-steps of rule 146, and its total coarse graining to rule 128, with grain 3, from Israeli and Goldenfeld (2006).

performing the mapping, gives the same result as performing the mapping, then running the coarse ECA for n time-steps. We relax this condition, to discover *partial* coarse grainings that, while not 'correct' in this commutative sense, nevertheless have good predictive properties (high mutual information). We do not consider 'trivial', and thus uninteresting, mappings, that map either to all zeros or to all ones.

Essentially, the coarse graining represents a system specification (language S) for the ECA at the fine level (language L). In one sense, the languages S and L differ only in the grain of the representation – both languages are the language of rules of ECAs. However, we could also take the language to be the specific ECA rule. In either case, ECA coarse graining reduces the language mapping to a tractable problem, and provides a starting point for exploring emergent behaviour.

Israeli and Goldenfeld (2006) show that almost all ECA rules can be coarse grained: their behaviour is mimicked exactly at a coarser grain by other ECA rules (figure 2). Note that the mapping is applied only in the initial state, to set up a correspondence between the fine and coarse grained initial state. Subsequently each ECA runs independently: the validity of the coarse graining ensures that the mapping always holds from then on.

Because of the exact mapping between the fine and coarse grained ECAs, the coarse grained ECA consistently predicts aspects of the state of the fine-grained ECA for any point in the future (since information is lost in the coarse graining, the prediction is not absolute, but it is consistent). For $g = 2$, running the coarse grained ECA requires only 25% of the calculations of the associated fine-grained ECA (assuming we calculate the next states naïvely).

As with any emergent system, one aim of coarse graining is to end up with a compact representation of the high-level behaviour of the underlying system (or some aspect of it). In the case of coarse graining, that compact representation takes the form of another ECA rule (but one that operates at a coarser grain). Clearly the high level model will predict only certain aspects of the system.

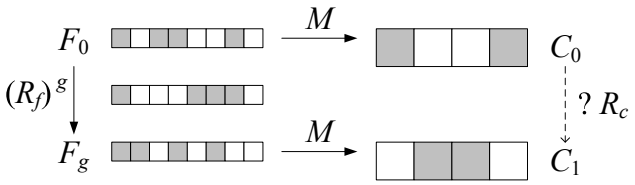


Figure 3: Steps in discovering a coarse graining, $g = 2$, illustrated for a 6-cell initial string. As in Israeli and Goldenfeld (2006), each non-overlapping block of g cells in a fine F state is mapped to one cell in the coarse C state. ECA rule R_f evolves through the three steps on the left, F_0 to F_g. State F_0 is mapped to state C_0 using the mapping 0110. The same mapping is used to produce C_g from F_g. When these steps are followed for a complete initial state (384 cells, see step 1), the candidate coarse grain rule, R_c can be read off, as in step 5, below.

Finding a Total Coarse Graining

Finding coarse grainings is a systematic process. Candidate mappings are successively proposed and applied; the result of each mapping is checked to determine whether the mapping generates a consistent coarse rule. We describe an algorithm that can be used to find all the total coarse grainings of an ECA rule R_f at grain g with non-trivial mapping M. Like Israeli and Goldenfeld (2006), we use the same grain g for the cells (spatial) and the time-steps (temporal). This maintains the speed of information propagation – for instance, if the spatial grain is less than the temporal grain, then information propagates too fast for ECA rule capture.

The application of steps 1 to 5 of the total coarse graining algorithm is illustrated in figure 3.

1. Construct the initial state for the fine-grained ECA, F_0. For a total coarse graining, we must guarantee that the coarse grained version of the initial state contains all eight (000, 001 ... 111) neighbourhood states. This will allow the coarse ECA rule to be read off in step 5. It is sufficient to include all possible states of $3 \times g$ cells (where 3 is the neighbourhood size of an ECA) in the fine-grained initial state, giving a string of length $2^6 \times 6 = 384$ for $g = 2$.

2. Run the fine-grained ECA for the equivalent of one coarse time step: g time-steps at the fine grain, resulting in fine-grained state F_g. We now have the underlying fine states for two successive time steps of the coarse CA.

3. Apply the mapping M to the initial state of the fine-grained ECA, F_0, to produce the initial state of a candidate coarse-grained ECA, C_0. Apply M to the final state, F_g, to produce the next state of a candidate coarse-grained ECA, C_1.

4. C_0 and C_1 are not necessarily related by an ECA rule. For each of the eight possible neighbourhood states σ_i,

check that every instance of σ_i in C_0 maps to the same value in C_1. (It is sufficient to consider only distinct triplets of cells in C_0: if these give consistent states in C_1, then all the overlapping neighbourhoods do so too, by construction of the initial state in step 1.)

5. If the coarse states C_0 and C_1 are consistent with successive states under some ECA rule, then 'read off' that rule R_c, by locating the eight neighbourhood states (111, 110 ...000) in C_0, and recording the values they map to in C_1 (as shown for R in figure 1). (Because of the consistency check, these values are unambiguous. Because of the construction of F_0 and the non-triviality of M, it is always possible to locate at least one instance of each of the eight neighbourhood states in C_0, so the rule is totally defined.)

The number of non-trivial mappings M is $2^{2^g} - 2$, which is low for small g (14 for $g = 2$), so an exhaustive search over mappings is efficient. (Even so, it is not guaranteed that a coarse graining of a rule at a particular granularity exists with any mapping.) At higher g, other factors render the discovery of coarse graining intractable, before the effect of the number of mappings becomes intractable.

It would also be possible to perform a coarse graining by stating the fine and coarse ECA rules and calculating a mapping, but this is a less efficient approach, with more consistency checks to perform.

For the 256 ECA rules, there are 182 non-trivial total coarse graining relationships at $g = 2$. For the 88 unique and non-trivial ECA rules, there are 35 non-trivial total coarse grainings.

Extending Coarse Graining to Partial Mappings

Information must be lost when coarse graining. Sometimes the fine detail of the original rule disappears: Israeli and Goldenfeld (2006) refer to this as a loss of irrelevant degrees of freedom (DOF). In other cases relevant DOF (Israeli and Goldenfeld, 2006) are lost, meaning that the information that is propagated by the fine CA cannot be modelled in all cases by its coarse counterpart. A total coarse graining precisely captures the relevant aspects of the underlying dynamics, losing only detail that is irrelevant at the coarse level. (Also, where detail that is relevant to propagation at the fine level is lost, it is unimportant, because propagation to the coarse state is unaffected.) In this sense, total coarse grainings are simply compressions of the fine ECA rule.

Israeli and Goldenfeld (2006)'s approach requires coarse grainings to be *total*; that is, enough information must be available in state C_0 for the coarse rule R_c to be read off unambiguously, with the consequence that the fine and coarse ECAs evolve consistently. If we relax the totality requirement, we can provide an initial fine-grained state that does not cover all possible input conditions, and deduce a set of *partial* coarse grainings. A partial coarse graining results in a coarse rule R_c that does not necessarily reflect the underlying rule R_f in all cases. Even so, this relaxation can allow the coarse grained rule to approximate more of the underlying behaviour than would otherwise be possible. An ideal partial coarse graining is one for which the initial conditions admit a broad range of the fine rule's behaviours, reflecting the features of most interest. An analogy can be drawn with physical emergent systems, where the emergent properties occur over a restricted set of all possible low-level states, such as a certain temperature range.

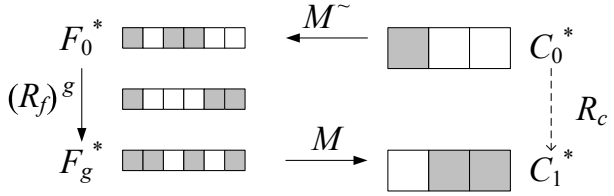


Figure 4: The extra steps involved in discovering a partial coarse graining (see text for details).

It may be thought that total coarse grainings are ideal, and we should aim to get partial coarse grainings as close to that as possible. While broadly true, being total is neither necessary nor sufficient for a good coarse graining – for the discovery of a coarse ECA rule that models the desired high level behaviour (we elaborate measures of goodness below). It is easy to see that some total coarse grainings capture uninteresting aspects of the underlying fine-grained rule; for example, even with non-trivial mappings, many rules coarse grain to rule 0 or rule 255; these are valid coarse grainings that convey no information about the underlying behaviour. In applying partial coarse grainings, we seek to find coarse rules that capture the maximum of useful (predictive) behaviours from the underlying rules, at the expense of allowing the coarse graining to make occasional mistakes.

Finding a Partial Coarse Graining

The approach for partial coarse graining follows almost the same steps is as the total coarse graining. However, when F_0 is constructed in step 1, it does not include all possible states of $3 \times g$ cells, and thus the initial state can be smaller than for total coarse graining. Consequently, to extract as much information as possible, steps 4 and 5 consider every (overlapping) combination of three cells in C_0.

Because the constraint on F_0 has been relaxed, C_0 may not include every possible rule case, meaning that the coarse rule cannot be read off unambiguously in step 5. To complete the rule set, we can add any rule case that gives a consistent result in the coarse graining. This can be derived as follows (see also figure 4):

5a. Create a coarse state C_0^*, comprising the neighbourhood

states missing from the coarse state C_0,

5b. Use the mapping in reverse, $M^{\sim}$, on C_0^* to create a corresponding F_0^*. This reverse mapping is a relation; any of its restrictions to a function can be used to create F_0^*.

5c. From F_0^*, run the fine ECA rule for g time-steps to create F_g^*.

5d. Apply M to F_g^* to produce C_1^*.

5e. Apply step 5 to C_0^* and C_1^*, thus reading off rule cases to total the coarse ECA rule set.

We have investigated various approaches to selecting an inverse mapping (step 5b). The most conservative results come from constructing all the inverse mappings, but accepting only those that are totally consistent. Since partial coarse graining is based on a proper subset of possible mapping relations, completions of the coarse rule set result in more valid coarse grainings of each rule than total coarse grainings. For the conservative approach to completion, and a grain $g = 2$, it is not unusual to get 50% more partial coarse grainings than total coarse grainings.

Exploring Partial Coarse Grainings

To assess the parameters of partial coarse graining more fully, we conduct a variety of experiments on ECA coarse grainings. For $g = 2$, we select different initial states, and find partial coarse grainings, using the conservative completion above. As the length of the initial state is increased, and the number of cases that it covers increases, there is a fall in the number of coarse grainings to valid ECA rules. In our experiments, a 6-bit input string (1.5% the length of the complete initial state used for total coarse graining) produced 57% more (322 : 182) coarse grainings than the total coarse graining under the same mappings. This is to be expected, since a short initial state is created by concatenation of only a small proportion of all possible states, so there are potentially many missing cases in the candidate coarse rule, with the potential for several different coarse rules to be derived from different completions of the rule set. However, the partial coarse grainings of a rule include all the rules to which there are total coarse grainings in each mapping, thus we can say that very short initial strings (and thus reduced calculation load) produce results that are consistent with those from the much longer initial state of a total coarse grainings.

There is a limit to the shortness of an initial state string. Here, this is set by features of the algorithm used for coarse graining. An initial state of fewer than 6 bits is not efficient, as it results in interference due to the wrapping of the state (periodic boundary conditions).

Our experiments also show that the form of a short initial state string appears to have a marked effect on the quality of partial coarse grainings obtained. The initial state string

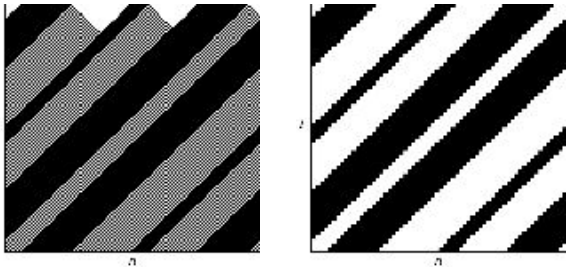


Figure 5: Space-time plot of 50 time-steps of rule 186, and its partial coarse graining to rule 170, with $g = 2$

101010, for instance, produces fewer partial coarse grainings to valid rules at $g = 2$ than the string 101101, but the partial coarse grainings from the string 101010 are judged to be of better quality as predictors of structure than those from string 101101. We cannot yet generalise from these results, not least because, at $g = 3$, these two initial strings produce very similar results. Furthermore, the initial state string 101010101101, made up of both these elements, produces good quality results at both $g = 2$ and $g = 3$. We are still investigating ways to determine what makes a good short initial state, and to determine how the quality of coarse grainings (or proportion of good coarse grainings) might relate to initial state.

We conclude that, although partial coarse grainings do not provide total accuracy in their predictions of fine grained behaviour, they can still provide 'good' descriptions. Figure 5 gives such an example: a partial coarse graining of rule 186 to rule 170. The coarse graining captures the significant, and persistent, "diagonal" structure of the fine rule. Note that under total coarse graining at $g = 2$, rule 186 coarse grains to rule 128, capturing only the transient "triangular" structure of the fine rule.

Intuitively, we can see that the capture of persistent structure by the partial coarse graining is more significant than the capture of the initial transient behaviour by the total coarse graining, but we would like an objective measure of this 'goodness'. In the next section, we show how to quantify what we mean by 'good' in information theoretic terms.

Quantifying Emergence

A challenge of emergent systems engineering is to be able to determine which low-level system gives a good emergent behaviour. In studying coarse graining, we want to be able to distinguish, as objectively as possible, good coarse grainings (that capture interesting properties of fine-grained ECAs) from other legitimate coarse grainings. Here, we use an information theoretic measure of mutual information.

Information theoretic measures of emergence have been proposed by Crutchfield (1994), and more recently by Shalizi (2001), Prokopenko et al. (2007), and others. It has been shown that the mutual information I between the implemen-

tation level L and the observational level S of a system is a measure of emergence. The intuition is that I measures the amount of information in a low level model (in a language L) that can be predicted by a higher level model (in language S). Modelling, or incremental system development, can be viewed as increasing the shared information between the specification and implementation. In the case of coarse graining, a good mapping can be thought of as a mapping that results in a high I between the fine-grained ECA (L) and the coarse ECA (S).

Mutual information can be calculated using a suitable entropy measure H of the systems, and can be expressed either in terms of the joint entropy $H(S, L)$ or, equivalently, in terms of the conditional entropy $H(S \mid L)$, or $H(L \mid S)$ of the systems.

$$\begin{aligned} I(S;L) &= H(S) + H(L) - H(S,L) \\ &= H(S) - H(S \mid L) \equiv H(L) - H(L \mid S) \end{aligned} \quad (1)$$

Intuitively, $I(S;L)$ is the correlation between the specified system (the coarse-grained ECA) and its implementation (the fine-grained ECA). In terms of conditional entropy, $H(S \mid L)$ is the information in the system specification that is not captured by its implementation, whilst $H(L \mid S)$ represents properties of L that do not explain, in information theoretic terms, the observed properties of S.

Calculating Mutual Information for ECAs

Mutual information requires a suitable entropy measure. We need an efficient entropy measure for ECAs, and, for calculating the I, we need to take account of the different spatial and temporal scales at the fine and coarse levels. Other research (for instance, (Zhao and Billings, 2006; Mori et al., 1998)) has used I in relation to ECAs, but with different purposes; their measures, though similar, do not directly adapt to our requirements in relation to spatial and temporal scales.

The key to a meaningful entropy measure is to measure over an appropriate scale, so that it identifies the structures that are important at that scale, without too much influence from order at other scales. This can be seen if we consider the entropy of a system such as a flocking simulation. We could measure the entropy at the level of each individual, in which case the entropy rises as individuals form flocks, because it is harder to characterise the behaviour of an individual in a flock (velocity, position, flock influences) than it is to characterise the behaviour of an isolated individual (velocity, position). Alternatively, we can attempt to measure the entropy of groups of individuals, in which case the entropy of a group that forms a flock is lower than that of a group that is incoherent, because it is easier to describe the flock's behaviour than that of an incoherent group of individuals.

Turning to ECAs, the rate of information transfer is limited by the neighbourhood, and a neighbourhood value represents one input in the initial state of an ECA; thus it seems reasonable to consider entropy in relation to the neighbourhood. Entropy is calculated from the probabilities of occurrence of each possible neighbourhood chunk value (000, 001, 010 ...), measured over many runs of each ECA. As we need to calculate I between two ECAs, we use the coarse grain for chunking, so entropy is measured over fixed chunks of three coarse cells. Therefore, the fine grain entropy, at $g = 2$, is calculated over chunks of six fine cells. Note that this means that fine-grained ECAs have a higher maximum entropy than coarse ECAs.

Accounting for the temporal difference between the coarse and fine grains requires identification of the generations of each ECA to be measured. After analysis of the practicalities of calculation and a number of experimental investigations, we determined that every coarse grained generation (row) should be measured, but only the corresponding fine-grained generation – for $g = 2$, that means every alternate fine-grained row. This means that the entropy of the fine grained rule overlooks some of the behaviour of the fine ECA, notably that over short time-scales. This would be a problem if we were interested only in entropy as a measure of complexity; the effects of the contraction are still being considered in our analyses.

The algorithm that we use to calculate entropy starts with the selection of a set of initial state strings on which to base the calculations. We calculate I for each coarse graining pair of ECA rules, and for each mapping that defines this coarse graining. Firstly, for one rule pair and one mapping, each test string is used as the initial state for the fine-grained ECA; the mapping is used to derive the equivalent coarse grained initial state, and the two ECAs are then run. We then select the rows representing the generations of interest, all coarse grained rows and the equivalent (gth) fine grained row, and apply the chunking equivalently to each pair of rows. We can then identify the number of times that each chunk occurs in each row.

These measurements are done for all the chosen initial state strings, and are then used to calculate the probabilities of the different chunks; the probabilities are used in the usual Shannon entropy calculations, yielding entropies $H(L)$ (from the fine grain), $H(S)$ (from the coarse grain) and $H(S, L)$. From these, I is calculated (equation 1).

We have completed these calculations for all $g = 2$ partial and total coarse grainings, and for a selection of coarse grainings at other granularities. We have also looked at I calculations based on different chunkings and time divisions, and, to date, this approach gives the most useful and cost-effective I measures.

Mutual Information and Coarse Grained ECAs

Mutual information can be a useful guide to the goodness of an emergent solution, and, here, an indicator of the quality of a coarse graining. A good mapping results in a high I

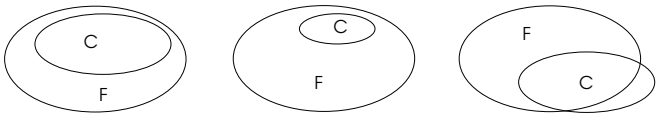


Figure 6: Representation of three different coarse grainings. The ovals indicate the amount of entropy: the large ovals represent the entropy in some fine grained ECA, F; the smaller ovals the entropy in the corresponding coarse grained ECA, C; the overlap represents the mutual information. The left figure shows a total coarse graining with high I; the middle a total coarse graining with low I; the right a partial coarse graining with high I.

between the ECAs. Here, we consider factors that influence I of mappings between ECA rules in coarse grainings.

I is high if the ECAs' behaviour is non-trivial (complex or chaotic rules) and tightly coupled (they mirror each other closely). The ECAs in a total coarse graining must always mirror each other's behaviour, so I is always maximal: the mutual information is exactly the entropy of the coarse ECA rule. However, if the behaviour of the ECA rules is trivial, and thus entropy is low, the maximal I for that coarse graining is lower than it is for "interesting" ECA rules. Furthermore, a non-trivial fine ECA rule (with high entropy) could be coarse grained to a simple rule (with low entropy), and the maximal I for the total coarse graining must then be low for that mapping. This is illustrated in figure 6. Total coarse grainings are accurate, but they are not necessarily good.

We have measured the I of total and partially coarse grained ECAs. As expected, many coarse grainings (both total and partial) have low mutual information – the coarse graining is not highlighting any significant structure from the fine grained ECA. This can occur even when the fine grained ECA rule produces non-trivial structure, with the extreme case being coarse grainings from a complex fine grained rule to a vacuous coarse grained rule such as rule 0 or rule 255. Furthermore, the additional coarse grainings that are valid under partial coarse graining include many vacuous cases with low I.

However, one result in particular is exciting in terms of the potential for using coarse graining to predict emergent structure – that high-I partial coarse grainings seem to predict high-I total coarse grainings at the current and higher grains. If we study all the total coarse grainings at a particular value of g, we find that some rules are coarse grained by many rules, and that the coarse grainings have notably high I – intuitively, this would suggest that there are structures in the fine-grained rule that are (a) non-trivial and (b) common to many rules at the coarse grain. When we consider the same results for partial coarse graining, firstly we observe that all the total coarse grainings are found by the partial approach, and that partial coarse grainings from rules that are associated with high-I total coarse grainings usually have higher I than the total coarse grainings – partial coarse grainings tend to predict more of the fine-grained structure than total coarse grainings. Next, we observe that the *additional* partial coarse grainings that have high I tend to be those that link rules with many total coarse grainings. It is also the case that these partial coarse grainings have higher I than partial coarse grainings that do not form links among rules with good total coarse grainings. Intuitively, where there is structure to exploit, the partial coarse grainings exploit more of the structure of the fine rule than the total coarse grainings, and do so preferentially where there are already good total coarse grainings.

Furthermore, where we observe high-I partial coarse grainings, these are good predictors of total coarse grainings at higher granularities, and in particular of non-trivial total coarse grainings (good partial coarse grainings at $g = 2$ predict total coarse grainings at higher g).

We note that we have measured I both very accurately and approximately. Tests with sufficient data to give good statistics (computationally rather expensive) were taken by averaging over 50 runs with different random initial states of 1000 characters and equal probabilities of the two cell values. An example of an approximate measurement is a single running of a 384-character initial state, with equal probabilities (for $g = 2$ this is the smallest complete state). For all mappings and all rules, the approximate I results are close to the results of the same extensive tests. Thus, we get good estimates of I, and hence of the 'goodness' of the coarse graining, more cheaply.

The Importance of the Mapping and Timing

We have noted elsewhere (Polack et al., 2005) that the mapping between the implementation and the observed system is essential to the construction of an emergent system. Without a good mapping it would be impossible to use a high level model (even an otherwise valuable one) to predict system behaviour. The work on ECA coarse grainings shows that, even where languages are essentially the same at two levels, a good mapping is hard to systematically derive, and is not always natural or obvious.

In fact, it is often the case that a fine rule R_f can be coarse grained to the *same* coarse rule R_c by different mappings M and M'. Figure 7 shows coarse grainings of fine rule 160 (the top row – the two columns represent two different initial states) to coarse rule 128 with two different mappings (second and third rows). Furthermore, some rules coarse grain to themselves with different mappings: for $g = 2$, rule 150 partially-coarse-grains to itself with 14 different mappings, of which six mappings (those with equal numbers of 0s and 1s) are also total coarse grainings. In general, mutual total coarse grainings with different mappings have similar Is, whilst for partial coarse grainings, some of the mappings have significantly lower Is than others; again rule 150 is

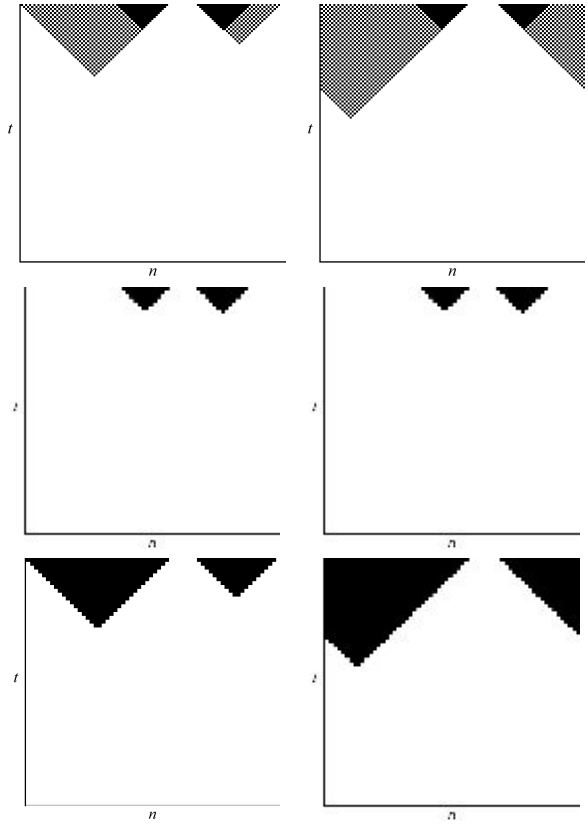


Figure 7: Space-time plots of coarse grainings of rule 160 to rule 128. The top diagrams are rule 160 with two different starting states. The next two diagrams are the results of coarse graining to rule 128 with the mapping 0001. The last two diagrams are the results of coarse graining to rule 128 with the mapping 0101. All figures represent 50 fine time-steps with $g = 2$

an extreme case, with total mappings to itself having reasonable predictive power (I values of about 3), whilst the partial coarse grainings have low predictive power (Is are around 0.6).

One important factor in prediction, that has a marked effect on measures of mutual information, is detection of transient or longer-term features of the ECA. In figure 7, we see that these ECAs become quiescent after about 20 fine time-steps; in the first coarse graining, quiescence is reached much sooner. Mutual information would be different if measured whilst transient behaviour dominates, compared to post-transient. The same is true for rules that have steady-state behaviours after transients die out, such as rules 186 and 170 (figure 5). Table 1 gives an example of mutual information and entropy measures for two partial coarse grainings of rule 162, measured at 4 and 10 coarse time-steps. Coarse graining to rule 128 (mapping 0001) has a similar short-term transient behaviour to that shown in figure 7, whereas the coarse graining to rule 170 (mapping 0111)

C	t	$H(162)$	$H(C)$	$I(C;162)$
170	4	4.354	2.386	1.879
	10	4.293	2.405	1.825
128	4	4.354	0.276	0.276
	10	4.293	0.093	0.093

Table 1: The effect of time-step on mutual information of coarse grainings of ECA rule 162, at $g = 2$. The start state has 1000 cells, and entropy is calculated over 50 runs.

picks up persistent diagonal features.

Coarse rule 170 shows only a slight (non-significant) change in mutual information between time-steps, which shows that rule 170 captures some persistent behaviour in rule 162. For coarse rule 128, however, there is a significant change in mutual information between time-steps 4 and 10. In most of the 50 runs, rule 128 reaches quiescence by coarse step 10 (no information, so no mutual information). Note that the mutual information data show that the partial coarse graining to rule 128 is also a total coarse graining – the mutual information is the same as the entropy of the coarse rule.

In most work on emergence, the focus is on the behaviours and languages at the high and low levels. The selection of mappings, and of total or partial coarse graining, is the subject of ongoing research; however, our work shows that the mapping between high and low levels is an important component of the emergence. This can be interpreted as the way the low level system is viewed through the mapping to form the high level (emergent) description.

Discussion

Total coarse graining of an ECA at $g = 2$ is efficient and fast. However, because of the number of calculations and checks to be performed, $g = 3$ is exponentially slower, and we found that $g = 5$ is beyond the limit of capability of a desktop computer. Partial coarse graining provides a tractable alternative, because high-I mappings can be determined from a small initial state. Furthermore, because the low-g partial coarse grainings are good predictors of higher-grain total coarse grainings, higher granularity searches can be focused rather than exhaustive.

One feature common in partial-only coarse grainings is the ability to predict beyond the transient behaviour of the rules. A total coarse graining often predicts early behaviour accurately but then dies out to quiescence – the total mapping over-constrains the ability to predict the long-term behaviour (as in figure 2). However, a partial coarse graining is less constrained, and is free to mismatch some early behaviour; thus it may be able to predict long-term behaviour (as in figure 5).

Despite the implication of the name "emergence", in information terms, emergence does not add anything; it re-

moves (or hides) aspects of the underlying system, to emphasise an apparently coherent core behaviour which we identify as a higher level phenomenon of interest. In effect, our high level view (constrained by language, time, etc.) blurs the underlying system so that only certain aspects of its behaviour are apparent. This is precisely equivalent to the DOF lost when coarse-graining an ECA.

Our work uses an information theoretic measure to compare the emergent quality of coarse grainings, but it does not explain how the measure can be maximised by selection of a good mapping rule, nor does it tell us how to relate information-theoretic measures to subjective qualities of desired (or deprecated) emergence.

Information theoretic approaches, such as comparison of I used here, have the potential to help determine a good solution, so long as we can map the desirable properties to information theoretic features. We have also found that a limited (and quick) I test approximates closely the results of an equivalent extensive (statistically valid) I test. This is important – if we consider that any valid coarse graining is identifying something emergent, then we have shown that finding emergence is easy; however it is the analysis of I that distinguishes useful (structure-finding) emergence from vacuous or trivial emergence.

We have found that a surprisingly small initial state string predicts results almost as well as using a complete initial state string, and moreover that the extra rules mapped by such a partial coarse graining are themselves useful indicators of interesting properties at coarse grains. If we think of the granularity as the scale of the emergent property, this is hinting at relationships across several scales, and hence across several levels of emergence.

We have demonstrated the importance of the mapping between the levels. Correctness is not necessarily an indication of goodness: having a mapping and a rule that works is not always enough. Indeed, it is often not enough (the result can have low I). Finding a valid coarse graining is much easier than finding a good one. It is, however, the extra 'goodness' properties that can be exploited to get robust implementations of S.

Our immediate future work is to investigate the relationship between partial fine state string and goodness of results (why do such short strings work so well here, and do they always?), and establish the relationship between 'good' coarse grainings at grain g and total coarse grainings at higher g. It is clear that different mappings and grainings focus attention on both different qualities of behaviour and different durations (transient, persistent) in the fine rule. It may ultimately be possible to tailor the selection according to what is of interest at the time.

Whilst coarse graining has a number of interesting properties, our main aim in this work is to gain understandings of these simple emergent systems that will allow us to progress two larger research goals: to understand and engineer emergence, and to find efficient solutions to difficult problems. Coarse graining is an efficient way to detect concealed structures, and thus might be applicable to guided search techniques. More importantly, our work shows that finding and exploiting mappings is likely to yield further progress, in guiding search for solutions.

Acknowledgements

Andrew Weeks is funded by a Microsoft Research European PhD Scholarship.

References

Crutchfield, J. P. (1994). The calculi of emergence: Computation, dynamics, and induction. *Physica D*, 75:11–54.

Israeli, N. and Goldenfeld, N. (2006). Coarse-graining of cellular automata, emergence, and the predictability of complex systems. *Phys. Rev. E*, 73:026203.

Mori, T., Kudo, K., Namagawa, Y., Nakamura, R., Yamakawa, O., Suzuki, H., and Uesugi, T. (1998). Edge of chaos in rule-changing cellular automata. *Physica D*, 116:275–282.

Polack, F., Stepney, S., Turner, H., Welch, P., and Barnes, F. (2005). An architecture for modelling emergence in CA-like systems. In *ECAL 2005*, volume 3630 of *LNAI*, pages 433–442. Springer.

Prokopenko, M., Boschetti, F., and Ryan, A. (2007). An information-theoretic primer on complexity, self-organisation and emergence. *to appear*. available at http://www.ict.csiro.au/staff/mikhail.prokopenko-/Publications/Agents/ITprimer-Nov-2007.pdf.

Ronald, E. M. A., Sipper, M., and Capcarrère, M. S. (1999). Design, observation, surprise! a test of emergence. *Artificial Life*, 5(3):225–239.

Shalizi, C. R. (2001). *Causal architecture, complexity and self-organization in time series and cellular automata*. PhD thesis, University of Wisconsin at Madison.

Weeks, A., Stepney, S., and Polack, F. (2007). Neutral emergence and coarse graining. In *ECAL 2007*, volume 4648 of *LNAI*, pages 1131–1140. Springer.

Zhao, Y. and Billings, S. A. (2006). Neighborhood detection using mutual information for the identification of cellular automata. *IEEE Transactions On Systems, Man, And Cybernetics Part B: Cybernetics*, 36(2):473–479.

Understanding robustness in Random Boolean Networks

Kai Willadsen[1,2], Jochen Triesch[1] and Janet Wiles[2,3]

[1]Frankfurt Institute for Advanced Studies, Johann Wolfgang Goethe University, 60438 Frankfurt am Main, Germany
[2]School of Information Technology and Electrical Engineering, University of Queensland, QLD 4072, Australia
[3]ARC Centre for Complex Systems, School of Information Technology and Electrical Engineering, University of Queensland, QLD 4072, Australia
willadsen@fias.uni-frankfurt.de

Abstract

Long used as a framework for abstract modelling of genetic regulatory networks, the Random Boolean Network model possesses interesting robustness-related behaviour. We introduce *coherency*, a new measure of robustness based on a system's state space, and defined as the probability of switching between attraction basins due to perturbation. We show that this measure has both upper and random-case bounds, and that these bounds are based on the size of individual attractor basins within the system. A mechanism for calculating these bounds is introduced, and the bounds are then used to define *structural coherency*, a measure of robustness attributable to system structure. Using these measures, we show that the decrease in coherency that occurs in the Random Boolean Network as its connectivity increases is related to a loss of structure in the system's state space.

Introduction

Since the introduction of the Random Boolean Network (RBN) model as a framework for modelling genetic regulatory networks (Kauffman, 1969), the robustness of the model has been an area of considerable research (Kauffman et al., 2003; Luque and Solé, 1998). This interest is due in part to the importance of understanding how robustness emerges in regulatory systems, and in part to the general applicability of the RBN model as a basis for understanding the dynamics of complex systems with epistatic interactions. Studies of the spontaneous emergence of robustness in the model have led to a greater understanding of the way in which robustness can be maintained in complex systems, as well as a better comprehension of what it means for a system to be stable.

Informally, robustness can be thought of as a system's ability to function normally under external perturbations. The investigation of robustness in RBNs generally focuses on the dependence between robustness and network connectivity. This focus sources from the discovery that the robustness of a RBN undergoes a phase transition around an average network connectivity of two ($K = 2$) (Kauffman, 1969). Later supported by theoretically based approaches (Derrida and Pomeau, 1986), this property demonstrates that robustness in large networks of interacting elements—such as genetic regulatory networks—can result from simple parameterisation, rather than being a property which must be designed or evolved. In addition, the connectivities at which RBNs display interesting and robust behaviour closely parallel the connectivities found in real-world genetic regulatory systems (Kauffman, 1969; Aldana, 2003).

A mix of theoretical (Derrida and Pomeau, 1986; Derrida and Flyvbjerg, 1987) and simulation-based (Kauffman, 1969; Aldana, 2003; Bastolla and Parisi, 1997) approaches have previously been used to understand the behaviour of robustness in RBNs. The most common technique used in theoretical analysis of discrete dynamic systems such as RBNs is the annealed approximation model (Derrida and Pomeau, 1986) which provides an analytically tractable approximation of RBN behaviour. This framework provides a useful theoretical prediction of the behaviour of infinite ensembles of networks under certain restrictive assumptions. However, this approach cannot be used to investigate single instantiations of the RBN model, or functional models of real-world systems. In contrast to the theoretical approach, simulation-based approaches are based upon the collection of metrics from multiple individual systems (Bastolla and Parisi, 1997; Aldana, 2003; Wuensche, 1998). As such, these approaches can be used to investigate robustness of individual networks in order to provide information about the range of behaviours observed under different circumstances, rather than just behavioural averages. So far, however, both theoretical and simulation-based approaches have generally focused on characterising the properties associated with robustness in the system (such as the areas of parameter space in which robustness is generally found), without focusing on understanding what system-level properties bring about the occurrence of robustness or lack thereof.

In this study, we define a new robustness metric we call *coherency*, which is based on the full enumeration of a system's state space. We use a combination of simulation-based and theoretical approaches to identify the relationship between the size of a system's basins of attraction and system coherency, as part of understanding the way in which robust-

ness arises in the RBN model. In addition, the formulation of coherency as a measurable property of individual systems means that coherency may be useful in characterising robustness in existing models of real-world systems.

Random Boolean Networks and state spaces

A RBN consists of N nodes or elements, each of which has exactly K incoming connections from other network nodes, and an average of K outgoing connections to other nodes in the network. Each node n in the network has a Boolean value σ_n which changes over time depending on a random Boolean function f_n of the inputs to the node (i.e., $\sigma_n(t+1) = f_n(\sigma_{n_1}(t), \ldots, \sigma_{n_K}(t))$, where σ_{n_i} indicates the ith input to node n). The Boolean function associated with an individual node is independently randomly generated for each node, and stays constant over the lifetime of a system.

RBNs are *discrete dynamic systems*: discrete because each node in a system has a discrete value; and dynamic because the value of individual nodes changes over time. Boolean-valued nodes imply that the system has a finite number of states (2^N). In addition, the unchanging (or quenched) nature of the random Boolean functions $f_1 \ldots f_N$ means that the model is also deterministic.

One way of conceptualising the dynamics of a system is through the system's *state space*. State space consists of the set of all possible states of the system, and the set of state transitions (system dynamics) defined by the collective action of the random Boolean functions $f_1 \ldots f_N$. Graphically, states in the system can be represented as nodes in a graph, with edges between nodes representing state transitions (see Figure 1). In a RBN, this state space will have points or limit-cycles (here referred to collectively as *attractors*) consisting of one or more nodes which define the dynamic behaviour of the system in the time limit; any system running indefinitely must end up in an attractor. The *length* of an attractor is the number of nodes in the attractor's cycle. In addition, associated with each attractor is a set of states which lead to that attractor, termed the *basin of attraction*. The size of a basin of attraction is the number of states in the basin; alternatively, size can be expressed as basin weight, being the proportion of state space occupied by the basin's states. For example, Figure 1 shows a RBN state space ($N = 8$, $K = 6$) with three attractors of length 3, 7 and 19 and three corresponding basins of attraction with size 20 (weight $\frac{20}{256}$), 94 (weight $\frac{94}{256}$) and 142 (weight $\frac{142}{256}$).

Attractors and basins of attraction are important concepts in defining the robustness of individual RBNs, as they represent the deterministic dynamics of a system leading to its steady state.

Defining coherency

There are two commonly used approaches to measuring robustness in RBNs: annealed-approximation methods; and simulation-based perturbation methods. The annealed approximation framework, which is applied in the context of a statistical ensemble of systems, uses a theoretical method to determine the probability of state perturbations propagating through a system (Derrida and Pomeau, 1986). In this framework, robustness is defined as the probability that a perturbation to the activation of a node will, over time, spread to affect other nodes in the network. A stable network is one in which a perturbation to the state of the network at time t will likely result in the same network state at time $t+\Delta t$ (i.e., the perturbation dies out). If the states of the perturbed and unperturbed network differ at time $t+\Delta t$, then the network is considered unstable (i.e., the perturbation has spread). The second method for measuring robustness is a simulation-based random-sampling approach that is best described as 'perturb and iterate'. In this approach, robustness is defined as the probability of a single-element perturbation to a system state s resulting in a system state s' that is in the same basin of attraction as the original state (Aldana, 2003; Geard et al., 2005; Reil, 1999). These two measurements of robustness are closely related, both defining robustness as a lack of change of network expression in the time limit. However, both of these measurements have different shortcomings. The annealed approximation approach cannot be used to describe individual RBNs, meaning that it is not useful for analysing models of specific real-world systems. In contrast, the 'perturb and iterate' approach is based on random sampling, which only provides an incomplete—and possibly inaccurate—picture of a system's robustness.

In this study, we define a measure closely related to the above approaches, which we term *coherency*, based on the full-enumeration of a system (i.e., generating every possible

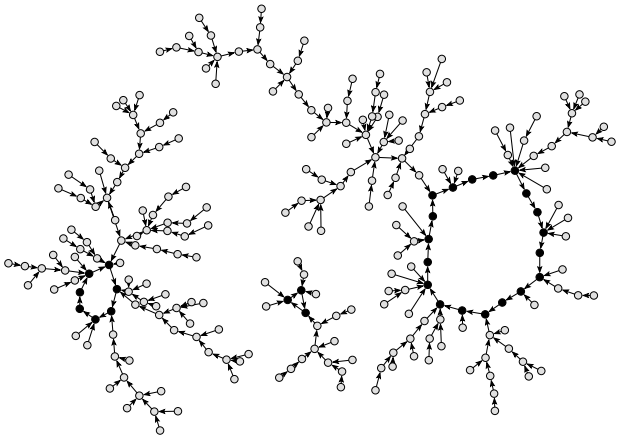


Figure 1: An example RBN state space graph with $N = 8$ and $K = 6$ containing three attractor basins. Nodes in the graph represent states of the network, with directed edges representing the state transitions determined by the Boolean functions of the network.

system state and identifying each state's successor). The coherency of a system is defined to be the probability that a single-element perturbation to any state of the system does not change the basin of attraction of the state. As a codification of robustness, coherency characterises a system as robust if a small perturbation to the system state is unlikely to affect the long-term behaviour of the system.

Coherency can be most simply defined in terms of an individual state. For a state s with neighbouring states nh (where neighbouring states are states that differ by only a single element; i.e., states having a Hamming distance to s of one), the coherency of that state ψ_s is the percentage of states nh that are in the same basin of attraction as s. This definition can be readily extended to arbitrary collections of states, the most interesting of which are whole systems, and attractor basins.

For a system S we define the system coherency ψ_S,

$$\psi_S = \frac{1}{|S|} \sum_{s \in S} \psi_s \ , \tag{1}$$

where $s \in S$ are individual states of the system, and ψ_s are coherencies of individual states. As the coherency of any group of states is the average of the coherency of each individual state, we can also formulate the coherency of a system as the average of the coherencies of its basins of attraction, weighted by the size of each basin,

$$\psi_S = \sum_{b \in B} \psi_b \frac{|b|}{2^N} \ , \tag{2}$$

where B is the set of attractor basins of the system, and $|b|/2^N$ is the weight of an individual basin.

The computational complexity of measuring coherency is prohibitive, being $O(N2^N)$ in both time and space, but is nevertheless feasible for small systems ($N \leq 25$); even at such a restricted system size, this measure is applicable to both abstract systems such as RBNs and interesting models of real-world systems (e.g., Albert and Othmer, 2003; Li et al., 2004; Mendoza and Alvarez-Buylla, 1998). In exchange for this computational complexity, coherency avoids the drawbacks associated with existing robustness measures for RBN systems: unlike the annealed approximation approach, coherency measures specific individual systems; and unlike the 'perturb and iterate' approach, it tests all possible single-element perturbations for all system states. In addition, the definition of coherency is general in that it can be applied to any collection of states simply by considering only a subset of system states. In other words, coherency, as defined here, is a comprehensive measurement of the robustness of an individual system or parts of the system.

Basin coherency, basin size and network connectivity

In order to understand this new measure of robustness, and to see what information it may provide, the standard RBN model was simulated ($N = 16$, $K = 1, 2, 3, 4, 8, 16$; 1000 trials). For each trial, the system coherency ψ_S and the average of individual basin coherencies $\bar{\psi}_B = \frac{1}{|B|} \sum_{b \in B} \psi_b$ were recorded and averaged over each parameter combination.

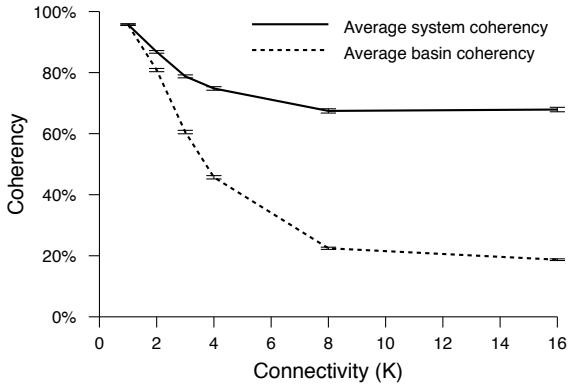


Figure 2: Coherency vs. connectivity for whole state spaces and individual basins of attraction in RBN systems ($N = 16$; error bars show standard error). System coherency (ψ_S) and basin coherency ($\bar{\psi}_B$) both fall as connectivity increases, with basin coherency falling more rapidly.

The coherency of whole systems is seen to decrease as connectivity increases (see Figure 2), which agrees with accepted knowledge about the robustness of RBNs (Aldana, 2003). However, this result does not provide any insights as to the mechanisms causing coherency in systems with low connectivity, or those underlying the loss of coherency in higher-connectivity systems. For these insights we must consider the coherency of sub-structures of the system's state space: attractor basins. The average coherency of basins of attraction in a system is also seen to decrease as K increases (see Figure 2), but this decrease is far more rapid than the corresponding decrease in the overall system coherency. These results can be explained in terms of the relationship expressed by (2), by observing that it is the interaction between coherency and weight of individual basins of attraction that is crucial in determining system coherency. Given this relationship, the difference between the decrease in basin coherency and system coherency suggests that not every basin of attraction contributes equally to overall system coherency.

In order to understand the transition between stable and chaotic regimes in the RBN model, as characterised by a changing system coherency, we need to understand the relationship between coherency and size of attractor basins within a state space. This relationship may be characterised by comparing the size of each basin of attraction with the basin's coherency, and observing how these values change

with respect to the connectivity of the system, K. The relationship between coherency and basin size does exhibit a change as the value of K increases, from an apparently logarithmic relationship to a linear one (see Figure 3). Just as one notable characteristic of the data is the variance of this relationship with K, another pertinent characteristic is the apparent restrictions on the upper and lower values of basin coherency with respect to basin weight. Making use of known properties of the RBN model, we can investigate the observed upper and lower bounds on the relationship between coherency and basin size.

Random-case bounds on coherency

In the RBN model, it is expected that the lower bound on the observed coherency will occur when the system is most highly disordered[1], a condition that occurs at maximum network connectivity. This lower bound on the observed relationship between coherency and basin size, referred to here as the random-case bound, is a probabilistic bound that describes system behaviour over a statistical ensemble of systems. The random-case bound occurs as $K \to N$, and is a linear relationship between the weight and the coherency of the attractor basin. It has been shown that for $N \to \infty$, $K \to N$, RBNs correspond to the random map model (Derrida and Flyvbjerg, 1987). In this model, the system dynamics implement a random mapping between states in the system. That is, given a state s, the system's dynamics define a transition that will deterministically move the system to a new state, s'; in the random map model, the target state, s', is randomly assigned for all such transitions. It follows that since there is no correlation between a state and its successor, there is also no correlation between the states in an attractor basin, since the basin is defined by these transitions. If there is no correlation between states in a basin of attraction, then the expected coherency—the probability of the result of a single-point perturbation belonging to the same basin of attraction—is proportional to the weight of the basin of attraction. After accounting for the fact that a perturbation cannot result in the original state, we have,

$$\psi_{rand}(b) = \frac{|b| - 1}{2^N - 1} \,. \tag{3}$$

As the coherency of each individual state in a basin can be seen as a Bernoulli trial, basin coherencies from complete-enumeration simulations will be binomially distributed with,

$$p = \frac{|b| - 1}{2^N - 1} \,,$$
$$n = N|b| \,.$$

[1]While there is an actual lower bound on coherency defined by a parity function, this bound does not correspond to any known parameterisation of the RBN model.

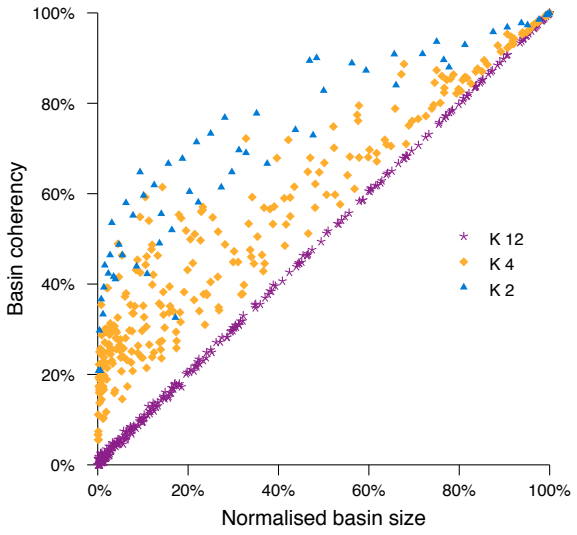


Figure 3: Basin weight vs. basin coherency for $N = 12$ and $K = 2, 4, 12$. The three connectivity levels demonstrate a clear separation. At low K, the data describes a curve that is significantly above the identity line, with the relationship tending toward linear as K increases.

As the $K \to N$ limit is the least stable point in the parameter space of a RBN (Bastolla and Parisi, 1997), (3) can be taken as a random-case probabilistic bound of the coherency of any RBN system, and a description of the lowest size-coherency pairs that are likely to be observed.

Upper bound on coherency

Determining the upper bound on the coherency of a basin of attraction is not readily amenable to an analytic solution. A simulation-based approach was therefore undertaken to investigate the way in which formation of highly stable basins of attraction occurred in RBN systems. Since $K = 1$ is the most stable point in the RBN model (Flyvbjerg and Kjær, 1988), an investigation of the structure of attractor basins in these systems was undertaken.

The structure of the state spaces investigated was analysed using schemata to represent the states present in a basin of attraction. The state of a Boolean network model can be represented by a vector of Boolean elements (e.g., [1 0 0 1 0] would be a state in an $N = 5$ system); using a ternary representation—a schema—with elements '0', '1' and a wild-card, '⋆', makes it possible to represent many states in a compact fashion (Bagley and Glass, 1996). It was discovered that representing basins of attraction in this manner provided insights into the formation of highly coherent basins of attraction in low K systems.

Highly coherent basins of attraction in these systems could generally be described by a very simple schema. For

example, a basin that covered exactly half of the state space of a five-element system could be described by a schema such as $[\star\star 0 \star\star]$. That is, the basin would be formed by making membership of the basin contingent on the value of a single element in the system. Similarly, a basin covering one quarter of the state space would be contingent on the value of two elements, and so on. In cases where a basin was of a size that could not be described by a single schema, basins were generally found to be composed of simple, non-overlapping schemata whose sizes formed a minimum binary partition of the basin size. For example, a basin of size 715 would likely be composed of six non-overlapping basins with sizes 512, 128, 64, 8, 2 and 1.

Given these observations, we conjecture that the upper bound on the robustness of a basin of attraction exists when the basin is composed of a set of non-overlapping schemata, I, with sizes corresponding to a minimum binary partition of the basin size. In other words, the set I satisfies the equation,

$$|b| = \sum_{i \in I} |i| \ , \tag{4}$$

and is the smallest set to do so. The constraints placed upon this set (i.e., that each number in the set is unique, that each number is a power of two, and that the schemata are non-overlapping) mean that it is possible to calculate an optimal coherency from I alone (see appendix). The maximum coherency of a basin ψ_{max} is given by,

$$\psi_{max}(I, N) = \sum_{i \in I} \frac{|i|}{N|b|} \left(\log_2 |i| + \sum_{j \in (I \setminus i)} \frac{1}{\lceil |i| \div |j| \rceil} \right) \ , \tag{5}$$

where I is the set of schema sizes described above, and N is the size of the network. The sum over $I \setminus i$ represents the coherency between the schema with size i and the other schemata in I. Since I is determined solely from the basin size, and is unique for any given value thereof, our conjectured upper bound (referred to below simply as the upper bound) can be calculated from the size of the network, N, and the size of the basin, $|b|$.

The conjectured upper and random-case bounds outline the area in which relationships between coherency and basin size are expected to fall within the gamut of RBN systems (see Figure 4). Like the random-case bound described above, this upper bound is not a theoretical limit describing the maximum possible coherency of a basin of attraction of the given size, but rather a bound on the expected coherency, based upon an analysis of observed system behaviour. Nevertheless, numerical simulations indicate that this bound is a useful approximation (see following section).

An interesting property of the upper bound is its change with respect to N. The coherency of a basin of attraction b of size $|b|$ can be approximated by,

$$\frac{\log_2 |b|}{N} \ . \tag{6}$$

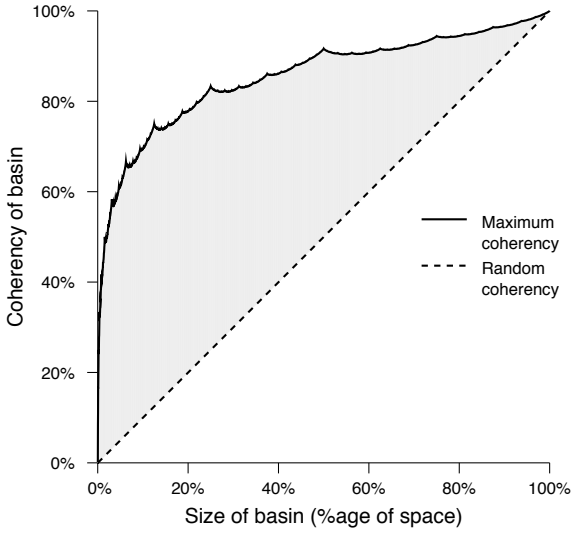


Figure 4: The conjectured upper (solid) and random-case (dashed) bounds on the expected coherency of a basin of attraction for an $N = 12$ system.

As N increases, the maximum possible coherency for a given basin weight (%age size) increases. In other words, a basin that occupies 20% of an $N = 50$ state space has a higher maximum coherency than a basin occupying 20% of an $N = 10$ state space (see Figure 5). This increase is most notable for relatively small basin sizes.

Testing bounds

The upper and random-case bounds developed above (see Figure 4) appear to provide useful and accurate bounds on the observable coherencies of basins of attraction (see Figure 6). As a simple measure of the accuracy of the predictions, we can investigate the frequency with which the bounds are violated (see Table 1). In all simulations, the upper bound described a hard upper limit on the relationship between coherency and basin size. While attractor basins with coherency-size pairs below the random-case bound were found, this is to be expected as described above.

Attributing robustness to structure

The upper and random-case bounds on expected coherency relate basin robustness to basin structure; unstructured basins have notably lower robustness than structured basins, given equality of size. However, since we can determine the maximum and random-case robustness for any given basin size, we can also quantify the degree to which an attractor basin's robustness depends on its structure rather than its size. By comparing the actual coherency of a basin to the maximum and random-case coherency, we can express basin

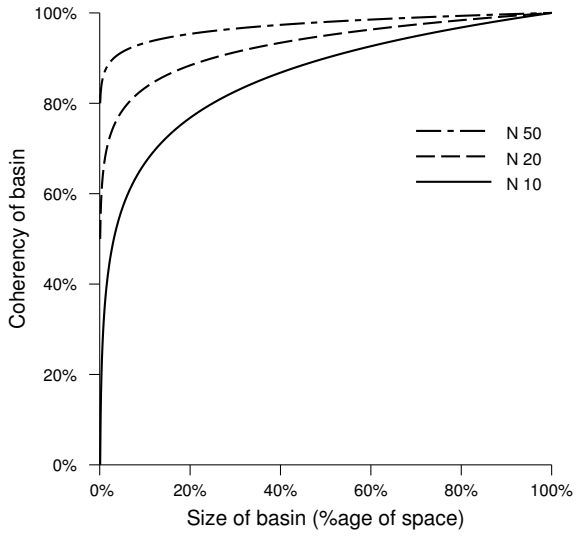


Figure 5: The change in the maximum possible coherency of a basin of attraction of a given normalised size with respect to N. As N increases, the coherency of an absolute basin size remains the same, but the coherency of the relative basin size increases. As N becomes large, even basins that are very small relative to the state space may be highly coherent.

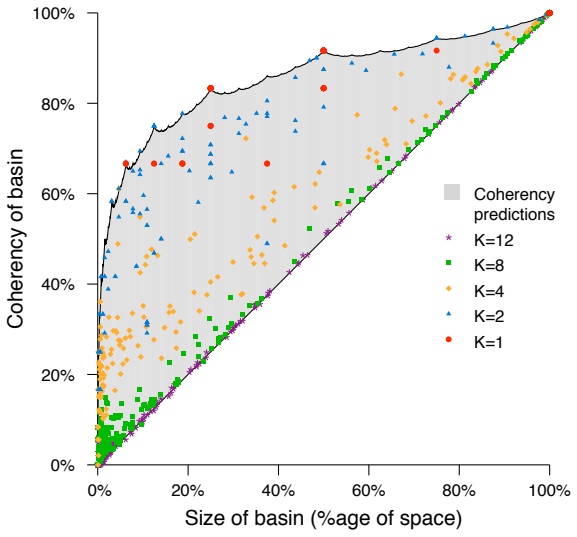


Figure 6: Observed coherencies for RBN systems ($N = 12$; $K = 1, 2, 4, 8, 12$; 50 trials). The various observations fall largely within the shaded area, which represents the expected boundaries; due to the probabilistic nature of the random-case bound, some data points are found below the expected boundary.

robustness as a proportion of the difference between size-dependent and maximum robustness. This measure, which we term *structural coherency*, can be expressed as ψ_{struct},

$$\psi_{struct}(b) = \frac{\psi(b) - \psi_{rand(b)}}{\psi_{max}(b) - \psi_{rand}(b)} \; . \qquad (7)$$

Structural coherency has no meaningful interpretation for basin sizes with the same maximum and random-case coherencies: 1, $2^N - 1$, and 2^N.

Measuring the structural coherency of RBNs with varying connectivity, we obtain a clear striated pattern showing structural coherency progressing from 100% to 0% as K increases from 1 to N (see Figure 7). In contrast to normal coherency, structural coherency seems to be approximately constant over basin size, while still varying as expected over K. This measurement shows that even in small basins of attraction, the basin structure within low K systems results in attractor basins with high relative degrees of coherency.

K	Lower	Within	Higher
1	0.0%	100.0%	0.0%
2	0.0%	100.0%	0.0%
4	0.6%	99.4%	0.0%
8	10.5%	89.5%	0.0%
12	43.6%	56.4%	0.0%

Table 1: Percentage of attractors that were below, within or above the bounds ($N = 12$; $K = 1, 2, 4, 8, 12$; 50 trials). The frequency of overstepping the random-case bound is consistent with the probabilistic nature of that bound.

Conclusions

Robustness or lack thereof in individual RBNs can be analysed in order to better understand when and how robustness arises. We have demonstrated a combination of simulation-based and simple theoretical techniques to provide information about the relationship between the robustness of a system and the size of individual attractor basins within that system.

It was suggested that basins of attraction in a system have both upper and random-case bounds on their coherency that depend only on the size of the basin of attraction and the size of the system as a whole. In isolating the maximum and random-case coherency values for a particular system, the concept of structural coherency was established to describe the proportion of basin robustness attributable to the structured organisation of a state space. These measures may subsequently be used in order to try and identify causal relationships between robustness and other system properties, such as network architecture or environmental influences (Willadsen and Wiles, 2007).

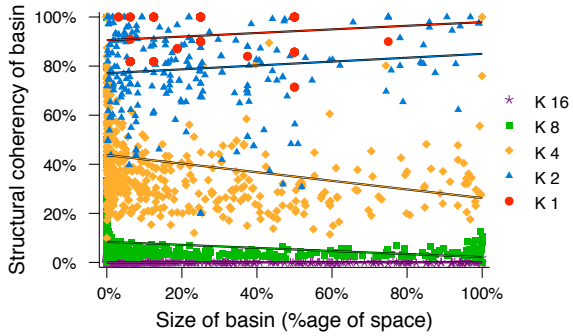


Figure 7: Observed structural coherencies vs. basin size ($N = 16$; $K = 1, 2, 4, 8, 16$; best-fit lines indicated). Structural coherency appears to be approximately constant over basin size, but varying over K.

The network connectivity of the RBN model, K, was shown to affect the coherency of a system by changing both the size and coherency of basins of attraction. The relationship between basin coherency and basin size for basins of attraction in $K = 1$ and $K = N$ systems demonstrated strong agreement with the upper and random-case bounds respectively. The structural coherency of attractor basins was shown to be indirectly proportional to the network connectivity, demonstrating that chaotic (high connectivity) networks have low robustness because their state space becomes disorganised. A notable difference in the results provided by coherency and other robustness measures (e.g., Kauffman, 1969; Derrida and Pomeau, 1986) is that the coherency results show no special phase-transition behaviour at $K = 2$.

From these results, we believe that it is possible to understand how, why and under what conditions robustness occurs in discrete dynamic systems such as Random Boolean Networks, and in discrete dynamic models of real world systems. While the methods presented here do not scale well (i.e., $O(N2^N)$), several pre-existing interesting, small-scale models of real-world biological systems (e.g., Albert and Othmer, 2003; Li et al., 2004; Mendoza and Alvarez-Buylla, 1998) are amenable to such analysis (Willadsen and Wiles, 2007). Understanding such model systems may eventually help in the analysis of robustness in real-world systems, such as developmental robustness and homeostasis in genetic regulatory networks.

Acknowledgements

The authors would like to acknowledge Nic Geard, Jennifer Hallinan, Ben Skellett and James Watson for useful discussion. The Pajek software package was used in the production of graph visualisations.

References

Albert, R. and Othmer, H. G. (2003). The topology of the regulatory interactions predicts the expression pattern of the segment polarity genes in *Drosophila melanogaster*. *Journal of Theoretical Biology*, 223(1):1–18.

Aldana, M. (2003). Boolean dynamics of networks with scale-free topology. *Physica D*, 185:45–66.

Bagley, R. J. and Glass, L. (1996). Counting and classifying attractors in high dimensional dynamical systems. *Journal of Theoretical Biology*, 183:269–284.

Bastolla, U. and Parisi, G. (1997). A numerical study of the critical line of Kauffman networks. *Journal of Theoretical Biology*, 187:117–133.

Derrida, B. and Flyvbjerg, H. (1987). The random map model: a disorded model with deterministic dynamics. *Journal de Physique*, 48:971–978.

Derrida, B. and Pomeau, Y. (1986). Random networks of automata: A simple annealed approximation. *Europhysics Letters*, 1(2):45–49.

Flyvbjerg, H. and Kjær, N. J. (1988). Exact solution of Kauffman's model with connectivity one. *Journal of Physics A: Mathematical and General*, 21(7):1695–1718.

Geard, N., Willadsen, K., and Wiles, J. (2005). Perturbation analysis: A complex systems pattern. In Abbass, H., Bossamaier, T., and Wiles, J., editors, *Recent Advances in Artificial Life*, volume 3 of *Advances in Natural Computation*, pages 69–84. World Scientic Publishing, Singapore.

Kauffman, S., Peterson, C., Samuelsson, B., and Troein, C. (2003). Random Boolean network models and the yeast transcriptional network. *PNAS*, 100(25):14796–14799.

Kauffman, S. A. (1969). Metabolic stability and epigenesis in randomly constructed genetic nets. *Journal of Theoretical Biology*, 22:437–467.

Li, F., Long, T., Lu, Y., Ouyang, Q., and Tang, C. (2004). The yeast cell-cycle network is robustly designed. *Proceedings of the National Academy of Sciences*, 101(14):4781–4786.

Luque, B. and Solé, R. V. (1998). Stable core and chaos control in random Boolean networks. *Journal of Physics A*, 31(6):1533–1537.

Mendoza, L. and Alvarez-Buylla, E. R. (1998). Dynamics of the genetic regulatory network for *Arabidopsis thaliana* flower morphogenesis. *Journal of Theoretical Biology*, 193(2):307–319.

Reil, T. (1999). Dynamics of gene expression in an artificial genome - implications for biological and artificial ontogeny. In Floreano, D., Nicoud, J.-D., and Mondada, F., editors, *Fifth European Conference on Artificial Life*.

Willadsen, K. and Wiles, J. (2007). Robustness and state-space structure of Boolean gene regulatory models. *Journal of Theoretical Biology*, 249:749–765.

Wuensche, A. (1998). Discrete dynamical networks and their attractor basins. In Standish, R., Henry, B., Watt, S., Marks, R., Stocker, R., Green, D., Keen, S., and Bossomaier, T., editors, *Complex Systems '98*.

Appendix: Calculating conjectured upper bound coherency

We have conjectured that the upper bound on basin robustness is characterised by a basin being composed of a set of schemata I, such that the schemata $i \in I$ are non-overlapping, and have sizes that correspond to a minimum binary partition of the basin size. Here we calculate the relationship between basin size and basin coherency under these conditions.

Each schema comprising a basin of attraction makes a coherency contribution proportional to the size of the schema, in the same way that basins of attraction make coherency contributions to a system. The coherency of a single schema is equal to the probability of a perturbation resulting in a state that is in the schema, $coh(i,i)$, added to the probability of a perturbation resulting in a state that is outside the schema, but is in another schema in the same basin of attraction $coh(i, I \setminus i)$. Therefore,

$$\psi_{max}(I) = \sum_{i \in I} \frac{|i|}{|b|} \Big(coh(i,i) + coh(i, I \setminus i) \Big) \ . \tag{8}$$

The coherency of the schema i with itself is simply the size of the schema, normalised with respect to the system size. Denoting the self-coherency as $coh(i,i)$,

$$coh(i,i) = \frac{\log_2 |i|}{N} \ , \tag{9}$$

where $|i|$ is the schema size, being 2^x where x is the number of wild-card elements in i.

The coherency of the schema i with all other schemata in the basin is the sum of the coherency of schema i with each individual schema $j \in I \setminus i$. Denoting this coherency as $coh(i, I \setminus i)$, we have,

$$coh(i, I \setminus i) = \sum_{j \in (I \setminus i)} coh(i,j) \ . \tag{10}$$

Determining coherency between two individual schemata can be accomplished by some simple logic. Given that the conditions for maximum coherency state that the collection of schemata is minimal, and that schemata are non-overlapping, we can divide the coherency calculation into two disjoint conditions: $|i| > |j|$ and $|i| < |j|$. When $|i| > |j|$, the probability that a perturbation of i will lead to j is dependent on the relative sizes of i and j. The total number of possible perturbations from schema i is $N|i|$, and the number of target states is $|j|$, implying $coh(i,j) = \frac{|j|}{N|i|}$. When $|i| < |j|$, we rely upon the hierarchical nature of the schemata; membership of each schema is determined solely by the value of a single element. Therefore, changing from a smaller schema, i, into a larger schema, j, implies perturbing the single appropriate element in a system of N elements, thus $coh(i,j) = \frac{1}{N}$. Combined, we can write,

$$coh(i,j) = \begin{cases} \frac{1}{N} & \text{if } |i| < |j| \\ \frac{|j|}{N|i|} & \text{if } |i| > |j| \end{cases} \ , \tag{11}$$

or alternatively,

$$coh(i,j) = \frac{1}{N \lceil |i| \div |j| \rceil} \ . \tag{12}$$

Combining (8) and (12) yields an equation for the maximum expected coherency of a system with N network nodes, and attractor basin sizes I,

$$\psi_{max}(I,N) = \sum_{i \in I} \frac{|i|}{N|b|} \left(\log_2 |i| + \sum_{j \in (I \setminus i)} \frac{1}{\lceil |i| \div |j| \rceil} \right) \ . \tag{13}$$

Evolving Referential Communication in Embodied Dynamical Agents

Paul L. Williams[1], Randall D. Beer[1,2,3] and Michael Gasser[1,2]

[1]Cognitive Science Program, [2]Dept. of Computer Science, [3]Dept. of Informatics
Indiana University, Bloomington, IN 47406 USA
plw@indiana.edu

Abstract

This paper presents results from three experiments which investigate the evolution of referential communication in embodied dynamical agents. Agents, interacting with only simple sensors and motors, are evolved in a task which requires one agent to communicate the locations of spatially distant targets to another agent. The results from these experiments demonstrate a variety of successful communication strategies, providing a first step towards understanding the emergence of referential communication in terms of coordinated behavioral interactions.

Introduction

Communication is traditionally viewed as the use of signals to transmit information (Hauser, 1997; Seyfarth and Cheney, 2003; Smith and Harper, 1995). We refer to this view of communication as the IT view, for *information transmission*. Numerous models of emergent communication adopt this view as their starting point (Cangelosi and Parisi, 1998; MacLennan and Burghardt, 1993; Steels, 2003). Agents are provided with signalling mechanisms and informational content (or "meanings"), and through some adaptive process they establish shared associations between signals and meanings. However, the IT view of communication is not uncontroversial (Di Paolo, 1997, 1998), providing motivation to study emergent communication without preconceived notions of signals and information transmission. Moreover, even if the IT view is accepted, models that begin with established signalling mechanisms cannot be used to address important questions of how signals may arise from initially non-communicative behaviors.

An alternative view of communication comes from autopoetic theory (Maturana, 1978; Maturana and Varela, 1980), with similar ideas expressed by researchers in cybernetics, psychology, and a wide range of other disciplines (see Di Paolo (1997) for an extended discussion). On this view, communication occurs whenever the behavior of one agent shapes the future behavior of another agent. Thus, communication is taken to refer to all kinds of socially coordinated behaviors. We refer to this view of communication as the CB view, for *coordinated behavior*. Importantly, the CB view does not assume the existence of signals or information transmission as fundamental aspects of communication. Rather, if anything, these ideas are left to the analysis of communication by scientific observers. The essential elements of communication are the structured interactions that take place between agents in a shared domain.

Several models have explored the emergence of communicaton from a CB perspective (Baldassarre et al., 2003; Di Paolo, 2000; Iizuka and Ikegami, 2003; Nolfi, 2005). In these models, agents are typically equipped with dedicated channels to use for communication. Through some adaptive process, the agents develop the ability to use these channels to signal each other, resulting in the improved coordination of their behaviors. Thus, since these models begin without pre-specified signals, they provide compelling demonstrations of how initially non-communicative behaviors can adapt to serve communicative functions. This is particularly true when agents are equipped with only sensors and actuators, and without dedicated communication channels (Quinn, 2001; Quinn et al., 2003). In this case, simulations can provide additional insights into the interplay between communicative and non-communicative behaviors, and explore how signals may emerge from behaviors that initially evolved for other purposes.

Models that study communication from the CB perspective have typically focused on certain kinds of tasks. For instance, common tasks are those which require agents to develop signals for the dynamic assignment of roles (e.g. "leader" and "follower") in some situation. In contrast, models of communication from an IT perspective have often studied tasks that are referential in nature, where agents must develop signals that refer or "point to" states of affairs that are removed in space and/or time. Referential tasks are certainly of principal importance for understanding the evolution of communication, but to our knowledge no such tasks have been addressed within a CB framework. Accordingly, referential communication provide an important challenge for models of communication based on a CB perspective.

In this paper, we present results from a series of experiments which explore the evolution of referential communi-

cation. In these experiments, agents interact with only simple sensors and motors, and hence without any pre-specified signalling mechanisms. These experiments thus provide an initial exploration of the evolution of referential communication through the lens of a CB perspective. The results demonstrate the successful evolution of referential communication using this approach; moreover, they provide insights into the kinds of subtle communication systems that are possible through the coupled interactions of embodied dynamical agents.

The rest of this paper is organized as follows. In the next section, we expand upon the notion of referential communication and consider how it might fit with the CB view. Then we present results from a series of three experiments which explore the evolution of referential communication under various conditions. Finally, we conclude with a general discussion of the experimental results, and outline some directions for future work.

Referential Communication

One of the most widely studied examples of referential communication in a nonhuman species is the waggle dance of honeybees (Crist, 2004; Dyer, 2002). When a forager bee discovers a lucrative food source, she will often return to her hive and congregate with other hive mates. The returned forager then performs a "dance", consisting of repeated runs in a small figure-eight pattern, amidst tightly packed adjacent bees. As was first identified by Karl von Frisch (von Frisch, 1950, 1967), and later elaborated by many others (Dornhaus and Chittka, 2004; Michelsen, 2003; Riley et al., 2005), various aspects of this dance correlate with the distance and direction to the previously identified food source. Having observed the dance, other bees are then able to successfully navigate to the new food source. Thus, the waggle dance provides an excellent example of referential communication, with signals used to communicate about states of affairs that are distant in both space and time.

In what follows, we present results from a series of three experiments which were inspired by the waggle dance of honeybees. Before proceeding, it is first necessary to establish an operational notion of referential communication in terms of behavioral coordination. The difficulty is that a standard conception of referential communication fits most naturally with an IT perspective. Intuitively, signals are "about" something, and that which they are about is the information that they convey. Identifying referential communication from a CB perspective, however, is less obvious.

Communication from a CB perspective is understood in terms of the effect that interactions between agents have on the future behavioral trajectories of those agents. Here we will consider only asymmetric interactions between pairs of agents, a *sender* and a *receiver*, in which case the primary concern is the effect of interactions on the behavior of the receiver. Intuitively then, we consider an interaction to be communicative when the future behavior of the receiver is sufficiently constrained as a result of its interaction with the sender. To demonstrate this idea, consider the waggle dance example. In absence of the dance interaction, the behavioral trajectories of would-be recruits are effectively unconstrained and, in principle, the recruits may travel to any location outside of the hive. Thus, the specific effect of the dance, and what makes it communicative, is that it serves to constrain the behavior of recruits to the subset of behavioral trajectories which result in their arrival at the food source.

The additional component necessary for an account of *referential* communication is the dependence of these constrained behavioral trajectories on an object of reference. That is, the receiver's behavior resulting from a communicative interaction should change as properties of the referent change. In the waggle dance, for example, the future behavioral trajectories of the receiver vary directly and reliably with properties of the referent, namely, the distance and direction of the food source.

Another important aspect of referential communication from a CB perspective is the nature of the interaction between the agents. Specifically, in order for an interaction to be considered referential there should be a degree of separation, spatial and/or temporal, between the communicative interaction and the object of reference. To demonstrate this idea, consider again the waggle dance. Rather than perform a dance, the forager could instead gather recruits and fly with them to the food source. This would result in the same behavioral outcome for the receiver, but in this case the interaction between the bees would persist until the recruits had reached the food source. Thus, we would not consider such an interaction to be referential. In contrast, the waggle dance is referential because the communicative interaction is spatially removed from the food source.

To summarize, we propose an operational notion of referential communication from a CB perspective based on the following considerations. Firstly, the future behavior of a receiver should be constrained by its interaction with a sender. Secondly, the nature of the receiver's constrained behavior should vary based on properties of the referent. Finally, the communicative interaction should have a degree of separation from the referent.

Methods

In all of the following simulations, two agents coexist in a one-dimensional circular environment (Figure 1). Agents are able to move around the circle with a maximum angular velocity of $\frac{\pi}{8}$ in either direction, with agents free to move past each other unimpeded. Each agent is equipped with two angular sensors, one each in the clockwise and counterclockwise directions, with each sensor having a maximum range of $\frac{\pi}{8}$. An angular sensor responds with a value inversely proportional to the distance at which it intersects the other agent, with sensor values $\in [0, 1]$. In this way, the

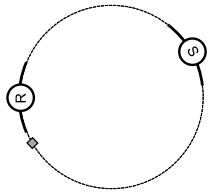


Figure 1: The agents and environment. Two agents, a sender (S) and a receiver (R), interact in a one-dimensional circular environment. On each trial, a target angle (gray diamond) is selected, and the sender must communicate the location of the target to the receiver.

angular sensors provide each agent with local information regarding the relative position of the other agent.

In addition to a pair of angular sensors, each agent is also equipped with two "bearing" sensors. The bearing sensors provide each agent with information about a certain angle ψ, with the two bearing sensors taking on the values $\frac{sin(\psi)+1}{2}$ and $\frac{cos(\psi)+1}{2}$, respectively. For one of the agents, henceforth the *receiver*, the angle ψ_R indicated by the bearing sensors is the receiver's current angular position. For the other agent, henceforth the *sender*, the angle ψ_S indicated by the bearing sensors is the separation between the sender's current angular position and a certain target angle in the environment. As will be elaborated later, the task for the agents is to get the receiver to the target angle, whose location is only available to the sender through its bearing sensors. As a result, the sender and the receiver must structure their interactions, which take place exclusively through the angular sensors, so that the target angle is successfully conveyed from the sender to the receiver.

The behavior of each agent is controlled by a continuous-time recurrent neural network (Beer, 1995) with the following state equation:

$$\tau_i \dot{s_i} = -s_i + \sum_{j=1}^{N} w_{ji}\sigma(s_j + \theta_j) + I_i \quad i = 1, \ldots, N$$

where s is the state of each neuron, τ is the time constant, w_{ji} is the strength of the connection from the j^{th} to the i^{th} neuron, θ is a bias term, $\sigma(x) = \frac{1}{1+e^{-x}}$ is the standard logistic activation function, and I represents an external input. The output of a neuron is $o_i = \sigma(s_i + \theta_i)$. In each agent, the two angular sensors and two bearing sensors are fully connected to a layer of five interneurons. The interneurons are fully interconnected and project fully to two motor neurons. The angular velocity of an agent is proportional to the difference between the outputs of the two motor neurons.

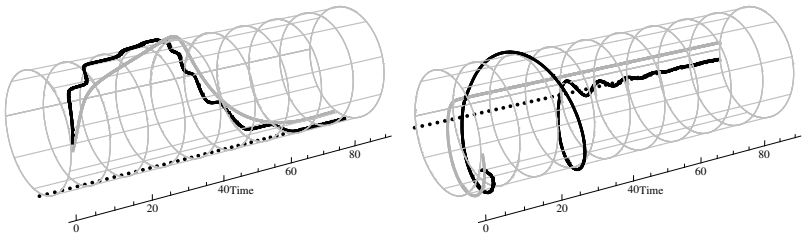
Neural parameters are evolved using a real-valued genetic algorithm with rank based selection. Each genome encodes two separate neural controllers for a sender and a receiver. Thus, rather than using a co-evolutionary procedure to evolve senders and receivers separately, we evolve a population of sender/receiver pairs. The following neural parameters, with corresponding ranges, are evolved: time constants $\in [0, 30]$, biases $\in [-16, 16]$, and connection weights (from sensors to neurons and between neurons) $\in [-16, 16]$. Successive generations are formed by first applying random Gaussian mutations to each parent genome with a mutation variance of 7 (see (Beer, 1996) for details). In addition, one-point modular crossover is applied with 5% probability, using two modules corresponding to the sender and the receiver neural controllers. A child replaces its parent if its performance is greater than or equal to that of the parent; otherwise the parent is retained.

Sender/receiver pairs are evolved for the ability of the receiver to successfully reach a number of specified target locations. On any given trial, a certain angle is designated as the target and the corresponding bearing angle inputs are given to the sender. The agents then interact for a fixed period of time, after which the receiver's final separation from the target angle is recorded. Since the information about the target angle is only available to the sender, success in this task requires that the sender and the receiver evolve a system for accurately communicating the target locations.

The performance of a sender/receiver pair is determined based on a number of evaluation trials. Each trial proceeds as follows. First, the neural states of both sender and receiver are initialized to 0. The sender is given an initial angular location of 0 and the receiver is positioned with an initial offset relative to the sender $\in \{-\frac{\pi}{32}, -\frac{2\pi}{32}, -\frac{3\pi}{32}, \frac{\pi}{32}, \frac{2\pi}{32}, \frac{3\pi}{32}\}$, such that the sender and the receiver are initially within sensory range of each other. Next, one of a set of target angles is chosen as the current target, with different sets of targets used for each experimental condition. The sender's bearing angle inputs are set to reflect the current target location and the agents are allowed to interact for an initial period of 80 time units, which is enough time for an agent moving at its maximum velocity to traverse the circular environment five times over. The duration of this initial period was chosen to minimize time pressure on the evolved communication systems. After the initial period has elapsed, the angular separation d_1 between the receiver and the target angle is recorded and normalized to run between 0 and π. The simulation is then continued for an additional 10 time units and the receiver's distance from the target is again recorded as d_2. This second value is incorporated into the fitness evaluation in order to ensure that the receiver remains at the target location. The score that a sender/receiver pair receives on a given trial is:

$$1 - \frac{d_1 + d_2}{2\pi}$$

and overall fitness is determined by averaging trial scores for every possible combination of target angles and initial receiver positions.



(a) (b)

Figure 2: Communication in the unconstrained condition. In each plot, the trajectories of the sender (gray line) and receiver (black line) are shown for an individual trial. The black dotted line indicates the target location. The shepherding strategy is shown in (a): the sender guides the receiver to the target location through a series of "push" or "pull" interactions. The "sit and wait" strategy is shown in (b): the sender sits a fixed distance away from the target location and waits for the receiver. Note that all plots of this kind in the rest of the paper follow the same format.

Unconstrained interactions

In the first set of simulations, agents were evaluated with a set of ten target locations uniformly distributed around the circle. Twenty evolutionary runs were performed, with a population of 400 sender/receiver pairs evolved for 2000 generations in each. The best sender/receiver pair in each run attained a fitness of at least 99%. Moreover, it was found that all of the best sender/receiver pairs had developed communicative strategies that could readily generalize to previously unseen target locations. When tested with 10,000 random trials, using target locations drawn uniformly from $[0, 2\pi]$ and initial receiver positions $\in [-\frac{3\pi}{32}, \frac{3\pi}{32}]$, the best pair from each run attained a score of at least 97%.

This first condition places no restrictions on the kinds of interactions that are available to agents. Any interaction which results in the receiver reaching the target location is acceptable, and the only aspect that matters for fitness is the receiver's final separation from the target location. Thus, in one sense, these simulations can be seen as an initial proof of concept, verifying that agents in this environment are capable of accomplishing the task. However, the results from these simulations also prove to be interesting in their own right, providing some initial insights into the kinds of interactions that agents may use for communication in this task. A preliminary inspection revealed two qualitatively distinct strategies, each of which we describe next.

The first strategy, used by 12 of the 20 best agent pairs, we refer to as "shepherding". An example of this strategy is displayed in Figure 2(a). Agents employing this strategy typically remain within or nearly within sensory contact for the entire duration of a trial. Thus, the trajectories of agents using this strategy are closely coupled. Over the course of a trial, the sender typically moves in a sustained direction towards the target location. In addition, the sender's movement towards the target is accompanied by a series of "push" or "pull" interactions with the receiver. That is, as the sender moves towards the target location, it brings the receiver along with it by closely governing its motion through repeated interactions. The sender then stops near the target location, causing the receiver to stop at or near the target. Thus, the sender's actions serve to effectively guide the receiver to the target.

The second strategy, used by 8 of the 20 best pairs, can be described as a "sit and wait" strategy. Figure 2(b) shows a characteristic interaction from a pair of agents using this strategy. At the beginning of a trial, the sender and receiver start off traveling together clockwise around the circle. The receiver continues traveling in this direction, making a full pass around the circle, while the sender stops and takes up a fixed location. Crucially, in all trials the position at which the sender stops is always the same fixed distance away from the target. The receiver continues traveling around the circle until it again reaches the position of the sender. At this point, the receiver begins moving in the opposite direction and oscillates back and forth before settling at the target location. In different versions of this strategy, the sender halts its motion at different distances away from the target and the specific details of the interaction vary, but the same general characteristics hold. To summarize, this strategy can be glossed as follows: (1) the sender moves to a fixed distance away from the target location and stops; (2) the receiver travels until it finds the sender; (3) the receiver positions itself the same fixed distance away from the sender, thus coinciding with the target location.

A common feature of the behavioral strategies evolved in this condition is that the communicative interactions continue until the receiver reaches the target location. In the "shepherding" strategy, the sender accompanies the receiver to the target location, while in the "sit and wait" strategy the sender indicates the location of the target directly using its own position. Consequently, these strategies are not considered referential, since there is no separation between the communicative interaction and the object of reference. The next two experiments address this issue by specifically

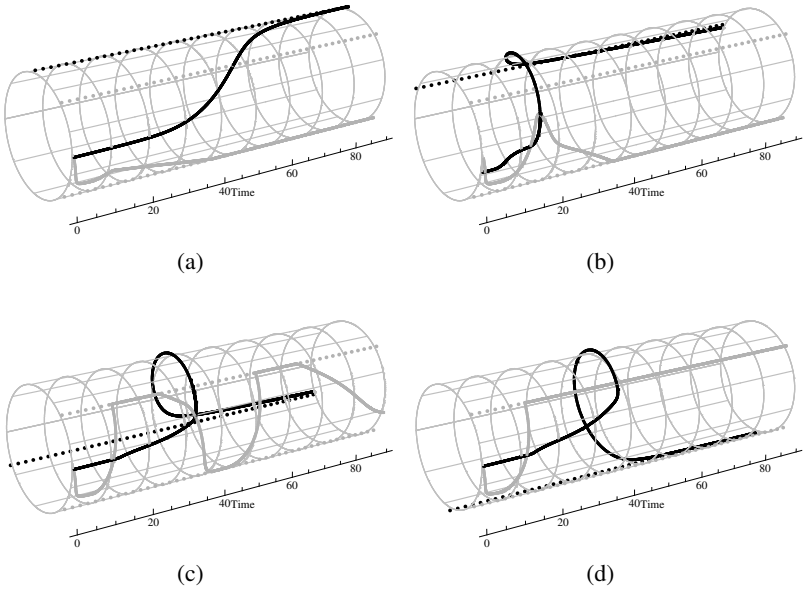


Figure 3: Communication strategy with discrete targets. Behavioral trajectories from the best sender/receiver pair are shown for each of the four target locations. The gray dotted lines indicate the boundaries of the sender constrained region. In this condition, the sender uses qualitatively distinct behavioral patterns to communicate each of the four target locations.

evolving for referential interactions.

Constrained interactions with discrete targets

There are several approaches that we could use to select for referential interactions in our simulation, which have interesting parallels with potential accounts for the evolution of referential communication in nature. One possibility would be to associate an energy cost with the behavior of the sender and to select for behaviors that minimize this cost. That is, we could select for interactions in which the sender does less while producing the same behavioral outcome for the receiver. In natural evolution, energy minimization presumably provides a strong selective advantage for referential communication. An alternative, though related, possibility is to impose spatial restrictions on the interactions between communicating agents. In considering the waggle dance, for example, it may be the case that an important selective pressure was the restriction to interactions that take place within the hive.

In the second experimental condition, we adopt the latter strategy of imposing a strict spatial constraint on the interactions between the sender and the receiver. Specifically, the sender is constrained to move within the region between $\frac{\pi}{4}$ and $-\frac{\pi}{4}$ (Figure 4). Agents in this condition are evaluated with a set of four target locations, uniformly distributed between $\frac{\pi}{2}$ and $\frac{3\pi}{2}$. Thus, given the constraint on the sender and the target locations, it is impossible for the agents to remain within sensory contact as the receiver goes to a target.

Twenty evolutionary runs were performed with a population size of 400 in each. Successful strategies proved to be more difficult to evolve in this condition, so populations were evolved for 10,000 generations as opposed to the 2,000 used in the first experiment. In all of the runs, the best pair attained a fitness measure of at least 92%, with the top 5 pairs achieving over 99%. Again we found a variety of behavioral strategies employed by different pairs of agents. We next outline one such strategy, which comes from the best sender/receiver pair.

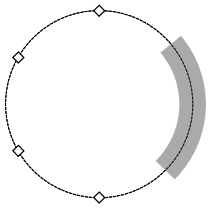


Figure 4: The constrained environment with discrete targets. The sender's motion is constrained between $-\frac{\pi}{4}$ and $\frac{\pi}{4}$ (gray region). Diamonds indicate the locations of the four targets.

Sample trajectories from the best pair of agents are displayed in Figure 3. Note the qualitatively different behavior exhibited by the sender for each of the four target locations. When the target is at $\frac{\pi}{2}$ (Figure 3(a)), the sender moves immediately to the lower boundary of the constrained region. As a result, the two agents interact for only the initial portion of the trial, after which time their trajectories quickly diverge. For the next target, proceeding counterclockwise

(Figure 3(b)), the sender motions back and forth at the beginning of the trial, before again moving to the lower boundary of the constrained region. In this case, the sender and receiver cross paths multiple times over the course of the trial. For the third target, the sender makes broad oscillatory motions between the two boundaries of the constrained region, resulting again in several intersections with the path of the receiver. Finally, for the last target location, the sender sweeps across the constrained region once before settling on the upper boundary of the constrained region, which results in the agents crossing paths relatively late in the trial.

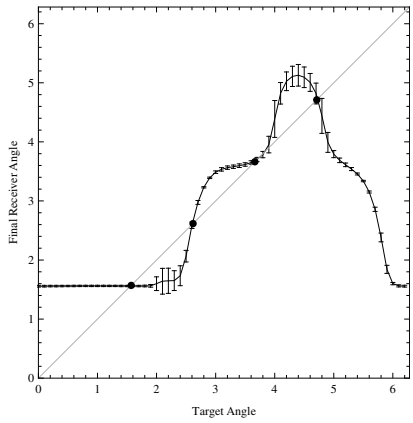


Figure 5: Generalization of the constrained agent with discrete targets. The receiver's final position across the range of target locations is shown. Black dots indicate the four target locations on which the agents were evolved.

The qualitatively distinct behaviors of the sender turn out to be a general feature of the behavioral strategies that evolved in this condition. For each of the four targets, the sender exhibits a different pattern of behavior which serves to distinguish the locations and appropriately guide the receiver. When tested on intermediate target locations, the sender makes abrupt shifts between its different behavioral regimes. A reasonable prediction then is that the behavior of the receiver will be similarly distinguished. Figure 5 shows the final position of the receiver across the range of target locations, and verifies this prediction. The receiver typically moves to one of the four standard target locations and exhibits sharp transitions between these locations for intermediate values. Thus, these results suggest that the evolved communication systems are, in a certain sense, symbolic: discrete and categorical "signals" are associated in an arbitrary fashion with a small set of possible locations.

Constrained interactions with a continuum of targets

The communicative strategies evolved in the previous experiment can be viewed as analogous to a simple form of words. Senders use a small set of essentially arbitrary signals to distinguish between a discrete number of targets. An alternative communicative strategy would involve the use of a continuum of signals to indicate a similarly continous range of possible targets. Such is presumably the case with the waggle dance, where a continuous range of dance maneuvers can be used to communicate a range of distances and directions. Other examples of this kind of communication system are the deictic indicators used by humans, such as finger pointing and eye gaze. The third experimental condition explores the evolution of this kind of referential communication.

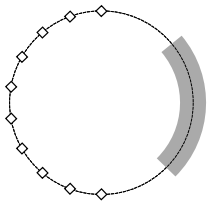
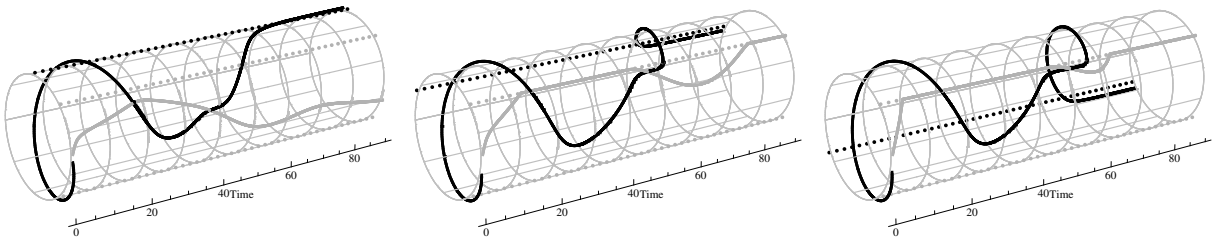


Figure 6: The constrained environment with a continuum of targets. The sender was again constrained (gray region), and the agents were evaluated with 10 target locations (diamonds).

In this experiment, agents were evolved with a set of ten target locations, uniformly distributed between $\frac{\pi}{2}$ and $\frac{3\pi}{2}$ (Figure 6). The larger number of targets was used to pressure agents into developing communicative strategies that can generalize to a continuum of locations, as opposed to using distinct signals for a small number of discrete targets. The sender was again constrained to move within the region bounded by $\frac{\pi}{4}$ and $-\frac{\pi}{4}$.

We again performed twenty evolutionary runs with this condition, with a population of 400 agents evolved for 10,000 generations in each. The best sender/receiver pair in each run achieved a fitness of at least 94%, with the top seven pairs attaining fitness scores over 99%. In order to verify that the evolved strategies could generalize to a continuum of targets, we evaluated each of the best pairs on 10,000 trials with target locations drawn uniformly from $[\frac{\pi}{2}, \frac{3\pi}{2}]$ and initial receiver offsets $\in [-\frac{3\pi}{32}, \frac{3\pi}{32}]$. The best pair from each run achieved a score of at least 90%, with the top seven pairs all scoring in excess of 98%. Figure 8 shows the generalization performance of the best sender/receiver pair. We next describe the behavioral strategy used by this pair of agents.

Sample trajectories from the best sender/receiver pair are displayed in Figure 7. In each case, the interaction between the agents consists primarily of two path crossings. At the beginning of the trial, the sender moves counterclockwise while the receiver makes a full pass around the circle in the clockwise direction. The paths of the agents then cross for the first time, after which the receiver turns back in the counterclockwise direction. The agents then cross each other for the second time, followed by the receiver continuing around



(a) (b) (c)

Figure 7: Communication strategy with a continuum of targets. The gray dotted lines indicate the boundaries of the sender constrained region. For targets further from the sender in the clockwise direction, the sender's trajectory becomes increasingly straightened against the upper boundary of the constrained region.

the circle in the clockwise direction before stopping at the target location. Note that, for targets further from the sender in the counterclockwise direction, the sender's trajectory becomes increasingly straightened, effectively flattening out against the upper boundary of the constrained region. Importantly, this change in the sender's trajectory affects the timing and location of the path crossings between the two agents. As the sender's path becomes straighter, the first intersection between the paths occurs earlier and the second intersection occurs later. Figure 9 demonstrates this more clearly, where trajectories are shown for a range of target locations. The interactions between the agents vary smoothly with the location of the target. As a result, the agents are able to systematically communicate the location of targets anywhere along the continuum.

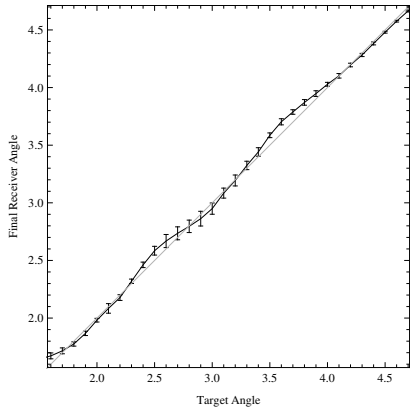


Figure 8: Generalization of the constrained agent with a continuum of targets.

Discussion

Previous models of emergent referential communication are based on the view of communication as information transmission. In this paper, we presented results from a set of experiments which explored the evolution of referential communication from the perspective of coordinated behaviors. Embodied dynamical agents, interacting with only basic sensory and motor capabilities, evolved strategies to communicate the locations of spatially distant targets.

In the first experimental condition, agents were evolved with no restrictions placed on their potential interactions. In this case, we found that agents developed two distinct strategies that were successful in guiding the receiver to the targets. These results provided an initial proof of concept, as well as giving some insights into the kind and variety of communicative strategies that are possible.

The second experiment selected specifically for referential interactions by placing spatial restrictions on the motion of the sender. With a small set of discrete targets we found that senders evolved a number of distinct behavioral patterns to communicate the different target locations. Thus, in a loose sense, the communication systems evolved in this condition can be viewed as similar to words.

Finally, in the third condition, we explored the possibility of evolving referential communication which could generalize to a continuum of target locations. The "signals" evolved in this condition were found to vary smoothly with the range of targets, resulting in successful generalization. This ability to indicate a continuum of locations is analogous to the various deictic indicators used in human communication, such as finger pointing and eye gaze.

Future directions

The next step for future work is to perform detailed analyses of the evolved communication systems. Using the mathematical tools of dynamical systems theory, we will explore how the underlying dynamical structure of individual agents forms the basis for their joint interactions. Additionally, we plan to apply analytical techniques from information theory to examine the possibility of developing a mathematically rigorous notion of information transmission between interacting agents.

In other work, we plan to extend the approach used here to a two-dimensional environment. With the added dimensionality, we could explore tasks in which agents have to

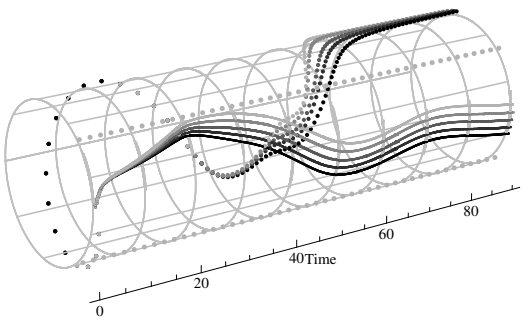


Figure 9: Signalling systematicity of the constrained agent with a continuum of targets. Trajectories of the sender (solid lines) and receiver (dotted lines) are shown for a set of equally spaced target locations, with color used to indicate trajectories from the same trial. The behavioral interactions of the agents vary smoothly with the location of the target.

communicate multiple kinds of information simultaneously. For example, we could bring the task closer to its waggle dance inspiration by evolving agents to communicate both the distance and direction to target locations.

Finally, we intend to investigate the evolutionary trajectories of the evolved communication strategies. To do so, we can track the individual lineages of sender/receiver pairs and study their behavioral interactions at different times during the course of evolution. Such an investigation should provide unique insights into the process by which various non-communicative behaviors adapt to serve communicative functions.

Acknowledgements

This investigation was supported by the National Institute of Health and National Research Service Award HD007475-13.

References

Baldassarre, G., Nolfi, S., and Parisi, D. (2003). Evolving mobile robots able to display collective behaviors. *Artificial Life*, 9(3):255–267.

Beer, R. D. (1995). On the dynamics of small continuous-time recurrent neural networks. *Adaptive Behavior*, 3(4):469–509.

Beer, R. D. (1996). Toward the evolution of dynamical neural networks for minimally cognitive behavior. *From Animals to Animats 4*, pages 421–429.

Cangelosi, A. and Parisi, D. (1998). The emergence of a 'language' in an evolving population of neural networks. *Connection Science*, 10(2):83–97.

Crist, E. (2004). Can an insect speak?: The case of the honeybee dance language. *Social Studies of Science*, 34(1):7–43.

Di Paolo, E. A. (1997). Social coordination and spatial organization: Steps towards the evolution of communication. *Fourth European Conference on Artificial Life*, pages 464–473.

Di Paolo, E. A. (1998). An investigation into the evolution of communication. *Adaptive Behavior*, 6(2):285–324.

Di Paolo, E. A. (2000). Behavioral coordination, structural congruence and entrainment in a simulation of acoustically coupled agents. *Adaptive Behavior*, 8(1):25–46.

Dornhaus, A. and Chittka, L. (2004). Why do honey bees dance? *Behavioral Ecology and Sociobiology*, 55(4):395–401.

Dyer, F. C. (2002). The biology of the dance language. *Annual Review of Entomology*, 47(1):917–949.

Hauser, M. D. (1997). *The evolution of communication.* MIT Press.

Iizuka, H. and Ikegami, T. (2003). Adaptive coupling and intersubjectivity in simulated turn-taking behaviours. *Advances in Artificial Life: 7th European Conference.*

MacLennan, B. J. and Burghardt, G. M. (1993). Synthetic ethology and the evolution of cooperative communication. *Adaptive Behavior*, 2(2):161–188.

Maturana, H. R. (1978). Biology of language: The epistemology of reality. *Psychology and biology of language and thought: Essays in honor of Eric Lenneberg*, pages 27–63.

Maturana, H. R. and Varela, F. J. (1980). *Autopoiesis and cognition: The realization of the living*. D. Reidel.

Michelsen, A. (2003). Signals and flexibility in the dance communication of honeybees. *Journal of Comparative Physiology A: Sensory, Neural, and Behavioral Physiology*, 189(3):165–174.

Nolfi, S. (2005). Emergence of communication in embodied agents: Co-adapting communicative and non-communicative behaviours. *Connection Science*, 17(3):231–248.

Quinn, M. (2001). Evolving communication without dedicated communication channels. *Advances in Artificial Life: 6th European Conference*, pages 357–366.

Quinn, M., Smith, L., Mayley, G., and Husbands, P. (2003). Evolving controllers for a homogeneous system of physical robots: Structured cooperation with minimal sensors. *Philosophical Transactions: Mathematical, Physical and Engineering Sciences*, 361(1811):2321–2343.

Riley, J. R., Greggers, U., Smith, A. D., Reynolds, D. R., and Menzel, R. (2005). The flight paths of honeybees recruited by the waggle dance. *Nature (London)*, 435(7039):205–207.

Seyfarth, R. M. and Cheney, D. L. (2003). Signalers and receivers in animal communication. *Annual Reviews in Psychology*, 54(1):145–173.

Smith, M. J. and Harper, D. G. (1995). Animal signals: models and terminology. *Journal of Theoretical Biology*, 177(3):305–311.

Steels, L. (2003). Evolving grounded communication for robots. *Trends in Cognitive Sciences*, 7(7):308–312.

von Frisch, K. (1950). *Bees: their vision, chemical senses, and language*. Cornell University Press.

von Frisch, K. (1967). *The dance language and orientation of bees*. Harvard University Press.

Species Selection of Aging for the Sake of Diversity

Owen G. Woodberry, Kevin B. Korb, and Ann E. Nicholson

Clayton School of Information Technology
Monash University, Clayton, Victoria 3800, Australia
{owenw,korb,annn}@infotech.monash.edu.au

Abstract

In their paper, "Punctuated Equilibria: An Alternative to Phyletic Gradualism", Eldredge and Gould (1972) argue that most evolution occurs during geologically rapid speciation events, with species exhibiting stasis the vast majority of the time. Gould (2002, 2007) demonstrates that an implication of Punctuated Equilibrium is that selectionist theory is expanded to the level of species - defining species as basic units of macroevolution. In our paper, we demonstrate the evolution of aging rates in a species selection scenario. We have developed an ALife simulation environment with mating governed by evolving compatibility signatures, resulting in the formation of reproductively isolated subpopulations (i.e., species). Given a co-evolving parasite population, heterogeneity in a host subpopulation is beneficial for the health of that subpopulation. This can result in group selection pressure at the species level for the evolution of altruistic traits, such as a faster aging rate.

Introduction

The mechanism of aging has perplexed those who have attempted to understand it in a Darwinian framework. The wide and enduring variations in longevity across species suggests that aging rate is the result of natural selection. The theory of evolution by natural selection holds that many of the traits we observe in organisms are the result of adaptations to the environments of their ancestors. However, any possible adaptive benefit of faster aging cannot accrue to its bearer, who will likely have fewer offspring. Therefore, the evolution of aging requires a selection mechanism beyond the individual. Such selection mechanisms have been proposed by Wynne-Edwards (1962), in the form of group selection theory, and Hamilton (1964), in the form of inclusive fitness theory. In previous papers we showed that these mechanisms are interdependent in a prominent simulation model of the evolution of aging (Woodberry et al., 2005, 2007).

Bell (1982) posited that the diversity created by sexual recombination provides a group benefit in the co-evolution arms race between hosts and parasites. In a similar manner we argue that aging benefits the group by increasing population turnover and, thus, genetic diversity. Here we reinforce our hypothesis by demonstrating its practicality in a species selection scenario. There have been investigations into the implications of Punctuated Equilibrium on selectionist theory — identifying species as basic units of macroevolution, due to their stability (and hence individuality) between punctuation events (see Gould, 2002, 2007). On the individuality of species, Gould (2002) says: "So long as most new species arise by branching (speciation) rather than by transformation (anagenesis), species can be individuated by their uniquely personal duration, bounded by birth in branching and death by extinction. "

In this paper, we describe the design and use of an ALife simulation containing co-evolving host and parasite populations. Host agents have mating signatures, which are used to determine mate compatibility, and vulnerability signatures, which govern the infection process. Mutation of the signatures may cause the host population to diversify into reproductively isolated subpopulations, i.e., to speciate. The parasite population, which has infection and virility signatures, flourishes when the host population's vulnerability signatures are genetically uniform, creating a positive selective pressure for the evolution of aging for the sake of species diversity. Here we further our argument for the evolution of aging by demonstrating it in a species selection scenario — which arguably is more realistic than classical group selection.

Background

The Evolution of Aging

Aging is defined as the general deterioration of an organism, and its eventual death, by internal causes (Williams, 1957). The rates at which different species age is a perplexing phenomenon. The divergence is extraordinary, ranging from a few hours for some phytoplankton cells (Agustí et al., 1998) to a few days for some insects to thousands of years for the bristlecone pine tree. Furthermore, these different rates have themselves not varied greatly during recorded history, so far as we can tell. The genetic control of aging is beginning to come into view, with multiple genes already identified as participating in aging rates (e.g., Belenky et al., 2007). All

of this seems to suggest that aging rates have evolved because of their adaptive value. However, the obvious fitness costs of fast aging on individuals would cause strong direct selection pressure against it, suggesting that aging may be a side effect of some more essential characteristic, i.e., that it is non-adaptive. Historically, both adaptive, e.g., Weismann (1889), and non-adaptive, e.g., Williams (1957) and Medawar (1952), explanations of aging have been proposed. Recently there has accumulated compelling experimental evidence that aging *is* an adaptation (Mitteldorf, 2004; Bredesen, 2004; Skulachev, 1997). This has lead to a resurgence of research into possible adaptive benefits of aging, including our own.

In Woodberry et al. (2007) we posited an adaptive explanation of aging. We argued that aging has a group fitness benefit which can outweigh the individual fitness cost. Groups with shorter individual life spans turn over faster and consequently have greater genetic diversity. In co-evolution scenarios, e.g., predator-prey and host-parasite interactions, groups with greater diversity will be less easily exploited, creating a stronger and healthier population.

Group and Kin Selection

An adaptive explanation of aging requires a selection mechanism accounting for the potential selection of altruistic traits. The papers of Wynne-Edwards (1962) on group selection and Hamilton (1964) on inclusive fitness theory attempted to give a mathematical analysis supporting selection mechanisms beyond individual selection. Maynard Smith (1976) went on to create a model demonstrating the logical possibility of group selection. He showed that the turnover of groups, via extinction and pioneering, can favor altruistic groups which, because of their altruism, have greater group lifespans and so greater opportunity to found new groups. This enables scenarios where cheaters, even while having a fitness advantage within groups, do not take over the population. Inclusive fitness (or kin selection) theory, by contrast, shifts the focus downwards from the individual to the gene, whether held by the individual or, as a replica, by a relative. The inclusive fitness of a gene is just the organism's individual fitness augmented by the harms and benefits caused to the fitness of others, weighted by their relatedness, i.e., the probability of their carrying the same allele (Hamilton, 1964). Inclusive fitness theory, or kin selection, has become widely accepted, especially as an explanation for the evolution of altruistic behavior. The group selection concept, however, has remained contentious. It has been doubted, for example, whether the selection pressure for selfish behavior within groups can be overcome in nature by selection pressure for altruism between groups. In Woodberry et al. (2005) we argued that group selection may be dependent upon kin selection, rather than in opposition to it, as most would have it. That is, kin selection may well provide selection pressure within groups for an altruistic trait that is also being selected for at the group level, when the latter selection pressure would be insufficient for evolutionary stability of the trait on its own.[1]

Species

The concept of species, as a taxonomic classification, remains central to biology and a host of related fields. The definition of species remains controversial, as there is an inherent vagueness in its application, e.g., asexual species, ring species and hybrids. The most generally accepted definition of species, which we follow, is a reproductively isolated sub-population (Mayr, 1963) — that is, a group of actually or potentially interbreeding populations that are reproductively isolated from other such groups. Studies of speciation are based on geographic circumstances: allopatric and peripatric speciation rely on geographic isolation, whereas sympatric and parapatric speciation are based on the emergence of new species with little, or no, geographic isolation. In our simulation, as there are no barriers to migration, speciation must be described in the latter terms.

Punctuated Equilibrium

Eldredge and Gould (1972) drew attention to what they saw as a mistaken view that evolution can only occur gradually and, indeed, can only occur at a constant, continuous rate — a concept they labelled Phyletic Gradualism. They argued instead that most evolution occurs during geologically short-term speciation events, with species exhibiting approximate stasis the vast majority of the time. They claimed that this punctuated equilibrium view of evolution is more consistent with the observations made in the fossil record.

Under the Punctuated Equilibrium concept, once a species becomes static and defined, it takes on a kind of individuality. It has a lifespan; it has the opportunity to reproduce through speciation; and, in the end, it will disappear. This supports a metaphorical similarity with individual reproduction and, therefore also, with individual fitness (Gould, 2002). But the similarity is more than metaphorical with group selection, for this just is a kind of group selection. Species become units of selection, competing with other species within the biosphere for the opportunity to create new species and to avoid early extinction; this creates a species selection mechanism which falls under the group selection model described above, and which caters for the evolution of altruistic traits.

Simulation Design

To test hypotheses about aging, we designed a multi-agent ALife simulation environment. Co-evolving populations of host and parasite agents interact within overlapping neighbourhoods on a board, sharing food sources and potentially

[1] Multilevel selection theory asserts the compatibility of multiple levels of selection, rather than their interdependency (Wilson and Sober, 1994).

reproducing sexually. Table 1 provides an overview of the simulation parameters, which are discussed in depth below. When designing simulations, it is necessary to consider the trade-off between the complexity of the model and its completeness. Although more complex models are harder to analyse, simpler models could neglect important mechanisms that allow validation against real systems (Grimm et al., 2005). In our design process, we tried to find a satisfactory trade-off for our simulation.

Table 1: Simulation Parameters

Parameter	Comment
Epoch Length	100 cycles
Run Length	100 epochs
Board Size	120×120 cells
Neighbourhood Size	3×3 cells
New Food	$N(1, \frac{1}{10})$ units
Initial Health	20 units
Parental Health Investment	10 units/parent
Health Energy Overhead	1 unit/cycle
Max Health	80 units
Mature Health	60 units
Accident Rate	0.1
Parasite Generate Rate	0.0001
Signature Length	100 bits
Signature Mutator	0.005
Initial Expiry Gene	20
Expiry Mutator	$N(Expiry, \frac{Expiry}{50})$
Initial Airborne Gene	0.05
Airborne Mutator	$N(Airborne, 0.001)$
Airborne Co-ordinate Jump	N(0,10)

World

Time: Simulation runs are divided into a number of epochs for statistics collection. During each epoch, the simulation world and agents are updated over a number of cycles, with statistical information collected and saved to file at the end of the epoch. The methods of agent and world updating are discussed in depth below.

Board: The simulation board consists of a square grid of cells, wrapped so that the edges meet, forming a torus shaped world. Each cell contains an occupant population, unlimited in size, and a food store. Each cycle the food store is replenished with new food, as determined by a normal distribution, and energy is recycled from the previous cycle, i.e., uneaten food and recycled agent energy (discussed later). The cell contents interact with the nine cells in the Moore neighbourhood — recycled food is distributed, evenly, to neighboring cells; and agents feed, mate and migrate freely within their neighbourhood.

Agents

Host Agent: The host agents are the focus of the simulation. Each host agent occupies the cell into which they were born. Each cycle all host agents have an opportunity, in a randomly selected order, to eat and reproduce, after which they are tested for death conditions. Figure 1 shows the algorithm used for updating hosts, discussed in detail below. The simulation maintains for each host agent its age, health, and chromosome. Age is initialised at zero and incremented each cycle. The health is incremented whenever the host agent successfully eats and is decremented each cycle by an energy overhead and also by a parental investment whenever the host agent reproduces. The chromosome is inherited at birth and, along with the states of the other variables and environment, determines agent behaviour.

Each cycle the host agent eats, unless its health is below zero or greater than a maximum value. A cell is selected, randomly, from the cells neighboring the agent's occupancy cell, and all the contents of that cell's food store are transferred to the host's health.

Host agents are genderless, but reproduce predominantly sexually. After the agent has eaten, it is tested against a health threshold; if the agent has sufficient health, it attempts to reproduce. In addition to avoiding suicidal mating, this forces agents to mature before reproducing, since new agents will lack sufficient health. When reproducing, the agent first checks its neighbourhood for any mate requests by compatible agents (compatibility is discussed later). If one is found, the agents reproduce sexually and two offspring are created. If the agent fails to find a mate, but its health exceeds its maximum health threshold, it will reproduce asexually. The initial health of the offspring is the sum of parent health donations.

There are three causes of death. The host agent dies if:

1. its health falls below zero;
2. its age exceeds a genetically determined expiry age (discussed later); or
3. it dies of external, accidental causes, as determined by an accident probability each cycle.

These first two causes of death are necessary to have an ecologically plausible test environment for examining theories of the evolution of aging. The implementation of accidental death is not a strict requirement of the simulation — however, it makes the simulation more realistic, by weakening selection pressures in favour of faster aging in a way we know operates in real populations. Having a closed ecosystem requires us to remove dead agents from the board and recycle any remaining energy held as health through the growth of new plant food.

```
decrement health
increment age
if parasite to be generated then
   generate and save parasite
if health < 0 OR age > expiry age OR accident then
   remove agent
   recycle health energy
else
   if health < max health then
      attempt eat action
   if health > mature health then
      if mate available then
         reproduce sexually
      else if health > max health then
         reproduce asexually
   save agent
```

Figure 1: Host Update Algorithm

Host Chromosome: The host agent chromosome contains:

- an expiry age gene;
- a mate compatibility bit string signature; and
- a vulnerability bit string signature.

The expiry age gene is used at conception to determine an expiry age for the agent, by sampling a normal distribution with variance proportional to its magnitude. It is inherited from a randomly selected parent with a chance of mutation, according to a normal distribution. As this gene has no side-effects, it is expected that fast aging would always be selected against unless the scenario provides aging its own selective value.

The mate compatibility signature is used to determine whether two agents are capable of mating. Compatibility is determined by testing whether the Hamming distance between the strings is greater than a fixed mating variance threshold (see Figure 2). The signature is inherited (via crossover) with a chance of mutation flipping each bit copied. This mechanism allows for the diversification of the mate signatures and thus the emergence of sexually isolated subpopulations, i.e., new species.

The vulnerability signature is used as an interface for parasite interaction. It is inherited and mutated in the same fashion as the mate compatibility signature. Its function is discussed in detail in the parasite section below.

Parasite Agents: The parasite agents live off the host agent population. There may be an unlimited number of parasites living off a single host; however, if the host has non-positive health, it cannot carry any more parasites, and will die when next updated. Figure 3 shows the algorithm

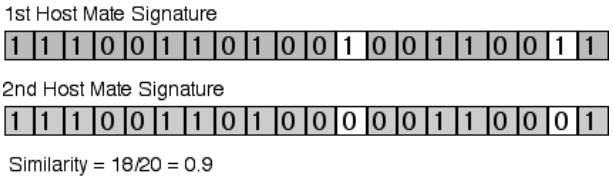


Figure 2: Example interaction between two host mate signatures. In this case, if the mate threshold parameter is set at a value lower than, or equal to, 0.9, the agents may mate, and thus are of the same species. (Note: all signatures in the simulation are 100 bits in length, not 20 as in this illustration.)

used for updating parasites, which we now discuss. Each cycle all parasite agents are transmitted to a new host in their neighbourhood, if there is one, which is randomly selected if there is more than one. Occasionally a parasite will become airborne (with a probability determined by a gene in its chromosome) and is transmitted to a random cell on the board and a random host within that cell, if there is one. The co-ordinates of the destination cell are determined by sampling a normal distribution. If the new location is unoccupied, the parasite fails to attach with a host and dies. Becoming airborne provides the parasite population the opportunity to infect new populations.

When a transmission is successful, airborne or otherwise, the parasite agent attempts, twice, to steal health from its host and use it to clone offspring (one for each successful health steal), which will act during the next cycle. Infection and reproduction (i.e., virility) are based on an interaction between the parasite and host chromosomes, discussed below. After the parasite has attempted reproduction, it dies. To ensure that the parasite population is never completely eradicated, there is a small probability a new parasite will be generated for every host agent updated.

```
transmit parasite to new host
if successful infection then
   for all reproduction attempts do
      if host health > 0 AND successful virility test then
         decrement health of host
         clone and save offspring
```

Figure 3: Parasite Update Algorithm

Parasite Chromosome: The parasite chromosome has three components:

- an infection bit string signature;
- a virility bit string signature; and
- an airborne probability gene.

The success of infection and parasite reproduction is determined via the interaction of the parasite's infection and virility signatures and the host agent's vulnerability signature. The probability of a successful infection is determined by the following function of the Hamming distance between the parasite's infection signature and the host's vulnerability signature:

$$P(infect) = \sqrt{\frac{Hamming\ Dist(infect, vulner)}{Signature\ Length}} \quad (1)$$

Likewise, the probability of successful reproduction is determined by inserting the Hamming distance between the parasite virility signature and the host vulnerability signature in the same function (see Figure 4). Both signatures are inherited from the parent parasite with a probability of mutation flipping each bit value.

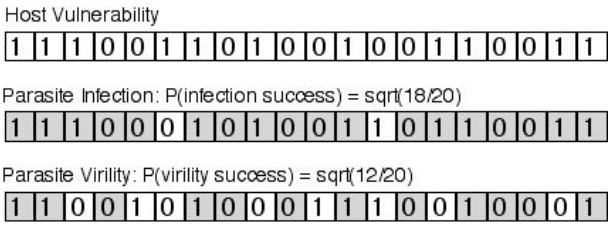


Figure 4: Example interaction between the host vulnerability signature and the parasite infection and virility signatures. (Note: all signatures in the simulation are 100 bits in length, not 20 as in this example.)

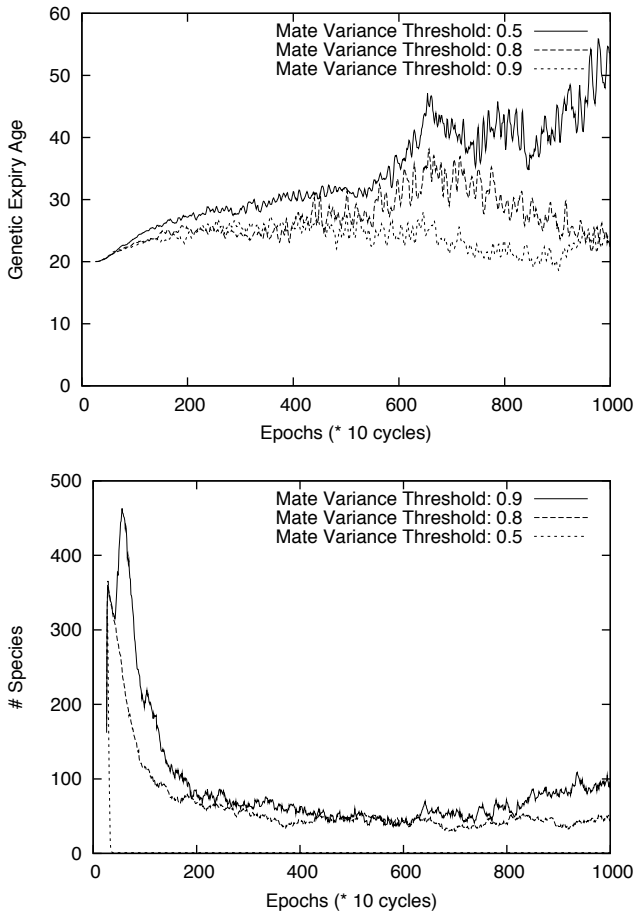


Figure 5: Figures tracking the evolution of (a) genetic expiry age and (b) number of species, for select cases of mate variance. Each of these plots represents a single simulation run. Resulting genetic expiry age and number of species are summarised in Figures 6(a) & 6(b).

Experiments

Evolution of Aging

In Woodberry et al. (2007) we demonstrated that our hypothesis of aging for the sake of diversity can be correct in a classical group structured simulation; here we extend that work, demonstrating its possibility also in a species selection scenario. In order to explore the effects of species groups on the evolved aging rate, simulations were run with a variety of mate variance thresholds (see Figures 5(a) & 5(b)) — a low threshold will allow host agents with greatly differing signatures to mate, reducing the number of species, whereas greater thresholds are more restrictive and thus produce a greater frequency of speciation. The resultant evolved genetic expiry ages and number of species for a range of mate variance thresholds are summarised in Figures 6(a) & 6(b).

From Figure 6(a) we can see that, as expected, the number of species present in the simulation increases as the limiting mate variance threshold increases. From Figure 6(b) we can see that as the mate variance threshold increases, and thus the number of species and the strength of inter-species competition increase, the expiry age gene evolves for shorter life spans. It is noteworthy that even when there is only one species, and thus no inter-species competition, the expiry age gene evolves to an equilibrium; this must be based simply on a background kin selection pressure, since there is no group or species structure. The species selection pressure acting on top of kin selection drives the aging rates higher — i.e., it drives lifespans downwards.

We conducted additional experiments to analyse the effect of varying the parasite virility (see Figure 7). We would expect virility to evolve via kin and group selection mechanisms in real parasite populations, however for simplicity we chose a parametric implementation of virility, enabling these experiments. These results show that when the parasites are over-virulent, the host population is quickly driven to extinction and consequently the number of species drops to zero. When the parasites are under-virulent, they fail to maintain a foothold in the host population. In consequence of this,

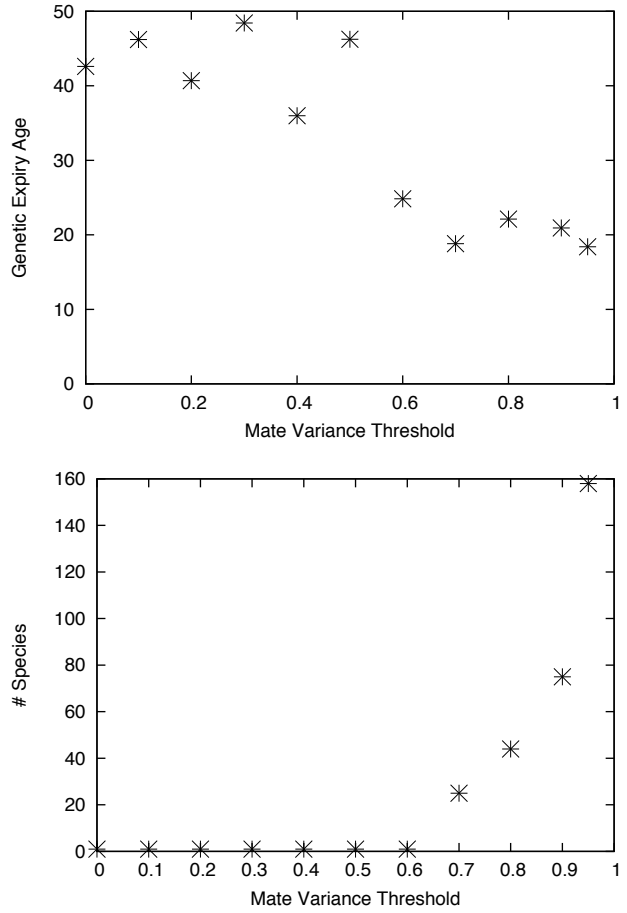


Figure 6: Resulting (a) number of species and (b) evolved expiry age, after simulation runs completed, with varied mate variance.

there are fewer dead-zones (where the parasites have killed off the host population), and the host population, which consumes the same amount of food regardless, spreads across the board, becoming less dense, which results in a greater rate of speciation.

Conclusion

The evolution of aging remains a puzzling phenomenon. Attempts to explain varying aging rates via individual selection have led many biologists to propose that it is non-adaptative — a side effect of some other beneficial trait — however, experimental evidence points to it being an adaptation. We argue that a primary benefit of aging is the generation of genetic diversity, which is of particular value in co-evolution scenarios. The evolution of altruistic traits such as adaptive aging requires an explanation of a selection mechanism that goes beyond the individual, such as kin and group selection. Punctuated equilibrium theory strongly suggests that species can support group selection. We have demonstrated with our

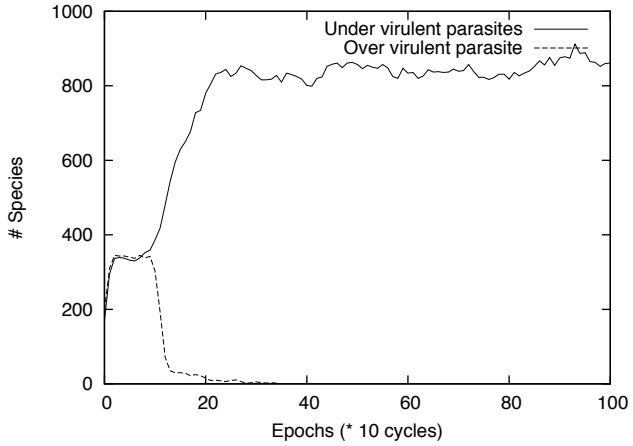


Figure 7: Experiments conducted varying the virulence of the parasites, i.e., changing the exponential in Equation 1, to 0.4 (under-virulent) and 0.6 (over-virulent).

simulation that species-level selection of an altruistic trait, in particular faster aging rates, can indeed occur.

References

Agustí, S., Satta, M., Mura, M., and Benavent, E. (1998). Dissolved esterase activity as a tracer of phytoplankton lysis: Evidence of high phytoplankton lysis rates in the northwestern mediterranean. *Limnology and Oceanography*, 43:1836–1849.

Belenky, P., Racette, F., Bogan, K., McClure, J., Smith, J., and Brenner, C. (2007). Nicotinamide riboside promotes sir2 silencing and extends lifespan via nrk and urh1/pnp1/meu1 pathways to nad+. *Cell*, 129:473–484.

Bell, G. (1982). *The Masterpiece of Nature: The Evolution and Genetics of Sexuality*. Croom Helm, London.

Bredesen, D. (2004). The Non-Existent Aging Program: How Does it Work. *Aging Cell*, 3:255–259.

Eldredge, N. and Gould, S. J. (1972). Punctuated Equilibria: An Alternative to Phyletic Gradualism. In Schopf, T. J. M., editor, *Models in Paleobiology*, pages 82–115, San Francisco. Freeman, Cooper and Company.

Gould, S. (2002). *The Structure of Evolutionary Theory*. Harvard University Press.

Gould, S. (2007). *Punctuated Equilibrium*. Harvard University Press.

Grimm, V., Revilla, E., Berger, U., Jeltsch, F., Mooij, W., Railsback, S., Thulke, H., Weiner, J., Wiegand, T., and DeAnglis, D. (2005). Pattern-Oriented Modeling of Agent-Based Complex System: Lessons from Ecology. *Science*, 310:987–991.

Hamilton, W. (1964). The Genetical Evolution of Social Behaviour, I and II. *Journal of Theoretical Biology*, 7:1–52.

Maynard Smith, J. (1976). Group Selection. *Quarterly Review of Biology*, 51:277–283.

Mayr, E. (1963). *Animal Species and Evolution*. Harvard University Press, Cambridge.

Medawar, P. (1952). *An Unsolved Problem in Biology*. H.K. Lewis, London.

Mitteldorf, J. (2004). Aging Selected for its Own Sake. *Evolutionary Ecology Research*, 7:1–17.

Skulachev, V. (1997). Ageing is a Specific Biological Function Rather Than the Result of a Disorder in Complex Living Systems: Biochemical Evidence in Support of Weismann's Hypothesis. *Biochemistry (Moscow)*, 62:1191–1195.

Weismann, A. (1889). *Essays Upon Heredity and Kindred Biological Problems*. Clarendon Press, Oxford.

Williams, G. (1957). Pleiotropy, Natural Selection, and the Evolution of Senescence. *Evolution*, 11:398–411.

Wilson, D. and Sober, E. (1994). Reintroducing group selection to the human behavioral sciences. *Behavioral and Brain Sciences*, 17:585–654.

Woodberry, O., Korb, K., and Nicholson, A. (2007). A Simulation Study of the Evolution of Aging. *Evolutionary Ecology Research*, 7:1077–1096.

Woodberry, O., Korb, K. B., and Nicholson, A. E. (2005). The Evolution of Aging. In *The Second Australian Conference on Artificial Life (ACAL 2005)*, Sydney, Australia.

Wynne-Edwards, V. (1962). *Animal Dispersion in Relation to Social Behavior*. Oliver and Boyd, Edinburgh.

Initial Modelling of the Alternative Phenotypes Hypothesis

Simon Worgan[1] and Rob Mills[2]

[1]Information: Signals, Images, Systems Research Group, University of Southampton, Southampton, UK, SO17 1BJ
[2]Science and Engineering of Natural Systems Research Group, University of Southampton, Southampton, UK, SO17 1BJ
rmm05r@ecs.soton.ac.uk

Abstract

The alternative phenotype hypothesis contends that multiple phenotypes exist in a single genotype and are expressed by environmental or genetic cues. It further states that these multiple phenotypes will be maintained and improved in a population where the environment is unstable, in spite of the increased cost of this plasticity. In this work we propose a simple computational model to investigate the conditions under which alternative phenotypes become beneficial, and persist over evolutionary timescales. We find that the environment must vary to realise this hypothesis, and that these adaptations not only provide a fitness benefit in highly unstable environments but also continue to arise despite increasing stability and a corresponding gradual decline in fitness.

Introduction

The Alternative Phenotypes Hypothesis (APH), put forward by West-Eberhard (1986, 1989, 2003) puts across the view that phenotypic plasticity in the form of condition-sensitive phenotype expression (*i.e.* alternative phenotypes) is key in a sequence of evolutionary processes that lead to organic novelty, and in turn speciation and higher macroevolutionary events. Although the name may suggest that the evolution of stable alternative phenotypes is most of the operation, the hypothesis in fact covers several events of evolutionary significance.

Key to this hypothesis are alternative phenotypes, which are defined by West-Eberhard as: 'different traits expressed in the same life stage and population, more frequently expressed than traits considered anomalies or mutations, and not simultaneously expressed in the same individual' (2003, p377). In essence alternative phenotypes are when individuals from a single population can develop into different discrete phenotypes, and when this is environmentally cued it is also termed polyphenism. A familiar example is the dimorphism in sexually reproducing species (Lande, 1980).

In brief, the APH suggests that these alternative phenotypes arise when novel traits become established within a population. So long as each alternative phenotype is expressed in an advantageous environment it will be 'buffered' from negative selection; this can be thought of as akin to diversity maintenance techniques employed in evolutionary computing. Potentially, over time each alternative phenotype becomes increasingly distinct, and if modified or localised conditions only favour one particular phenotype this could lead to the emergence of new lineages.

West-Eberhard's hypothesis has many stages and the supporting evidence provided is stronger for some portions than others. One aspect that is not as well substantiated is the specific conditions that afford polyphenic populations a selective advantage.

The APH is based on (extensive) surveys of experimental evidence connected together by verbal arguments. A complex multi-stage theory is sometimes hard to rigorously assess experimentally. Evolutionary simulation models of the processes involved in this hypothesis can assist in constructing more elaborate thought experiments, validating the consistency between stages of the argument, and help identify underlying mechanisms (Barandiaran and Moreno, 2006; Dennett, 1994). In this study we make some early steps towards these goals by modelling one stage of APH; specifically, the fixation of novel traits as viable alternative phenotypes.

The study of organism development is a large and active area of research in evolutionary biology (Wolpert, 2007; Hall, 1998), and is frequently the subject of studies in Artificial Life, *e.g.* Lindenmayer systems (Hornby and Pollack, 2001), ontogeny (Geard and Wiles, 2005), learning (Nolfi and Parisi, 1998) and phenotypic plasticity (Mills and Watson, 2006). However, much of this work focuses on aspects of evolvability arising from developmental representations. Rather than looking at the robustness of specific developmental trajectories themselves, this study focuses on a different aspect of development: the adaptive consequences of environmental influences on the development of several possible alternative phenotypes.

Accordingly, in this paper we propose a computational model to investigate when the evolution of stable, alternative phenotypes provides an adaptive advantage as compared to an evolving population that had no mechanism to support multiple phenotypes.The model does not address the evolu-

tion of these mechanisms themselves, but instead assumes that they are available (at a cost).

As we have already briefly mentioned, we should only expect to see a benefit to polyphenism if each alternative phenotype can find advantageous conditions within the population's environment or niche. To simulate this we vary the environmental conditions over time (although the incorporation of a spatial component with a corresponding temperature or chemical gradient would be one alternative). Thus, we expect to find that in a rapidly varying environment populations maintaining multiple alternative phenotypes would be at an advantage. Conversely, in a static environment we expect that this type of genotype will be selected against, due to the constraints and costs of unnecessarily supporting multiple phenotypes. We investigate both of these scenarios and find evidence to match our expectations supporting the alternative phenotypes hypothesis. Furthermore, the inverse relationship between polyphenism benefit and environmental stability indicates the potential for a more rigorous treatment of the APH.

In the next section, we provide further background on the APH. In Section III, we describe the evolutionary model that is used to support our claims. In Section IV, we describe the experiments performed, and provide results. A discussion is presented in Section V and finally future avenues for research are outlined.

The Alternative Phenotypes Hypothesis

West-Eberhard proposes a conceptual framework to aid the understanding of the role of phenotypic plasticity in providing organic innovation that can ultimately result in speciation. Here we outline the stages that comprise this framework of the alternative phenotypes hypothesis:

1. Prior to alternative phenotypes: the entire population/species exhibits a single phenotype
2. A switch mechanism arises in the population providing the capability for the context-sensitive expression of phenotypes
3. Novel alternative phenotypes evolve and stably persist in the population
4. Each alternative is subject to improvement by natural selection and becomes more specialist to some set of environmental conditions (the context-sensitive expression can prevent the alternative phenotypes from competing with one another)
5. Conditions in some locale may change to favour one alternative over the others; this phenotype will now become exclusively expressed
6. Character release: the genotype no longer has to support all of the alternative phenotypes
7. Accelerated speciation from parent population

A switch mechanism is required to determine which phenotype develops, and this could be allelic (genetic), condition-sensitive (environmental), or a combination of these factors. For the purposes of the APH it is not important which type of switch gives rise to the multiple phenotypes.

After a switch is established each phenotype can be evolved in semi-independence, but this need not lead to reproductive isolation. Since each of the phenotypes will only be selected on when they are expressed, each is buffered from negative selection provided that their expression is cued by the environment in which they are advantageous.

West-Eberhard's ideas on the significance of environmental influence on evolution are generally well accepted (see, *e.g.* Moran, 1992; van Buskirk, 2002; Bourke and Franks, 1991), although light criticism is given with respect to the sparsity of underlying mechanisms (Schlichting, 2003). Comprehensive evidence supports several of the stages contained within the alternative phenotypes hypothesis, including the existence of alternative phenotypes, phenotype fixation, character release and resulting speciation. However some aspects of the APH are not so clear, such as the identity of mechanisms that can provide alternative phenotypes, and the conditions that afford a selective advantage to populations that exhibit these mechanisms. The stages that have the largest evolutionary impact, speciation and macroevolution, can only proceed if conditions exist that are favourable for alternative phenotypes to stably exist. Thus, we limit the scope of this paper to address the investigation of stage 3: the conditions that favour polyphenic populations.

Examples of Alternative Phenotypes in Nature

There are many examples of species that exhibit alternative phenotypes, including the different castes of social insects (Wilson, 1971) with widely varying lifespans across these different castes (Jemielity et al., 2005), reproductive strategies of males (*e.g.* Spinney et al., 2006), and seasonal polyphenisms in aphids (Tauber et al., 1986). We focus on one single example as it provides an excellent illustration of several stages of the alternative phenotypes hypothesis: the buttercup *Rannunculus flammula* as studied by Cook and Johnson (1968). These plants can develop either lanceolate or linear leaves, when terrestrial or immersed in water, respectively. If plant populations are found in either permanently aquatic or permanently terrestrial conditions, they will only develop one type of leaf, but it is possible for a single plant to develop leaves of both types in response to changing conditions (*e.g.* in a lake with 'seasonally fluctuating water levels'). Cook and Johnson also described experiments where plants from monomorphic populations were put in environments unlike their own. When compared with heteromorphic populations, the monomorphic survival ability is reduced. This indicates that where phenotype fixation

has occurred (stage 5), specialisation to that environment is in progress towards character release (stage 6) and speciation (stage 7).

A Model of Alternative Phenotypes in a Varying Environment

In this section we describe the model system used to simulate the evolution by natural selection of a population with the capacity to support environmentally cued alternative phenotypes.

Maintaining multiple phenotypes is only beneficial if each one can become specialised within a different environment. Accordingly, we provide two environments that switch over time, according to a fixed rate, as we believe that these environments are sufficient to motivate the formation of alternative phenotypes. Additionally, the environmental niches should have a significant amount of overlap, since the two resulting phenotypes are still part of the same species and will share many ecological requirements. This overlap indicates that an individual that has specialised to one niche will not be completely unfit in another niche (although it will obviously be outcompeted by a specialist in this second niche).

As such, we find it suitable to extend and modify the framework used by Kashtan and Alon (2005; 2007). These authors are interested in understanding the mechanisms that might explain the modularity observed in biological systems. Following Lipson et al. (2002) they investigate how a varying environment could bring about the evolution of modularity by simulating the evolution of electronic logic circuits towards a target logical function F. They find that if the environment varies between modular functions, modular networks emerge despite an inherent cost: non-modular solutions that use fewer gates do exist. However, these were found to be much more vulnerable to the switching between environments. This is due to the target logic functions that form each environment: the functions share logical substructures and so good solutions to one target require only a small number of changes to satisfy the second target. The smaller, specialist, non-modular solutions would require significant re-working to satisfy the second target. As such, the more expensive modular solutions were favoured in modularly varying environments. We believe that the phenotypic distance between these two modular solutions can be reduced, by the APH, such that it can be reliably traversed within a single generation.

Adapting this work we define a target function F that switches between two states F_1 and F_2, defined as follows:

$$F_1 = AND\left(XOR\left(I_1, I_2\right), XOR\left(I_3, I_4\right)\right) \quad (1)$$

$$F_2 = OR\left(XOR\left(I_1, I_2\right), XOR\left(I_3, I_4\right)\right) \quad (2)$$

with I_1 to I_4 representing Boolean inputs. Note that for 50% of the input patterns these two functions return the same value.

This could be thought of as, for example, a model of seasonal variation in which both photoperiod and temperature influence the development of the *Pieridae* family of butterflies (Shapiro, 1978). Alternatively, it could correspond to the environmental factors of diet, temperature and pheromones which determine the caste of social ants and other insects (Wheeler, 1986). Clearly, the specified target functions represent a substantial simplification of these examples. As the switch between these two functions is periodic we are abstracting away from the complex instability found in biological systems. However, since our representation for individuals has no capacity to learn this periodicity we feel that it is suitable for our purposes.

Given this environmental set-up, a population of individuals of size p, evolves using a generational genetic algorithm (Mitchell, 1996), towards a solution for each function in turn. Each individual G is represented by two sets of integers A and W. Formally these are constrained as follows:

$$A = \{x \in \{0, 1, 2\}\} \quad (3)$$

$$W = \{(y, z)|y, z \in \mathbb{Z}; -j \leq y, z \leq l\} \quad (4)$$

Each gene, g_i drawn from an individual consists of two linked parts x_i and $(y, z)_i$. Each value x represents the response of an individual gene to the environment. If $x = 0$ it will be expressed in either environment, if $x = 1$ it will be expressed in response to F_1, and correspondingly if $x = 2$ it will be expressed in response to F_2. When expressed each gene forms a NAND logic gate with index i, with two inputs connected to the gates denoted by y and z. The range of y and z is limited by the number of genes in the genome, l, and the number of inputs j. Accordingly, if y or $z < 0$ it will denote a connection to a corresponding input. Experiments in this paper use $j = 4$. Finally, if y or $z = 0$ it will denote a connection to the output of the output gate.

The inclusion of a set of environmental switches, A, and a variable length genome, l, represents a significant but necessary departure from Kashtan and Alon (2005). In limiting our changes to the introduction of alternative phenotypes we hope to couple the clear benefits of Kashtan and Alon's work to our own investigation of the APH.

To assess the quality of a given individual we determine its accuracy at solving a particular target logic function, F_1 or F_2. This is performed by applying all 2^4 input patterns, from I_1 to I_4, to the logic circuit defined by the expressed phenotype of an individual. The resulting output is then recorded and the proportion that matches the target function forms the initial fitness of the individual. To limit genetic growth and impose a penalty on the maintenance of alternative phenotypes this initial fitness value, f_{init}, is modified accordingly:

$$f_{final} = f_{init} \times 0.99^{(l-12)} \quad (5)$$

Having assigned a final fitness value to all members of the population, the next generation is constructed. It is composed of three parts: the fittest s individuals, S^+, are copied directly without modification; the set of the least fit s individuals, S^-, is replaced by a copy of S^+; the remaining $(p - 2s)$ individuals inherit their genes from a randomly selected member of S^+, subject to mutation at a rate of m. No crossover mechanism is used. Three events are possible if a mutation occurs:

1. **Gate change** — 50% of all mutations. A random gene, g_i is selected, and one of its inputs, y_i or z_i, is randomly changed with uniform probability within the range $-j \leq y, z \leq l_G$.

2. **Length change** — 30% of all mutations. With equal probability, a new random gate is added to the individuals genotype $(l_G + 1)$ or a random gate is removed $(l_G - 1)$.

3. **Environment sensitivity change** — 20% of all mutations. A random gene, g_i is selected, and its environmental switch, x_i, is changed with equal probability according to $x \in \{0, 1, 2\}$.

When initialised a population of size p is created, each individual within this population has a genome of length l_{init} and for all environmental switches $x = 0$. The simulation is then run for a fixed number of time steps t with the environment switching between the two logical functions F_1 and F_2 at a fixed rate r. In the following simulations each configuration is run 30 times, from which a number of averages are taken. Specifically, we record and plot the mean and maximum final fitness values, f_{final}, at each time step and the mean number of environment specific switches, $x = 1$ or $x = 2$, within each genome. If there is no selection pressure we can calculate the frequency with which environment specific switches will arise:

$$P(p - 2s) \times m \times P(x \in 1, 2) \times 0.2 \qquad (6)$$

With 0.2 representing the fixed probability of and environment sensitivity change. Using the specified experimental parameters we would expect these switches to arise with a 5.2% frequency. We will now detail the specific parameter settings for our experimental work.

Simulated Experiments

Here we provide information regarding the experiments performed using the model described above, and detail their results.

The parameter settings that are common to all experiments described in this section are as follows: m=0.5, p=100, l_{init}=13, s=10, and t=50,000. All experiments were replicated 30 times.

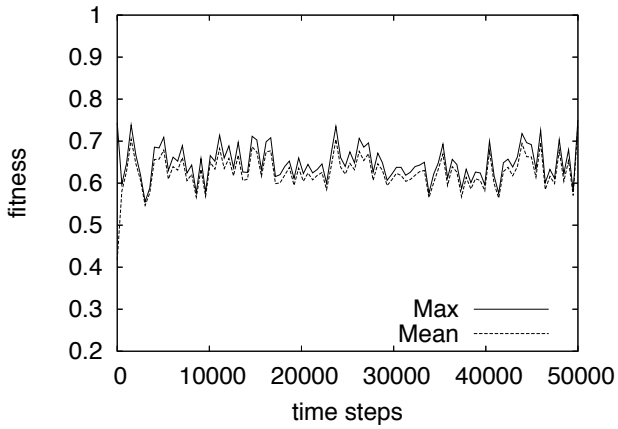


Figure 1: A reproduction of Kashtan and Alon's work with the addition of alternative phenotypes and a variable length genome. These modifications inflict a fitness penalty

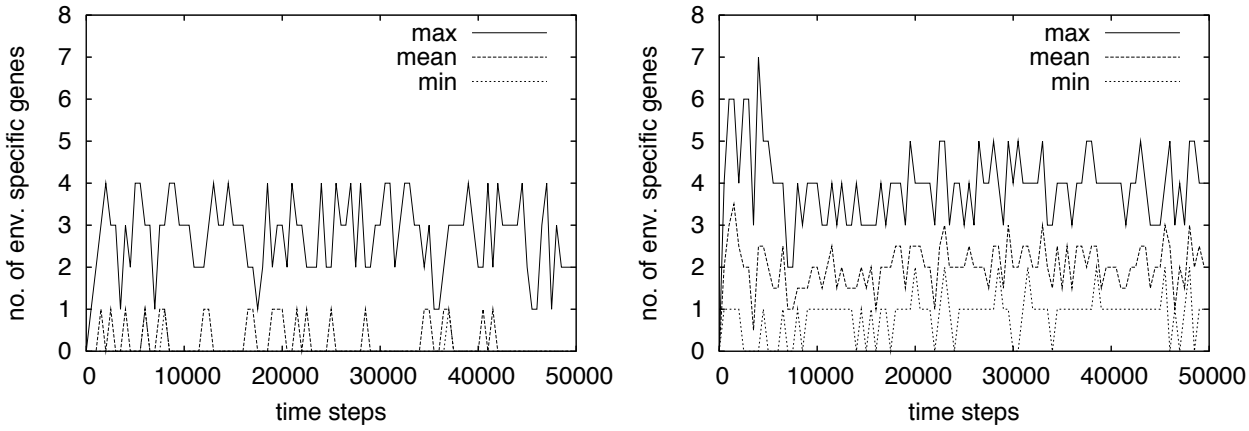
Experiment 1: Introduction of Capacity for Alternative Phenotypes

Initially we test how the introduction of alternative phenotypes modifies Kashtan and Alon's model, by reproducing their experiments with the environment switching at r=20.

Figure 1 shows the mean and maximum fitness of the polyphenic population, averaged across all 30 repeats. We note the mean fitness of the population moves between values of 0.55 and 0.75 throughout the experiment. This is somewhat contrasting to Kashtan and Alon's system, where this population can evolve to an ideal solution for each environment between each switch. This provides us with some indication of how disruptive the inclusion of this plasticity mechanism is.

Experiment 2: Fixed Environment

An expectation stated in section I is that no selective advantage is conferred on a polyphenic population unless the environment varies. Thus we consider the case where environmental conditions are kept fixed throughout the experiment: r is set to t and the target function is set to F_1. Figure 2 (a) shows the number of environment-sensitive loci in these static conditions. For comparison, frame (b) shows the same measure when r=1. We observe that in the static environment the population does not evolve a significant number of environment-sensitive genes. The maximum in the population is between 1 and 4, and the mean occasionally moves above zero due to drift. Alternative phenotypes are evidently not a strong feature of the population under static conditions. This behaviour contrasts with the results from the rapidly varying environment, where the mean number of environment-sensitive loci is around 2, and frequently the entire population has at least one such locus. This experiment was also duplicated with the target function held at F_2,



(a) constant environment

(b) varying environment

Figure 2: The effect of a rapidly switching environment compared to a static environment. In a static environment, frame (a), the population is almost entirely made up of specialists without any environmentally sensitive genes. By comparison, in a rapidly switching environment (frame (b)) alternative phenotypes are the norm

with qualitatively equivalent results.

Experiment 3: Rapidly Varying Environment

To investigate the potential benefits of maintaining alternative phenotypes in rapidly varying environments, we compare a population that has this capability with a population that does not. The environment is set to switch in each generation (*i.e.* r=1). In the monophenic population, all genes in an individual will be expressed regardless of the environment state, and mutations are restricted to modifying the phenotype length or gate assignments.

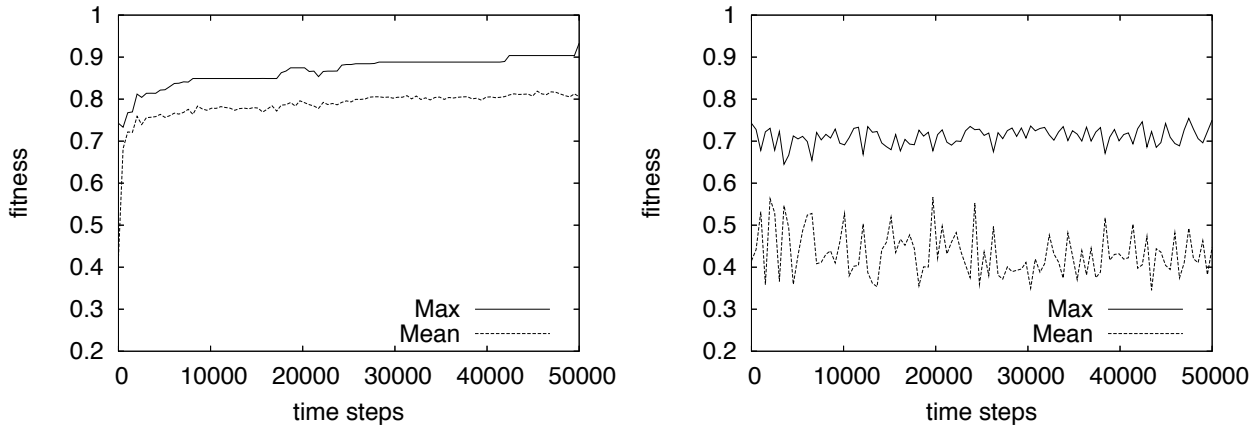
Only the population with polyphenic capability finds high fitness genotypes when in this rapidly varying environment. Figure 3 (a) shows the results from all 30 repeats of this experiment. We see that the population contains individuals of fitness of approximately 0.9, and the population mean is approximately 0.8 by the end of the experiments. This behaviour contrasts with the results shown in Figure 3 (b), which depicts the evolution of a population without polyphenism. The results indicate a population fluctuates around a fitness of 0.75 at best, and a significantly larger gap between the mean and maximum fitness values (the mean fitness fluctuates between 0.375 and 0.55). Consideration of $F1$ and $F2$ shows that for 50% of the input patterns these two functions return the same value, partially accounting for this mean fitness fluctuation.

To verify that environmentally cued loci are responsible, we consider Figure 2 (b). This illustrates the number of loci that will only appear in selected environments, for the polyphenic population in the rapidly varying environment. The mean number is around 2 and some individuals have many more (up to 5 in the long term). For the experiments in a static environment, we saw the results in Figure 2 (a). Here the population is largely made up of individuals with no environmentally cued loci.

The results plotted in Figure 3 are for the most disruptive configuration that we tested, where the switching occurred at every generation. Experiments with r=2, r=5, and r=10 behave qualitatively similarly, although the highest fitness discovered degrades with increasing switching period. When we slow the environment switching to r=20, the behaviour moves towards that of the monophenic population. However, the polyphenic population mean fitness is much closer to the maximum fitness found than for the monophenic population.

Discussion

The results from experiments performed provide a perception on some of the conditions that affect the adaptive merit of maintaining alternative phenotypes. Experiment 1 reveals the cost of the plasticity that has been incorporated into the model. Experiment 3 demonstrates a clear advantage to populations maintaining alternative phenotypes when in an unstable environment, in spite of these additional costs. The polyphenic population has significantly higher fitness than the monophenic population in the most rapidly varying environments. Additionally, the mean and maximum fitness of the polyphenic population are a lot closer indicating a more stable region of adaptive space had been reached and the population had converged. When the environmental switching rate is reduced, the fitness improvement become less pronounced. However, when the fitness improvement is negligible at r=20, we still see the persistence of environment-sensitive genes (see Figure 4). There comes a point where maintaining alternative phenotypes is no longer viable as is shown in the limit case in experiment 2. This test confirmed



(a) alternative phenotypes available

(b) only a single phenotype per genotype

Figure 3: A population with the capacity to support alternative phenotypes compared against a population that cannot. The population in frame (a) is far more successful in handling the rapidly varying environment than the population in frame (b)

that polyphenism does not evolve in a fixed environment. Although we note that some drift is possible in the number of environment-sensitive genes: when stably in environment 1, an environment switch set to 1 will have the same effect on the phenotypic expression as that locus being set to 0 (and equivalently for environment 2). It is also worth considering the selection pressures on genes set to be expressed in environment 2 when it is fixed in environment 1. Although any such genes will not be expressed and thus cannot have a negative impact on the phenotype, there is nevertheless a cost to maintaining this gene. There is no pressure to maintain such a gene, and so we should expect these genes to be purged. Figure 2 does not show the values independently, but we can report that in experiment 2 all environmentally-cued genes match their respective environments (barring for a single generation in one anomalous result).

When considering the conditions in experiment 3, the population initially contains genotypes with low fitness to either environment. However, selection favours genotypes that express phenotypic traits that can contribute to high fitness in both environments, such that the population will move to a portion of the fitness landscape that overlaps. This can only be the case when the environments share a significant portion of their structure — and the target functions chosen by Kashtan and Alon have exactly this property.

There are sets of genes that co-occur due to the environmental cuing, and these sets are buffered from one another, forming an interesting parallel to evolutionary computing. A problem often faced in evolutionary algorithms, known as premature convergence, is when population diversity is lost rapidly. This can lead to the population converging on low fitness optima, and attempts to alleviate this are known as diversity maintenance (see Singh and Deb, 2006). These typically restrict the competition between individuals such that portions of the population can focus on different parts of the fitness landscape. The condition-sensitive portions of a genome could potentially inspire a new diversity maintenance technique. Because one set of genes will only be expressed when in an environment that is advantageous for that particular phenotype, direct competition between alternatives is avoided.

A study into evolvability by Earl and Deem (2004) uses a 'DNA swap' mechanism that makes large, but non-random genetic changes in addition to small-scale changes by mutation. The DNA swap involves the substitution of genetic material for a particular genetic subdomain from a pool of low-energy alternatives for that subdomain. This can be considered as a form of diversity maintenance: the pools contain many different options to be swapped in and the alternative selected for the current environment, restricting competition being restricted to the subdomain (contrast this with the buffering provided by environmentally sensitive gene expression to a subset of genetic material in a single individual). They also investigate the suitability of each mechanism across a range of rates of environmental change, and find that large-scale variation is favoured increasingly in rapidly varying environments, further supporting the position that mutation alone is inadequate to cope with unstable conditions.

The exploration of a buffering-driven diversity maintenance mechanism for evolutionary computing is outside the scope of the current body of work, but considering condition sensitive switching in this light may help us to better understand the types of environment that it may prove advantageous within.

In our model all genes have the potential to be conditional on environmental cues, and this in principle allows several different configurations:

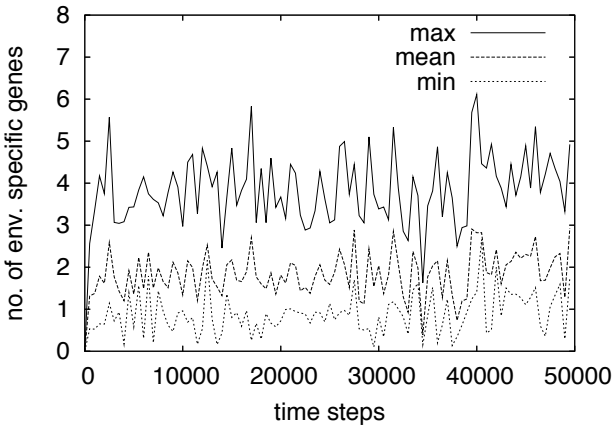


Figure 4: A population with a switching rate of 20 gains no fitness benefit from alternative phenotypes, see Figure 1. However, switching alone is sufficient for the persistence of alternative phenotypes within the population

- specialists: ideally adapted to a single environment; no condition-sensitive genes present
- modular generalists: suited to relatively slowly switching environment, where there is a region in the genetic space that has solutions for each environment nearby
- generalists with alternative phenotypes: suited to a faster-switching environment, where fit configurations for each niche overlap somewhat; consequently many genes are expressed in all environments and a small proportion of genes are environmentally sensitive
- polyphenic specialists: best suited to an environment that varies rapidly, but there is significant distance between fit phenotypes for each set of conditions; consequently an almost independent phenotype is supported for each environment

However we only encounter the first and third of these possibilities with the experimental conditions performed so far. One can imagine a line of inquiry that explores the conditions sufficient to give rise to the currently unseen genotype structures, and the potential trajectories between each of these types.

There is a heavy penalty associated with the number of genes required to code for two fit specialists, so it is unsurprising that they don't appear in the experiments performed to date. However a pair of target functions that varied in a non-modular way (*i.e.* with very little overlap) may give rise to such genotypes. This would still require two sets of genes to be co-adapted, and the isolation provided by the environment-sensitivity could maintain the correct diversity to lead to such adaptation.

Since selection in our model preserves the fittest individuals from each generation, one might expect the highest fitness to increase, or at least stay constant. We do observe fitness decreases in Figure 1, in contradiction with this expectation. This is because the high fitness solutions that are preserved from one environment to the next are not necessarily of high fitness in this new environment. They consequently may not be preserved long enough to return to high fitness in the second environment. However as shown in Figure 3, the inclusion of environmentally-switched gene expression goes a long way towards mitigating this problem.

We will now outline a number of aspects for future research. Initially, we would like to understand the disruption inflicted upon polyphenic populations, and the factors to which it can be attributed. We also wish to perform similar experiments to those reported here, but with different sets of target functions. As discussed above, this includes considering environments that do not have modular overlap with the aim of evolving more independence in the alternative phenotypes of one individual.

There are also avenues that involve extensions to the current framework. This includes connecting together additional stages of the APH. For example, investigating how long character release might take would be possible by changing the schedule of environmental conditions experienced by a population with established alternative phenotypes. Additionally, it would be valuable to test the hypothesis that an environmentally cued alternative phenotype can lie dormant for many generations without being expressed. The identification of candidate switch mechanisms would require significant extension to the model. We could investigate what conditions might enable useful cuing mechanisms to arise, using the framework that we have established to test the plausibility of a particular switch mechanism. Finally, we could employ a slightly different experimental set up to study direct competition between polyphenic and monophenic populations instead of comparing the populations in isolation.

As far as we know, we have presented the first individual-based simulation model of a portion of the alternative phenotypes hypothesis, illustrating some capabilities and limitations of polyphenism under different abstract environmental conditions. The results that we have obtained so far, whilst modest, provide support for the later stages of the APH that require the stable existence of alternative phenotypes to proceed. We feel that there are many aspects that could be better understood when using simulation models to enrich thought experiments (Di Paolo et al., 2000). In developing this initial model, we have identified abstract environmental conditions that could account for the previously observed phenomena within the APH. We believe that the conceptualisation provided by this approach could ultimately unify the current experimental evidence into an increasingly rigorous underlying framework.

References

Barandiaran, X. and Moreno, A. (2006). Alife models as epistemic artefacts. In Rocha, L. M., editor, *Procs ALifeX*, pages 513–519.

Bourke, A. F. and Franks, N. R. (1991). Alternative adaptations, sympatric speciation and the evolution of parasitic, inquiline ants. *Biological Journal of the Linnean Society*, 43(3):157–178.

Cook, S. A. and Johnson, M. P. (1968). Adaptation to heterogeneous environments. I. variation in heterophylly in ranunculus flammula l. *Evolution*, 22(3):496–516.

Dennett, D. C. (1994). Artificial life as philosophy. *Artificial Life*, 1(3):291–292.

Di Paolo, E., Noble, J., and Bullock, S. (2000). Simulation models as opaque thought experiments. In *procs Articial Life VII*, pages 497–506, Cambridge, MA. MIT Press.

Earl, D. J. and Deem, M. W. (2004). Evolvability is a selectable trait. *PNAS*, 101(32):11531–11536.

Geard, N. L. and Wiles, J. (2005). A gene network model for developing cell lineages. *Artificial Life*, 11(3):249–268.

Hall, B. K. (1998). *Evolutionary developmental biology*. Chapman & Hall, London, 2nd edition.

Hornby, G. S. and Pollack, J. B. (2001). Body-brain co-evolution using L-systems as a generative encoding. In Goodman, E. D., editor, *Procs GECCO*, pages 868–875. Morgan Kaufmann.

Jemielity, S., Chapuisat, M., Parker, J. D., and Keller, L. (2005). Long live the queen: studying aging in social insects. *AGE*, 27(3):241–248.

Kashtan, N. and Alon, U. (2005). Spontaneous evolution of modularity and network motifs. *PNAS*, 102:13773–13778.

Kashtan, N., Noor, E., and Alon, U. (2007). Varying environments can speed up evolution. *PNAS*, 104(34):13711–13716.

Lande, R. (1980). Sexual dimorphism, sexual selection, and adaptation in polygenic characters. *Evolution*, 34(2):292–305.

Lipson, H., Pollack, J. B., and Suh, N. P. (2002). On the origin of modular variation. *Evolution*, 56(8):1549–1556.

Mills, R. and Watson, R. A. (2006). On crossing fitness valleys with the baldwin effect. In Rocha, L. M., editor, *Procs ALifeX*, pages 493–499.

Mitchell, M. (1996). *An Introduction to Genetic Algorithms*. MIT Press.

Moran, N. A. (1992). The evolutionary maintenance of alternative phenotypes. *American Naturalist*, 139(5):971–989.

Nolfi, S. and Parisi, D. (1998). Neural networks in an artificial life perspective. In *Procs ICANN*, pages 733–737.

Schlichting, C. D. (2003). Environment rules. *Trends in Ecology and Evolution*, 18(10):496–497.

Shapiro, A. M. (1978). The evolutionary significance of redundancy and variability in phenotypic-induction mechanisms of pierid butterflies (lepidoptera). *Psyche*, 85(2–3):275–283.

Singh, G. and Deb, K. (2006). Comparison of multi-modal optimization algorithms based on evolutionary algorithms. In *Procs GECCO*, pages 1305–1312.

Spinney, L. H., Bentley, G. E., and Hau, M. (2006). Endocrine correlates of alternative phenotypes in the white-throated sparrow (zonotrichia albicollis). *Hormones and Behavior*, 50(5):762–771.

Tauber, M. J., Tauber, C. A., and Masaki, S. (1986). *Seasonal Adaptations of Insects*. Oxford University Press.

van Buskirk, J. (2002). A comparative test of the adaptive plasticity hypothesis: Relationships between habitat and phenotype in anuran larvae. *American Naturalist*, 160(1):87–102.

West-Eberhard, M. (1986). Alternative adaptations, speciation, and phylogeny (a review). *PNAS*, 83(5):1388–1392.

West-Eberhard, M. (1989). Phenotypic plasticity and the origins of diversity. *Annual Review of Ecological Systems*, 20:249–278.

West-Eberhard, M. J. (2003). *Developmental Plasticity and Evolution*. Oxford University Press.

Wheeler, D. E. (1986). Developmental and physiological determinants of caste in social hymenoptera: Evolutionary implications. *American Naturalist*, 128(1):13–34.

Wilson, E. O. (1971). *The Insect Societies*. Harvard University Press, Cambridge, MA.

Wolpert, L. (2007). *Principles of Development*. Oxford University Press, 3rd edition.

Passive and Driven Trends in the Evolution of Complexity

Larry Yaeger[1], Virgil Griffith[1,2], and Olaf Sporns[3]

[1]School of Informatics and [3]Department of Psychological and Brain Sciences
Indiana University, Bloomington, IN 47408
[2]Computation & Neural Systems, California Institute of Technology, Pasadena, CA 91125
larryy@indiana.edu

Abstract

The nature and source of evolutionary trends in complexity is difficult to assess from the fossil record, and the *driven vs. passive* nature of such trends has been debated for decades. There are also questions about how effectively artificial life software can evolve increasing levels of complexity. We extend our previous work demonstrating an evolutionary increase in an information theoretic measure of neural complexity in an artificial life system (Polyworld), and introduce a new technique for distinguishing driven from passive trends in complexity. Our experiments show that evolution can and does select for complexity increases in a driven fashion, in some circumstances, but under other conditions it can also select for complexity stability. It is suggested that the evolution of complexity is entirely driven—just not in a single direction—at the scale of species. This leaves open the question of evolutionary trends at larger scales.

Introduction

The existence of an evolutionary trend towards greater complexity is undeniable, whether one measures complexity by organism size (Cope, 1871), distinct cell types (Bonner, 1988; Valentine et al., 1994), morphology (Thomas and Reif, 1993; McShea, 1993), or ecological webs of interaction (Knoll and Bambach, 2000). Historically, it has often been suggested that such growth is the result of an evolutionary bias towards forms and functions of greater complexity, and a great variety of rationales has been offered for why this should be the case; e.g., (Rensch, 1960a,b; Waddington, 1969; Saunders and Ho, 1976; Kimura, 1983; Katz, 1987; Bonner, 1988; Arthur, 1994; Huynen, 1996; Newman and Engelhardt, 1998); see McShea (1991) and Carroll (2001) for reviews. However, Maynard Smith (1970), Raup et al. (1973), Gould (1994), and others have questioned whether that growth has been the outcome of natural selection or simply, in Maynard Smith's words, the "obvious and uninteresting explanation" of a sort of random walk away from an immutable barrier of simplicity at the lower extreme—a growth in variance relative to the necessarily low complexity at the origin of life.

Bedau et al. (1997) and Rechsteiner and Bedau (1999) provide some evidence of an increasing and accelerating "evolutionary activity" in biological systems not yet demonstrated in artificial life models. However, other attempts to characterize complexity trends in the fossil record have produced mixed results at best (McShea, 1996; Heylighen, 2000; Carroll, 2001), leaving us with no clear picture of the influence of natural selection on complexity. McShea (1994, 1996, 2001, 2005) has, over the years, attempted to clarify (and, where possible, empirically address) the debate, by identifying distinct classes of complexity and, importantly, by distinguishing between "driven" trends, in which evolution actively selects for complexity, and "passive" trends, in which increases in complexity are due simply to asymmetric random drift.

Simple computational models of branching species and clade lineages in simple numerical parameter spaces (arbitrary values standing in for complexity, size, or the like) have been used to investigate this distinction between driven and passive evolutionary trends (Raup et al., 1973; Raup and Gould, 1974), however it is not always possible to distinguish a passive system from a weakly driven system (McShea, 1994). Furthermore, the anagenetic component of some of these models, while intended, by definition, to address within-lineage change, is equivalent to branching lineages that effectively compete with one another for parameter space, by requiring branched, descendant lines to replace ancestral lines. The typical assumption of equal extinction rates across all scales may also unintentionally color the results from these models. It is common wisdom, for example, that a driven system necessarily implies an increase in the minimum value of whatever parameter is being used to distinguish taxonomic branches (Wagner, 1996; McShea, 2001; Carroll, 2001), yet an evolutionary system in which fitness at smaller scales is independent of fitness at larger scales could possess a drive towards larger scales without eliminating or even disadvantaging organisms at the lower end of the spectrum. Indeed, McShea (1994) acknowledges a dramatically lower rate of growth in this minimum in a purely cladogenetic model compared to a mixed anagenetic and cladogenetic model. One of us [LY] is currently investigating the effects of the uniform extinction rate assumption.

Given the difficulty and ambiguity one encounters when

attempting to answer questions about the evolution of complexity from paleontological data or simple branching models, it makes sense to turn to computer models of evolution to address these questions. Turney (1999, 2000) has used a simple evolutionary model to suggest that increasing evolvability is central to progress in evolution and predicts an accelerating increase in biological systems that might correlate with complexity growth. Adami et al. (2000); Adami (2002) has defined complexity as the information that an organism's genome encodes about its environment and used Avida to show that asexual agents in a fixed, single niche always evolve towards greater complexity of this narrowly defined type.

Modern compute power and artificial life methods allow us to rewind the "tape of life", as Gould (1989) put it, and let history unfold again and again under slightly or dramatically different influences. Here we use a method for "replaying the tape" that is substantially beyond the perturbed playback Gould envisaged, demonstrating a method for carrying out parallel simulations in which natural selection either does or does not play a part, yet with all other population and genetical statistics being held constant. This is similar in spirit to the extreme "behavioral noise" null model used by Bedau (1995) and subsequent "neutral shadow models" (Bedau et al., 1998). Being able to effectively turn natural selection on and off in this fashion allows us to tease apart and distinguish evolutionarily driven trends from passive trends in a formal, quantitative fashion.

The trend we investigate is a particular information-theoretic measure of complexity (Tononi et al., 1994; Lungarella et al., 2005), **C**, for the neural dynamics of artificial agents in an evolving computational ecology, Polyworld (Yaeger, 1994). In previous work (Yaeger and Sporns, 2006) we demonstrated an increasing trend in **C** in the agents of Polyworld over evolutionary time scales, and were able to relate these increases to increasing structural elaboration of the agents' neural network architectures and an increase in the learning rates employed at the Hebbian synapses in these networks. We did not, however, address the evolutionary source of these increases, or whether that source should be construed as driven or passive, in the McShea (1996) sense. Here we use a novel technique that allows us to make such a distinction, and discover that, at least at the scale of single species and ecological niches, evolution of complexity is always driven, but, interestingly, not always driven in the same direction.

Tools and Techniques

Polyworld

Polyworld (Yaeger, 1994) is an evolutionary model of an ecology populated with haploid agents, each with a suite of primitive behaviors (move, turn, eat, mate, attack, light, focus) under continuous control of an Artificial Neural Network (ANN) consisting of summing and squashing neurons and synapses that adapt via Hebbian learning. The architecture of the ANN is encoded in the organism's genome, expressed as a number of neural groups of excitatory and inhibitory neurons, with genetically determined synaptic connection densities, topologies, and learning rates. Input to the ANN consists of pixels from a rendering of the scene from each agent's point of view, like light falling on a retina. Though agent morphologies are simple and static, agents interact with the world and each other in fairly complex ways, as they replenish energy by seeking out and consuming food and by attacking, killing, and eating other agents. They reproduce by simultaneous expression of a mating behavior by two collocated agents.

Agent population is normally bounded above and below, but unlike the simulations discussed in (Yaeger and Sporns, 2006), there is no "smite" function invoked at maximum population, which the authors felt risked introducing a bias associated with the simulator's *ad hoc* heuristic fitness function. Nor does a minimum population any longer invoke a steady-state GA, which also would necessarily depend upon that fitness function. Instead, as the population grows towards the upper bound, the amount of energy depleted by all agent behaviors, including neural activity, is increased in a continuous fashion. Conversely, as the population drops towards the minimum, energy depletion is decreased, and agent lifespans may be artificially extended. This does not guarantee a viable population (one that sustains its numbers through reproduction), since unsuccessful, unfit agents may be all that remain by the time a failing population bottoms out, but it does provide very effective population control without denying births to agents capable and "desirous" of doing so, while simultaneously eliminating any possible effects of the now purely informative heuristic fitness function.

Complexity

For our purposes, complexity, **C**, is computed using a new C++ implementation based on the methods of (Lungarella et al., 2005) for approximating the information-theoretic measure of complexity originally developed by (Tononi et al., 1994). Though non-trivial to derive and implement, the intuition behind **C** is straightforward: Cooperation amongst various elements of a network, called *integration* and measured by a multivariate extension to mutual information, increases network complexity, to a point. But specialization of network subunits, called *segregation* and measured by the difference between maximum and actual integration at different scales, also increases network complexity. Maximal complexity is achieved in networks that optimally trade off the opposing tensions between integration and segregation—between cooperation and specialization—and maximize both to the extent possible. The original mea-

sure of complexity is given by:

$$C_N(X) = \sum_{k=1}^{n} [\, \langle H(X_j^k) \rangle - \frac{k}{n} H(X) \,] \qquad (1)$$

where $H(X)$ is the Shannon entropy of the entire system of n variables, k is the size of a subset of variables, and j indicates that the ensemble average $\langle H(X_j^k) \rangle$ is to be taken over all $n!/(k!(n-k)!)$ combinations of k variables. The simplified approximation we use was introduced in (Tononi et al., 1998) and explored computationally in (Sporns et al., 2000):

$$C(X) = H(X) - \sum_{x_i \in X} H(x_i | X - x_i) \qquad (2)$$

where $H(X)$ is the entropy of the entire system and the $H(x_i|X - x_i)$ terms are the conditional entropy of each of the variables x_i given the entropy of the rest of the system.

Natural Selection *vs.* Random Drift

While probably impossible to eliminate natural selection from an evolving biological ecology, artificial ecologies are more flexible. In order to distinguish between evolutionarily driven and passive trends, we designed a mode for running our simulator in which natural selection had effectively been eliminated, yet which could be compared directly with runs in which natural selection operated normally. This was accomplished by implementing a new "lockstep" mode of operation in Polyworld. First a simulation is run in the system's normal, natural-selection mode of operation. During this natural-selection run, the birth and death of every agent is recorded (along with the usual statistics, brain states, etc.). Then the simulator is run in the lockstep mode, starting from the same initial conditions as the natural-selection run and using the birth and death data recorded during the natural-selection run. No "natural" births or deaths are allowed during a lockstep run. Instead, every time a birth occurred in the original natural-selection run, a birth is forced to occur in the lockstep run, only instead of being produced by the original parents, the birth is produced by two agents chosen at random from the population. Similarly, whenever a death occurred in the natural-selection run, a death is forced to occur in the lockstep run, only instead of the original agent dying, a random agent is killed and removed from the population.

By so doing, population statistics are forced to be identical between the paired natural-selection and lockstep runs. As a result, the genetical statistics—number of crossover and mutation operations—are forced to be comparable in the paired runs. Note that since crossover and mutation are applied to different genomes and since the number of crossover points and the mutation rate are themselves embedded in these genomes (Yaeger, 1994), these genetic operations are only statistically comparable between paired runs, not identical. Similarly, the "life experiences" of a given agent—its trajectory through the world and the inputs to its visual system—are only comparable statistically between paired runs. Since the agents' life experiences do impact the values of neural complexity we compute, this could produce extraneous differences between paired runs, but we do not expect this to have any consistent, measurable influence. The controlling statistics, such as the entropy and mutual information in the visual inputs, are comparable between paired runs, so we expect computed complexity values to be similarly comparable. While it would be possible to record number of crossover points, mutation rates, agent trajectories, and even sensory inputs during the natural-selection runs and play them back during the lockstep runs, we do not believe this would alter the relevant statistics or the measured outcomes in a substantive manner, and therefore have not made any such attempts. One could also argue that since complexity is affected by agent behaviors and their resulting sensory inputs, agents in lockstep runs must be able to control their actions in order to obtain valid measures of their neural complexity.

The end result of these machinations is that gene states are subject to natural selection, based on the evolutionary viability—the fitness—of the agents' behaviors, in the natural-selection runs. While gene states are subject only to the same degree of variation, with no evolutionary fitness consequences or effects, in the lockstep runs. Additionally, population statistics are identical and sensory input statistics are comparable between paired runs.

Simulations and Data Acquisition

A set of 10 paired simulations, differing only in initial random number seed, were run in natural-selection and lockstep modes; i.e., 20 simulations in all. Each was run for 30,000 time steps. As Polyworld is continuous rather than generational, determining the number of generations is non-trivial. In the past estimates have been known only to fall within a large range. A low estimate based on average lifespan (about 300 time steps) would be 100 generations. A high estimate based on the minimum age of fecundity (25 time steps) would be 1,200 generations. A newly implemented lineage tracer produces a more accurate estimate of about 400 generations.

The world is seeded with a uniform population of agents that have the minimum number of neural groups and nearly minimal neuron and synapse counts. While predisposed to some potentially useful behaviors, such as running towards green (food) and away from red (aggressive agent behaviors; see (Yaeger, 1994) for details on color use in Polyworld), these seed organisms are not a viable species. That is, unless they evolve they cannot sustain their numbers through reproduction and will gradually die out.

During simulations, the activation of every neuron in the

brain of every agent is recorded at every time step. These brain function recordings are grouped into arbitrary (here, 1,000) time step bins, for all agents that died during the specified interval. Utility programs are then used to calculate the complexity, **C**, of the neural dynamics of every agent's complete lifespan (hence the requirement for the agent's death). We then compute mean complexities for these binned populations of agents as a function of time. Finally, we compute means and standard deviations of the population means for the multiple natural-selection and lockstep simulations as a function of time to study general evolutionary trends in complexity.

Complexity can be calculated across all neurons, just the input neurons, or just the "processing" neurons (all neurons except inputs). All complexities presented here are based on processing neurons. (In general, there is little difference in complexity trends between all neurons and processing neurons.) Complexity varies as evolution produces changes in the parts of the genome that specify the neural architecture.

Results and Discussion

Figure 1 shows complexity versus time for the previously described series of 10 paired driven (natural-selection) and passive (lockstep) simulations. The lighter lines depict population means from individual runs. The heavier lines depict means of all runs of a particular type (driven or passive). Data is presented in this individual-plus-mean fashion, rather than mean-plus-standard-error fashion, to give a better feel for the nature of the variance between runs, and to identify some interesting events in a small number of the runs (discussed later). Plotted beneath the complexity lines is a single dotted line that measures a paired or dependent Student's T-test computed on the same time interval as the complexity data are computed and plotted. (A dependent test is used because these are paired runs with common initial conditions and enforced common population and genetic statistics.) Where this line is above the horizontal T-critical (T*) line, this standard measure of statistical significance rejects the null hypothesis with $p < 0.05$; at or below the T-critical line the null hypothesis cannot be rejected (at least not as reliably). Given 10 pairs of runs, the number of degrees of freedom is 9, and T-critical is 1.833.

The first thing to note is a statistically significant faster growth rate in complexity in the driven runs than in the passive runs during approximately the first 4,000 time steps. Evolution is clearly selecting for an increase in complexity during this early time period. This makes sense intuitively since the seed population is known to be non-viable and must evolve or die out. Increases in complexity during this period are of a distinct evolutionary advantage, producing descendant populations that are more capable of thriving in this particular environment than their ancestors. During this time period, the evolution of complexity is clearly driven, with a bias towards increasing complexity.

The next thing to note is the early plateauing of complexity in the driven runs, allowing the randomly drifting complexity of the passive runs to catch and surpass them by around t=7,000. This is the result of evolution having found a solution that is "good enough", and the concomitant spread of the genes producing this solution throughout the population. Seven out of the 10 natural selection simulations remain relatively stable around this modestly complex solution once it is found. The intuition here is that any change away from this "good enough" solution is likely to be detrimental, hence evolution selects for stability. Note that this actively suppresses genetic drift and, indeed, a statistically significant difference between driven and passive runs, with passive complexity now being the larger of the two, is maintained from about t=8,000 to the end of the runs at t=30,000.

There is also a consistent, but less interesting, plateauing of complexity in the passive runs. This is due solely to the individual bits of the underlying genome approaching a state of approximately 50% on, 50% off. Effectively, the random walk has maximized variance as much as it can given the model parameters. Though generally higher than the driven mean, complexities in the passive/random model are nowhere near the maximum obtainable with the full range of gene values (as observed in the complexity-as-fitness-function experiments discussed below); they just correspond to the range of complexities representable by the genome with an even mix of on and off bits. Such larger values of complexity are potentially meaningful, but do not confer any evolutionary advantage on agents in these lockstep runs.

Finally, if one looks carefully at driven complexity for the individual runs, three (of the 10) runs make secondary transitions upward in complexity between t=20,000 and the end of the run, coincidentally reaching about the same level of complexity as the passive runs. In this subset of runs, apparently a new or improved behavior emerges late in the simulation, and the genes producing this behavior spread throughout the population fairly rapidly. Despite the simplicity of the world design used for these experiments, multiple viable, competing solutions have emerged and it is always possible that more solutions would emerge given time. Importantly, for the future, it appears that this complexity measure provides a useful tool for ascertaining the onset of new, improved strategies, including speciation events, as well as a quantitative tool for assessing the neural changes that produced the new strategies.

Two of us [LY,OS] independently realized some years ago that if we were in a position to measure neural complexity in an artificial system, it might make sense to accelerate the evolution of complexity by using it as an explicit fitness function. Although not elaborated upon here, preliminary experiments to this end have been carried out. With the same values for the parameters controlling neural architecture, using complexity as a fitness function pushes mean complexity up to around 0.9, roughly three times the levels

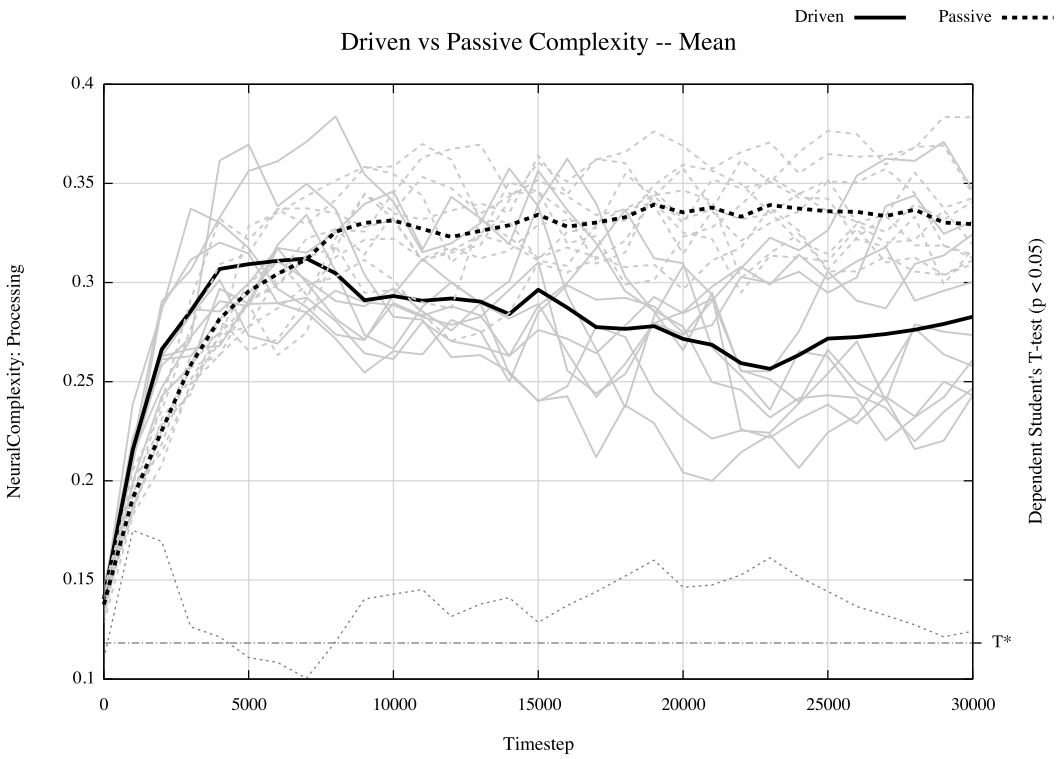


Figure 1: Driven and passive complexity *vs*. time. Light solid lines show population mean complexity for each driven, natural-selection run. Light dashed lines show population mean complexity for each passive, lockstep run. Heavy lines show means of all ten runs for corresponding line style. Light dotted line at bottom shows dependent Student's T-test relative to horizontal T-critical line (labeled T*) for $p > 0.05$.

obtained by this series of driven (0.275) and passive (0.33) runs. However, selecting purely for complexity consistently produces stereotypic spinning behaviors by the agents, that would not be of much value under natural selection, and we suspect are the result of a maximization of entropy and mutual information in the sensory inputs. This differs from the direct coupling between complexity and behavior found in (Sporns and Lungarella, 2006), probably due to differences in the range of possible behaviors and the nature of the sensory inputs in the simulation environments used for these two studies.

Adami et al. (2000); Adami (2002) has defined a measure of genomic complexity in terms of how much information a species' genome encodes about its environment, based on the cross-population entropy at each genomic site (here, individual bits). The measure has acknowledged constraints—it only applies to a single species in a single, static niche, thus failing to capture issues related to biodiversity, environmental variability, or broader ecologies. More generally, we suspect it may be a better measure of genomic consistency or specialization of a species than of complexity, but there is no question that the aggregate stability of gene sites in a species' population is an important measure of the success of that species at encoding information about its environment in its genome. Since our current series of simulations are deliberately simple and may probably best be thought of as the evolution of a single (or at least highly related) species in a single, static niche, we decided to investigate the evolution of this genomic consistency.

Figure 2 shows the evolution of Genomic Consistency (GC) over time. Since the world is seeded with a uniform population, GC is initially extremely large, as the measure effectively assumes the current genetic structure is an evolved response to the environment and perfect uniformity is maximally consistent. (However, we do not feel this should be seen as maximally complex, hence our renaming of Adami's genomic complexity.) Accordingly, the vertical extent of the graph has been truncated in order to focus on the more interesting results. The main observation is the dramatic difference between driven and passive runs. The passive runs produce extensive random gene edits, thus minimizing GC. The driven runs demonstrate a larger, stabilized GC across the population over time due to natural selection for specific traits that increase the evolutionary fitness of the agents. There is a hint of a modest upward trend in GC after it bottoms out around t=12,000, suggesting a possible continued incorporation of information about the environment in the genome of these agents, but so far no attempt has been

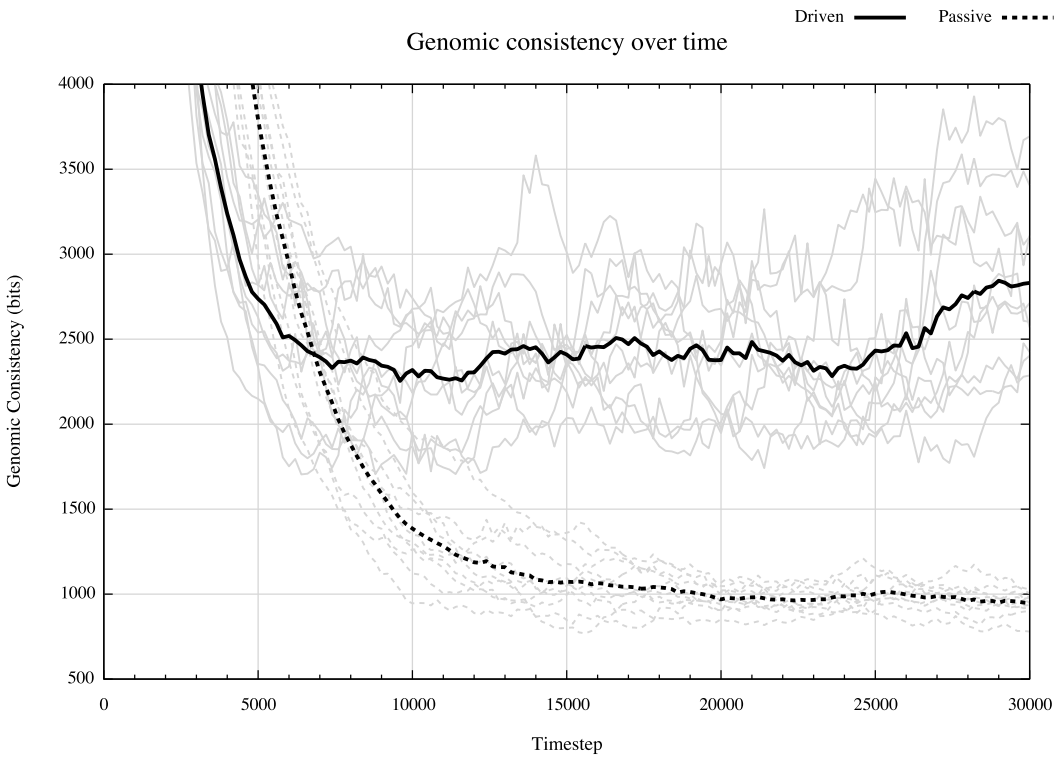


Figure 2: Genomic consistency *vs*. time. Light solid lines are for each driven, natural-selection run. Light dashed lines are for each passive, lockstep run. Heavy lines show means of all ten runs for corresponding line style.

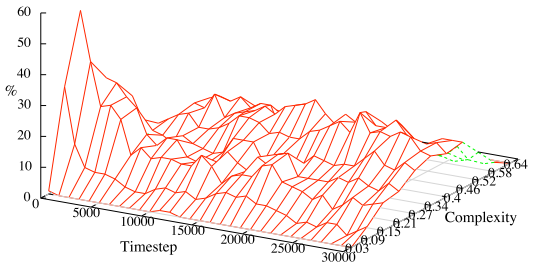
made to establish statistical significance.

Another way to draw some understanding from this series of driven *vs*. passive runs is to look at the distribution of complexity in the populations over time. Figure 3 shows time-series histograms of complexity throughout the population for four sample simulations—two passive, two driven. The two passive runs show a generalized increase in variance, due to the diffusive random walk away from the low complexity of the seed agents. The two driven runs show a more peaked distribution around the complexity values attained by the viable populations emerging as a result of natural selection. Figure 3(c) is representative of the majority of the driven runs, showing a shift towards a modest level of complexity. Figure 3(d) is representative of the small number of driven runs in which a secondary transition to a different behavior and higher level of complexity emerged late in the simulation.

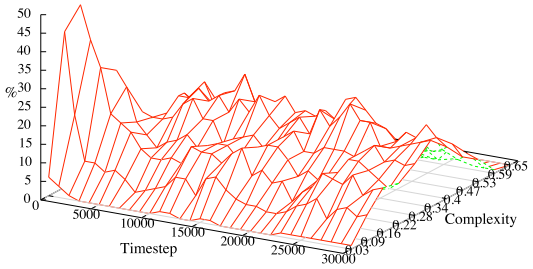
Conclusions

We have demonstrated a technique for directly comparing and assessing neural complexity growth in equivalent driven and passive systems. Using this technique we have shown evolutionary selection for *increased* complexity, in a "driven" fashion, as well as selection for complexity *stability*. Though we have not demonstrated it here, there is little doubt that a system in which the cost of neural complexity exceeded its value would result in a driven *reduction* in complexity, the way dark dwelling organisms in a cave have been known to give up their eyes. This paints a complex picture of evolutionary selection for increasing, stable, and decreasing complexity, none of which corresponds to a purely "passive" mechanism of complexity change. At this scale, evolution is entirely driven, with changes in complexity always being selected for or against. Scale, however, is very important to this discussion.

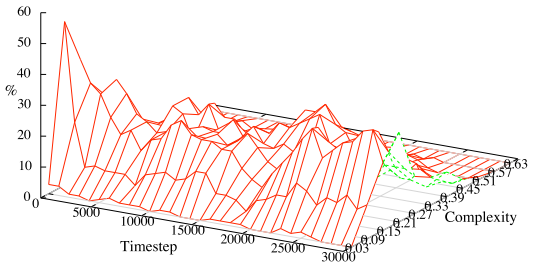
Gould (1996) and Dawkins (1997) have argued strongly for passive and driven evolutionary trends, respectively. However, much of the disconnect between them seems to be precisely an issue of scale. Dawkins is unquestionably correct about evolution being driven on a short time scale, for a small set of related species. Yet Gould *may* be correct, as well, about evolution being fundamentally passive on a longer time scale, over the entire tree of biological life. In one of the earliest works to model evolution computationally in order to characterize active versus passive trends, Raup et al. (1973) called attention to the fact that fully deterministic, driven trends acting at small scales are in fact likely to be at the base of larger scale trends, even if those large scale trends turn out to be passive. A mix of many, potentially opposing trends might very well appear random and undirected when integrated together. What our current simulations show is that, indeed, while evolution undoubtedly



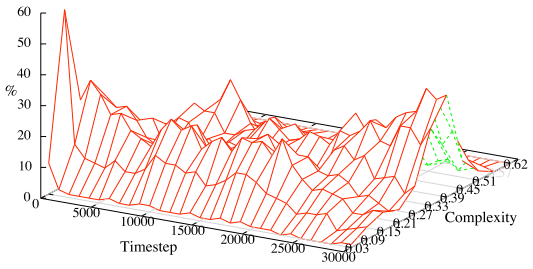
(a)



(b)



(c)



(d)

Figure 3: Histograms of complexity over time for individual runs. (a) and (b) are passive runs. (c) and (d) are driven runs.

drives complexity changes, according to perfectly standard expectations about the evolutionary fitness of those changes, it does not drive in just one direction. When complexity increase is of an evolutionary advantage it will be selected for, just as will complexity decrease. And when a species' complexity is "good enough", so that any increase or decrease is likely to involve a step away from a local fitness maximum, evolution will mildly select for and stabilize the existing level of complexity. This goes a long way towards explaining the observation by Dennett (1996), "The cheapest, least intensively designed system will be 'discovered' first by Mother Nature, and myopically selected."

Looking forward, though we have yet to address the issue experimentally, we expect any increase in agent interactions with the world, any increase in complexity of the environment, and any increase in the available range of niches—Knoll and Bambach (2000)'s expanding *ecospace*—to produce an increase in evolved neural complexity of agents in the world. All niches are not created equal, and we suspect that evolutionary occupation of more and richer parts of ecospace will, as Knoll suggests, result in a fundamentally driven growth in complexity both at the largest scales of biology and in our artificial worlds. And in Polyworld we expect **C**, our measure of neural complexity, to quantify and document that trend.

Acknowledgements

Thanks to David Brinda for Python and gnuplot hacking.

References

Adami, C. (2002). What is complexity? *BioEssays*, 24:1085–1094.

Adami, C., Ofria, C., and Collier, T. (2000). Evolution of biological complexity. *PNAS*, 97(9):4463–4468.

Arthur, B. (1994). On the evolution of complexity. In Cowan, G. A. e. a., editor, *Complexity: Metaphors, Models, and Reality*, pages 65–81. Addison-Wesley, Reading, MA.

Bedau, M., Snyder, E., Brown, C., and Packard, N. (1997). A comparison of evolutionary activity in artificial evolving systems and in the biosphere. In Husbands, P. and Harvey, I., editors, *Proceedings of the Fourth European Conference on Artificial Life*, pages 125–134. MIT Press, Cambridge, MA.

Bedau, M. A. (1995). Three illustrations of artificial life's working hypothesis. In Banzhaf, W. and Eeckman, F. E., editors, *Evolution and Biocomputation: Computational Models of Evolution, Lecture Notes in Computer Science Vol. 899*, pages 53–68. Springer, Berlin.

Bedau, M. A., Synyder, E., and Packard, N. H. (1998). A classification of long-term evolutionary dynamics. In Adami, C. e. a., editor, *Artificial Life VI, Proceedings of the Sixth International Conference on the Simulation and Synthesis of Living Systems*, pages 228–237. MIT Press, Cambridge, MA.

Bonner, J. (1988). *The Evolution of Complexity by Means of Natural Selection*. Princeton Univ. Press., Princeton, NJ.

Carroll, S. (2001). Chance and necessity: the evolution of morphological complexity and diversity. *Nature*, 409:1102–1109.

Cope, E. D. (1871). The method of creation of organic forms. *Proc. Am. Phil. Soc.*, 12:229–263.

Dawkins, R. (1997). Human chauvinism (a review of gould's full house). *Evolution*, 51(3):1015–1020.

Dennett, D. C. (1996). *Kinds of Minds: Toward an Understanding of Consciousness*. HarperCollins.

Gould, S. J. (1989). *Wonderful Life: The Burgess Shale and the Nature of History*. Norton, New York.

Gould, S. J. (1994). The evolution of life on earth. *Scientific American*, 271(4):62–69.

Gould, S. J. (1996). *Full House*. Harmony Books, New York.

Heylighen, F. (2000). Evolutionary transitions: how do levels of complexity emerge? *citeseer.ist.psu.edu/375313.html*.

Huynen, M. A. (1996). Exploring phenotype space through neutral evolution. *Journal of Molecular Evolution*, 43(3):165–169.

Katz, M. J. (1987). *Is evolution random?* Alan R. Liss, New York.

Kimura, M. (1983). *The Neutral Theory of Molecular Evolution*. Cambridge University Press.

Knoll, A. and Bambach, R. K. (2000). Directionality in the history of life: Diffusion from the left wall or repeated scaling of the right? *Paleobiology*, 26:4 (Supplement):1–14.

Lungarella, M., Pegors, T., Bulwinkle, D., and Sporns, O. (2005). Methods for quantifying the information structure of sensory and motor data. *Neuroinformatics*, 3(3):243–262.

Maynard Smith, J. (1970). Time in the evolutionary process. *Studium Generale*, 23:266–272.

McShea, D. W. (1991). Complexity and evolution: What everybody knows. *Biology and Philosophy*, 6:303–324.

McShea, D. W. (1993). Evolutionary change in the morphological complexity of the mammalian vertebral column. *Evolution*, 47(3):730–740.

McShea, D. W. (1994). Mechanisms of large-scale evolutionary trends. *Evolution*, 48:1747–1763.

McShea, D. W. (1996). Metazoan complexity and evolution: Is there a trend? *Evolution*, 50:477–492.

McShea, D. W. (2001). The minor transitions in hierarchical evolution and the question of a directional bias. *J. Evol. Biol.*, 14:502–518.

McShea, D. W. (2005). The evolution of complexity without natural selection, a possible large-scale trend of the fourth kind. *Paleobiology*, 31:2 (Supplement):146–156.

Newman, M. E. J. and Engelhardt, R. (1998). Effects of neutral selection on the evolution of molecular species. *Proc. R. Soc. London B*, pages 1333–1338.

Raup, D. M. and Gould, S. J. (1974). Stochastic simulation and evolution of morphology–towards a nomothetic paleontology. *Systematic Zoology*, 23(3):305–322.

Raup, D. M., Gould, S. J., Schopf, T. J. M., and Simberloff, D. S. (1973). Stochastic models of phylogeny and the evolution of diversity. *Journal of Geology*, 81:525–542.

Rechsteiner, A. and Bedau, M. A. (1999). A generic neutral model for quantitative comparison of genotypic evolutionary activity. In *Advances in Artificial Life: Fifth European Conference on Artificial Life Proceedings*, pages 109–118. Springer-Verlag, Heidelberg.

Rensch, B. (1960a). *Evolution above the species level*. Columbia Univ. Press, New York.

Rensch, B. (1960b). The laws of evolution. In Tax, S., editor, *The Evolution of Life*, pages 95–116. Univ. of Chicago Press, Chicago.

Saunders, P. T. and Ho, M. W. (1976). On the increase in complexity in evolution. *J. Theor. Biol.*, 63:375–384.

Sporns, O. and Lungarella, M. (2006). Evolving coordinated behavior by maximizing information structure. In Rocha, L. e. a., editor, *Artificial Life X: Proceedings of the Tenth International Conference on the Simulation and Synthesis of Living Systems*, pages 323–329. MIT Press (Bradford Books), Cambridge, MA.

Sporns, O., Tononi, G., and Edelman, G. (2000). Theoretical neuroanatomy: Relating anatomical and functional connectivity in graphs and cortical connection matrices. *Cerebral Cortex*, 10:127–141.

Thomas, R. D. K. and Reif, W.-E. (1993). The skeleton space: A finite set of organic designs. *Evolution*, 47(2):341–360.

Tononi, G., Edelman, G., and Sporns, O. (1998). Complexity and coherency: integrating information in the brain. *Trends in Cognitive Sciences*, 2(12):474–484.

Tononi, G., Sporns, O., and Edelman, G. (1994). A measure for brain complexity: Relating functional segregation and integration in the nervous system. *Proc. Nat. Acad. Sci.*, 91:5033–5037.

Turney, P. (1999). Increasing evolvability considered as a large-scale trend in evolution. In Wu, A., editor, *Proceedings of the Workshop on Evolvability at the 1999 Genetic and Evolutionary Computation Conference (GECCO-99)*, pages 43–46.

Turney, P. (2000). A simple model of unbounded evolutionary versatility as a largest-scale trend in organismal evolution. *Artificial Life*, 6:109–128.

Valentine, J. W., Collins, A. G., and Meyer, C. P. (1994). Morphological complexity increase in metazoans. *Paleobiology*, 20(2):131–142.

Waddington, C. H. (1969). Paradigm for an evolutionary process. In Waddington, C. H., editor, *Towards a Theoretical Biology, Vol. 2*, pages 106–128. Aldine, Chicago.

Wagner, P. J. (1996). Contrasting the underlying patterns of active trends in morphologic evolution evolution. *Evolution*, 50(3):990–1007.

Yaeger, L. S. (1994). Computational genetics, physiology, metabolism, neural systems, learning, vision, and behavior or Polyworld: Life in a new context. In Langton, C. G., editor, *Proceedings of the Artificial Life III Conference*, pages 263–298. Addison-Wesley.

Yaeger, L. S. and Sporns, O. (2006). Evolution of neural structure and complexity in a computational ecology. In Rocha, L. e. a., editor, *Artificial Life X: Proceedings of the Tenth International Conference on the Simulation and Synthesis of Living Systems*, pages 330–336. MIT Press (Bradford Books), Cambridge, MA.

Fitness Based Identification of a Robot Structure

Juan Cristobal Zagal[1], Javier Ruiz-del-Solar[2], Adrian Galo Palacios[1]

[1]Centro de Neurociencia de Valparaíso, Universidad de Valparaíso,
and Instituto de Sistemas Complejos de Valparaíso, Chile
[2]Departamento de Ingeniería Eléctrica, Universidad de Chile, Chile
jzagal@ing.uchile.cl

Abstract

Embodiment theory suggests that recurrent processes of sensorimotor activity give rise to cognitive structures. In the case of robots, internal sensorimotor activity generated with physics simulators can be exploited to expand the historical domain of action, however pre-engineered simulations are limited by the reality gap problem. Alternatively simulation might be inferred and self-constructed out of data collected during robot functioning. Fundamental to this line of research is defining a distance function to asses the potential of candidate robot simulations to reproduce real world activity. In this paper we study the characteristics of a distance function based on behavioral fitness measurements. We show how this function can be applied for the generation of behaviors using an algorithm that co-evolves a robot and its simulation. The experiments show how the monotonicity of the function increases with the number of behaviors being tested in reality and with the genotypic diversity of corresponding robot controllers. Moreover it allows for the accurate identification of behavior-relevant parameters contained in the simulation. The metric shows an advantage, when compared to other metrics, for assessing the quality of simulators over long time scales of robot behavioral evaluation.

Introduction

The long-term motivation of this investigation is exploring how real robots can generate more interesting (and practical) behaviors during their ontogeny. The experience (Hornby et al., 1999) shows that a large amount of time is required for obtaining such behaviors as the result of interaction with a real environment. The use of simulation has been explored in order to expand the historical domain of action that a robot undergoes (Ziemke, 2003; Jakobi, 1998). However, according to the reality gap problem, controllers generated in simulation fail to perform similarly once transferred to a real environment. Alternatively we consider that robot simulation must be grounded and self-constructed rather than as a result of a pure engineering design process.

Following this line of thought, we have developed a dynamic reconfigurable robot simulator (Zagal and Ruiz-del-Solar, 2004), together with an algorithm (Zagal et al., 2004; Zagal and Ruiz-del-Solar, 2007) that allows a robot to continuously construct and validate its simulation by co-evolving it with its controller. The method have shown the highest learning performance[1] when compared to other machine learning methods (Genetic Algorithms, Policy Gradient Reinforcement Learning, Evolutionary Hill Climbing With Line Search, Powell Direction Set) that have been applied to the task of gait generation with AIBO robots. It was also successfully applied to the automatic generation of unconstrained ball kick behaviors with AIBO robots (Zagal et al., 2004; Zagal and Ruiz-del-Solar, 2007).

Similarly in (Philipona et al., 2004) the question of whether *there is an algorithm linked to an unknown body that can infer by itself information about the body and the world it is in* was raised. According to experiments with a simulated head they concluded that *sensorimotor laws possess intrinsic properties related to the structure of the physical world in which an organism's body is embedded*. In (Bongard and Lipson, 2004) the *Estimation Exploration* algorithm is proposed as a way to co-evolve a robot and its simulator. They later applied this algorithm to real robots (Bongard et al., 2006). Converging approaches are presented in (Vaughan and Zuluaga, 2006) where self-simulation is proposed for robot planning, and (Ziemke et al., 2005) where internal simulation of robot perception is explored.

Central to this work is defining a distance function to asses the quality of candidate robot simulations. Different functions have been applied; the *rolling mean metric* (Bongard and Lipson, 2004) aims at comparing sensor time series resulting from a target robot and candidate robot simulations. However, according to their proponents *quantitatively comparing sensor data from two highly coupled, highly non linear machines,... is very difficult: slight differences between the two machines rapidly leads to uncorrelated signals*. In (Lungarella et al., 2005) it is proposed that sensorimotor activity can be characterized by looking at their statistical regularities. From this idea in (Mirza et al., 2007)

[1]Learning performance defined as $LP = \frac{IF \times d}{e}$, where $IF = \frac{f_{end} - f_0}{f_{max}}$ is the normalized fitness (f) improvement, d is the controller dimensionality and e is the number of evaluations performed in a real robot.

the *experience metric* over a temporal window of sensorimotor observation (statistical distance) was proposed, giving some insights on the relation of the horizon of experience and cycle time of interactions. Such statistically based metrics have the potential of overcoming the limitations of a time dependent comparison.

In this paper we explore a distance function based on the average fitness discrepancies of a set of robot behaviors tested in reality and in candidate robot simulations. The function has already been shown to be useful at reducing fitness discrepancies of real versus simulated robots during co-evolutionary experiments (Zagal et al., 2004). The intention of this paper is to address some questions that remain to be clarified:

1. We wonder if a minimization of this function does necessarily imply a better identification of aspects from reality such as structural robot variables contained in the simulation, or alternatively if it generates bizarre representations that are good just for reproducing behavior.

2. If the first is true we wonder if the function just allows for good approximations or whether it might allow a perfect match of measurable quantities to be achieved.

3. To aid genetic search over a space of candidate simulations the function should be monotonic with respect to the identification error, thus we wonder about how is this monotonicity being affected by the number of behaviors.

4. Another question is about how this monotonicity is affected by the genotypic diversity of these behaviors.

Answering these questions is fundamental towards generating robots with self-modeling capabilities. Since the task requires a reality that can be measurable at the hands of the experimenter we use a simulated reality defined by parameters that are to be uncovered by the methodology under inspection.

The remainder of this article is organized as follows: Definitions are presented in section II. The principle of operation of the function under analysis is presented in section III. Experimental results are presented in section IV. Comparisons with alternative metrics are presented in section V. Conclusions and projection of this work are given in section VI.

Definitions

Simulation: A robot simulation is defined by a vector $s = \{s_1, \ldots, s_{N_s}\}$ in the space S of possible simulations. The dimensions of this space might be defined by morphological aspects such as the length, width, shape or weight of robot components as well as their topological relation. It might also include aspects such as the friction among different elements, gravitational forces, motor parameters such as PID servo constants, etc. The experimenter defines the S boundaries of each parameter $s_i \in [min_i, max_i]$ with $i = \{1, \ldots, N_s\}$. In the particular case in which reality is defined as a point $s_r \in S$ we will refer to a simulation as any point $s \neq s_r$ in S.

Reality: Reality is the target operational environment of the robot. We present experiments in which reality corresponds to a particular realization of the simulation $s_r \in S$. As it will be described, s_r is unknown for the robot and it should be determined by the algorithm by relying on behavioral comparisons.

Controller: A robot controller is defined by the vector $c = \{c_1, \ldots, c_{N_c}\}$ in a space C of possible robot controllers. The space C might include morphological descriptors of the robot besides controller-related parameters. However, we have not performed experiments for the evolution of robot morphologies. The experimenter defines the C boundaries of each parameter $c_i \in [min_i, max_i]$ with $i = \{1, \ldots, N_c\}$.

Behavior: It is the set of actions that a robot executes in response to the environment E. The characterization and qualification of a robot behavior necessarily depends on the observer. From a single viewpoint we can model behavior in discrete time $t_j = j \cdot \Delta t$ as a time series $B = \{X_0, \ldots, X_{N_b-1}\}$ of N_b vector states $X_j = \{x_1, \ldots, x_{N_d}\}$, each describing N_d dynamical parameters, such as position, rotation and velocity, of bodies composing the robot or interacting with it. If we assume a set of fixed initial conditions, a fixed reference system and fixed evaluation period $T_e = N_b \cdot \Delta t$, we can establish that the robot behavior B is a function of the robot controller c and the evaluation environment E. Thus we have $B = B(E, c)$. In this context E might be either a simulation defined by a point s in S or the reality itself. Clearly in the later case $''E = reality''$ is just an abstraction[2] for the sake of consistency.

Fitness: It is the behavioral evaluation provided by the experimenter. From a set of M robot controllers we note the fitness of robot controller c_k, with $k = \{1, \ldots, M\}$ that elicit behavior $B(E, c_k)$ as f_{Ek} with $E = r$ for reality and $E = s$ for simulation. For example, at the end of T_e it might be the distance traveled by the robot, the distance traveled by a ball that the robot kicks, the amount of consumed energy, etc.

Exploiting Behavioral Consistency

In this section we discuss the problem of how to construct a robot sensorimotor simulation from data collected during robot functioning. If the behavior elicited by robot c in simulation s is similar to the behavior observed in reality we write

[2] We intend to approximate relevant aspects of reality, but not to represent it.

$$B(s,c) \approx B(r,c) \quad (1)$$

more precisely this means that at any time step j we also have

$$X_{sj} \approx X_{rj} \; \forall j = \{0, \ldots, N_b - 1\} \quad (2)$$

where X_{sj} stands for the state vector of $B(s,c)$ at time step j, and X_{rj} is the state vector of $B(r,c)$ at time step j. In other words there should be a match along time of the robot state in both simulation and reality.

If we somehow measure the degree in which this match is obtained for each simulation s, we can derive a useful distance for adapting simulation to match behavior. Unfortunately the state X_{rj} is generally unobservable[3]. On the other hand, by definition of an Evolutionary Robotics problem the behavioral fitness obtained in simulation f_{sk} and reality f_{rk} are always available. We can thus define in terms of these measurements the behavioral fitness discrepancy elicited by robot c_k as

$$\delta_k = |f_{sk} - f_{rk}| \quad (3)$$

If behavioral discrepancies, as expressed in (3), are reduced for various behaviors, then it is natural to expect that simulation approximates reality better in those characteristics that are relevant for the execution of these behaviors. We note the average behavioral differences $\Delta_{fitness}$ as:

$$\Delta_{fitness} = \frac{1}{m} \sum_{k=1}^{m} \delta_k = \frac{1}{m} \sum_{k=1}^{m} |f_{sk} - f_{rk}| \quad (4)$$

Back-to-Reality algorithm

Using this distance we follow the Back-to-Reality (BTR) algorithm steps in order to construct simulations during robot ontogeny. Detailed descriptions of the algorithm are reported in (Zagal et al., 2004) and (Zagal and Ruiz-del-Solar, 2007). At each iteration i the algorithm co-evolves robot controllers with robot simulators by executing the following steps:

Step 1, *robot controller search under simulation:* Using the best known simulation s_{i-1} as environment, genetic search is conducted over the controller solution space C. A starting population of M controller individuals is obtained by performing bit changes with probability p_m over the solution c_{i-1} which is given as a seed. For the first iteration the population can be generated as random or biased by a known starting solution c_0.

The search is steered towards maximizing the fitness $f(B(s_{i-1}, c_i))$. The amount of generations during which genetic search is conducted in this step should be small if the problem present a high tendency for drift[4].

Step 2, *selection, transfer and test:* A set of ($m < M$) controllers are selected in order of descending fitness and tested in reality. Corresponding fitness values f_{rk}, with $k = \{1 \ldots m\}$ are stored. If it is not possible to find m transferable individuals from the last generation, they will have to be taken, in descending order, from the previous generations obtained in Step 1.

Step 3, *simulation search:* The best existing simulator solution s_{i-1} is used in order to bias a population of L simulator individuals. In the case of the first iteration this population is generated as random or as biased by a known starting simulator solution s_0. A simulation s_i is obtained by steering the evolution towards minimizing $\Delta_{fitness}$.

The algorithm continues by taking the simulation obtained in step 3 as a new environment for step 1 in the next iteration. There is a genotypic similarity among the m controllers that triggers a phenotypic similarity among corresponding behaviors. A probability p_m per bit controls the rate of mutation for a population constructed from a given controller c_i.

Results

Experimental settings

A dynamic simulation of an ant-like robot (hexapod) was implemented using the UCHILSIM simulator (Zagal and Ruiz-del-Solar, 2004). Figure 1 (a) shows the configuration of the 15 rigid robot bodies. The alitrunk, petiole and gaster are represented by bodies b_0, b_1 and b_2. Bodies $\{b_3, \ldots, b_8\}$ correspond to the femur and $\{b_9, \ldots, b_{14}\}$ are the tibia and tarsus of each leg. The joint j_0 connect body b_0 and b_1, similarly the joint j_1 connect body b_1 with b_2. The joints $\{j_2, \ldots, j_7\}$ connect each femur with a corresponding alitrunk having vertical and frontal axis of motion as depicted on the figure. The joints $\{j_8, \ldots, j_{13}\}$ connect each femur and tibia with a frontal axis of motion. Each independent axis of motion i (a total of 18) is motorized and torque is applied according to the output of PID dynamic compensators that follow a motion reference signal $r_i = \theta_i + a_i sin(wt + \phi_i)$, where θ_i is a pre-defined central angle of oscillation for each motor i. Equal uniform mass density is given to all bodies.

The default robot posture (when $r_i = \theta_i \, \forall i$) is presented in Figure 1 (b). The behavior evaluation time T_e is set to 1.7 seconds. A physics integration step takes $\Delta_t = 5 \times 10^{-4}$ seconds, therefore the state vector $X_{\{s/r\}j}$ is computed $N_b = 3400$ times along a behavior evaluation. The frequency is selected to be $w = 60Hz$. The amplitude a_i and motion phase ϕ_i are defined by the robot controller vector

[3] In control theory a system is observable if, for any possible sequence of state and control vectors, the current state can be determined in finite time using only the outputs.

[4] A pathology of co-evolving systems in which the selection pressure of one population has no influence in the co-evolving population.

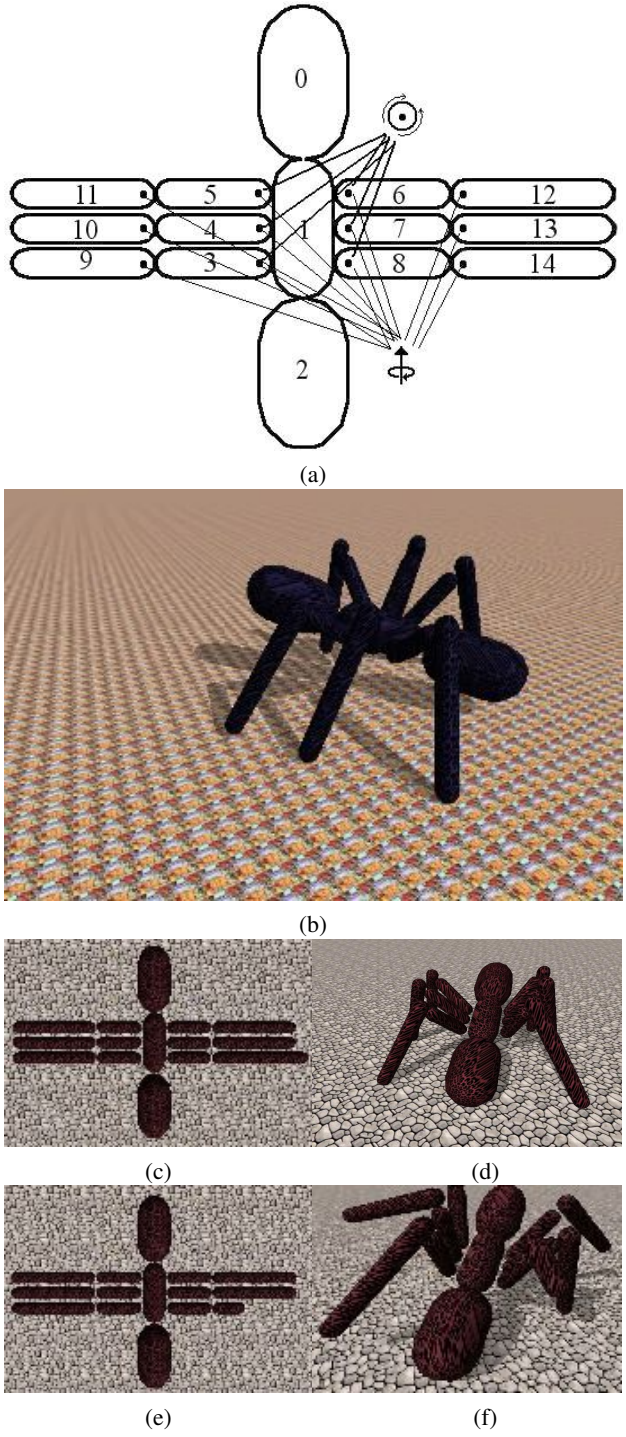


Figure 1: In (a) configuration of the 15 rigid bodies of the robot; motorized joints are connecting bodies $\{3, \ldots, 8\}$ with body 1 in two orthogonal axis of motion. Bodies $\{9, \ldots, 14\}$ are connected to the previous set constrained to one axis of motion. In (b) the robot under its default posture. In (c,d) reality defined such that body 14 length is $bl_{max} = 1.6$. In (e,f) example of other realization of the simulator for which body 14 length is $bl_{min} = 0.4$. Remaining body lengths are $\{bl_3, \ldots, bl_8\} = 0.64$ and $\{bl_9, \ldots, bl_{13}\} = 1.36$ in both cases.

under search, $c = \{a_1, \phi_1, \ldots, a_{18}, \phi_{18}\}$ of 36 elements. These parameters are in the range $a_i \in \{a_{min}, a_{max}\}$ and $\phi_i \in \{\phi_{min}, \phi_{max}\}$ for each joint. The robot simulator is defined as a vector s representing the actual lengths of the ant limbs $s = \{bl_3, \ldots, bl_{14}\}$ of 12 elements in the range $bl_i \in \{bl_{min}, bl_{max}\}$.

We have selected the following parameter range: $a_{min} = 0.12989$, $a_{max} = 0.314$, $\phi_{min} = 0$, $\phi_{max} = 3.1416$, $bl_{min} = 0.4$, $bl_{max} = 1.6$. It is important to notice that the selected parametrization might bring about radically different behaviors since no symmetry simplifications have been made for the leg motion pattern and the parameter range is sufficiently large to produce a variety of different behaviors (falling upside-down, walking in circles, backwards, sideways, forward, jumping, etc).

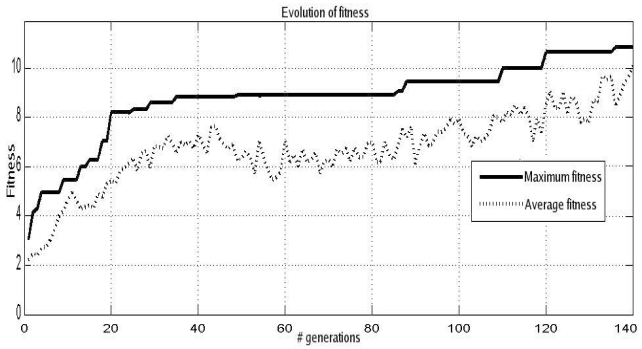
Fitness discrepancy and morphology

As result of applying BTR step 1 using a starting simulator s_0 and and robot controller c_0 we obtained a population of $m = 15$ robot controllers. Corresponding fitness was measured in reality according to BTR step 2, stored and ranked. Figure 2 (a) shows corresponding evolution of controller fitness. The best transferable controller c_1 achieved $11m/s$ in simulation.

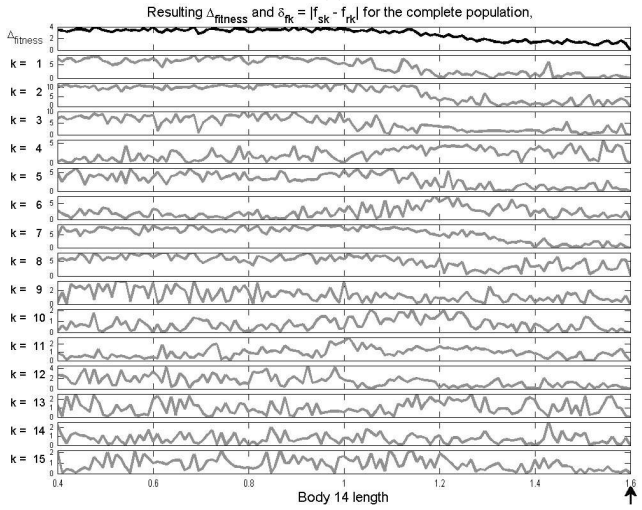
Using this population of controllers we scanned $\Delta_{fitness}$ as function of morphological variations (of simulated versus real robot). In this case simulation is defined by a vector s that differs only in one parameter with respect to reality s_r, and then we evaluate $\Delta_{fitness}$ while varying this parameter along its whole range. As a first test, let us define reality s_r in S such that the length of body 14 takes the value $bl_{14} = bl_{max} = 1.6$ while the remaining body lengths are set to $\{bl_3, \ldots, bl_8\} = 0.64$ and $\{bl_9, \ldots, bl_{13}\} = 1.36$ as shown in Figure 1 (c,d). In order to illustrate the range of search a point s_0 different from reality is shown in Figure 1 (e,f) such that $bl_{14} = bl_{min} = 0.4$.

Figure 2 (b) shows results of scanning $\Delta_{fitness}$ along the complete parameter range of bl_{14}. The partial components $\delta_k = |f_{rk} - f_{sk}|$ used for computing $\Delta_{fitness}$ are presented for all behaviors $k = \{1, \ldots, 15\}$ as functions of bl_{14}. Figure 2 (c) presents results from a second test in which the length of body 5, bl_5, is chosen as the varying parameter. In this case reality is such that the length of this body is $bl_5 = 0.88$, while the remaining body lengths are $\{bl_3, \ldots, bl_4\} = 0.64$, $\{bl_6, \ldots, bl_8\} = 0.64$ and $\{bl_9, \ldots, bl_{14}\} = 1.36$. Similarly as before $m = 15$ individuals are selected from a population of robot controllers.

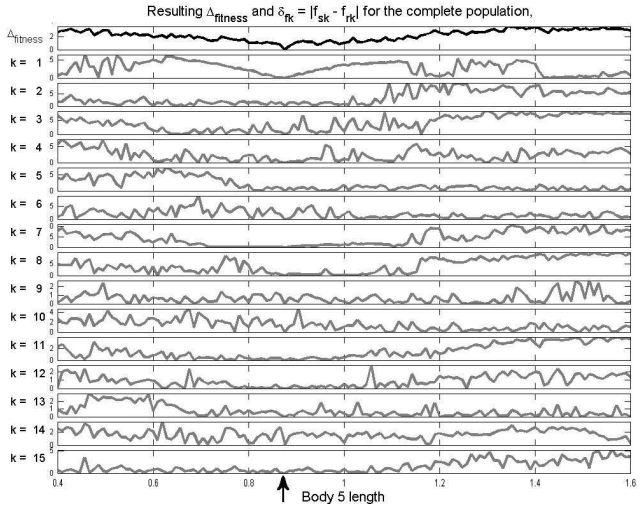
In order to understand the influence of m on the monotonicity of $\Delta_{fitness}$ we generated $M = 45$ controllers by modifying c_1 genotype bits with a probability $p_m = 0.02$ per bit. Figure 3 (a) shows scans of δ_k computed separately around body $bl_5 = 0.88$, and in (b) scans of $\Delta_{fitness}$ computed as a function of different amounts m of behaviors. The same figure shows in two subplots corresponding fitness val-



(a)



(b)



(c)

Figure 2: In (a) evolution of controller fitness. In (b,c): $\Delta_{fitness}$ together with its $m = 15$ behavioral components $\delta_k = |f_{rk} - f_{sk}|$ resulting from scanning body 14 (b) and body 5 (c) lengths along the full parameter stroke $bl_{14,5} = [0.4, \ldots, 1.6]$. It is particularly interesting to observe the decreasing monotonic behavior of $\Delta_{fitness}$ when approximating the real value of 1.6 (a) and 0.88 (b). Further observations are described in the main text.

ues resulting from each one of the 45 behaviors when measured in reality (left) and for the starting simulation (right). Figure 3 (c) shows 16 $\Delta_{fitness}$ curves computed for a fixed number of behaviors (15) but increasing the diversity parameter p_m with the darkness of curves.

Finally, following BTR step 3, $\Delta_{fitness}$ is used to identify a hidden simulation point defined by $bl_5 = 0.54$, $bl_6 = 0.74$, $bl_3 = 0.54$, $bl_8 = 0.74$. Figure 4 (a) shows the minimization of $\Delta_{fitness}$ along 100 generations. In (b) the resulting parametric convergence is shown. We have made the following observations from these experiments:

1. A first observation is that when simulation equals reality, this is when the varying parameters match corresponding real values ($bl_{14} = bl_{max} = 1.6$ in the first test and $bl_5 = 0.88$ in the second test) with the remaining parameters left equal, we verify for the $k = \{1, \ldots, m\}$ partial behavioral discrepancies that $\delta_k = 0 \; \forall k$. This is an experimental support for the following theorem:

   Theorem 1 $s = s_r \Rightarrow \delta_k = 0, \; \forall k \, / k = \{1, \ldots, m\}$.

   This theorem is trivial since we have assumed a set of conditions in order to guarantee that if $s = s_r$ we can reproduce behaviors and thus obtain the same fitness values.

2. Similarly we verify that when simulation equals reality $\Delta_{fitness} = 0$. This result is an experimental support of the following theorem:

   Theorem 2 $s = s_r \Rightarrow \Delta_{fitness} = 0$.

   This theorem is also trivial (but useful as well), from previous theorem and equation (4) we have

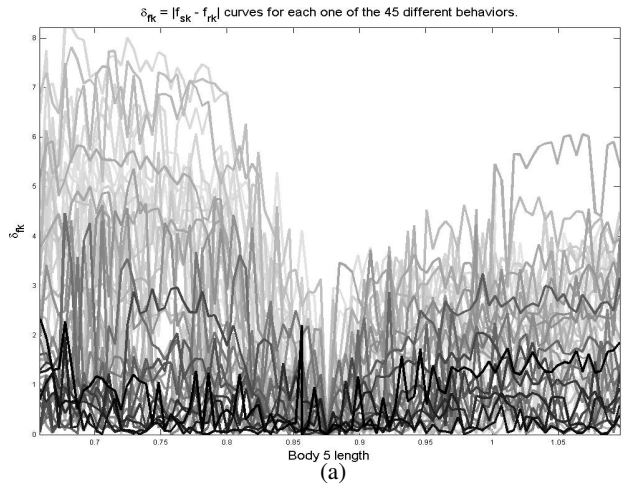
   $$\Delta_{fitness} = \frac{1}{m} \sum_{k=1}^{m} \delta_k = \frac{1}{m} \sum_{k=1}^{m} 0 = 0 \qquad (5)$$

3. We also observe that $\Delta_{fitness} \geq 0 \; \forall s \, / \, s \neq s_r$ and that $\Delta_{fitness} = 0 \;\Leftrightarrow\; s = s_r$ for the observed range of s. This suggests the following two hypotheses:

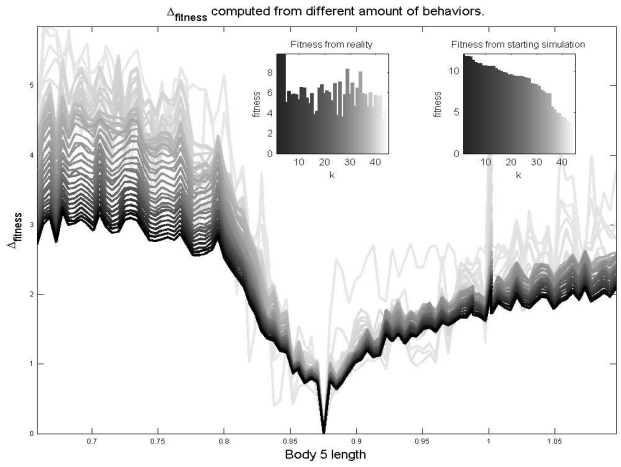
   Hypothesis 1 $\Delta_{fitness} = 0 \Rightarrow$ *a subset of parameters in S that are relevant for the execution of m behaviors has been correctly identified by s from s_r.*

   Hypothesis 2 $\Delta_{fitness} = 0 \Rightarrow s = s_r$ *if and only if S contains only parameters which are relevant for the execution of m behaviors, with $m \to \infty$.*

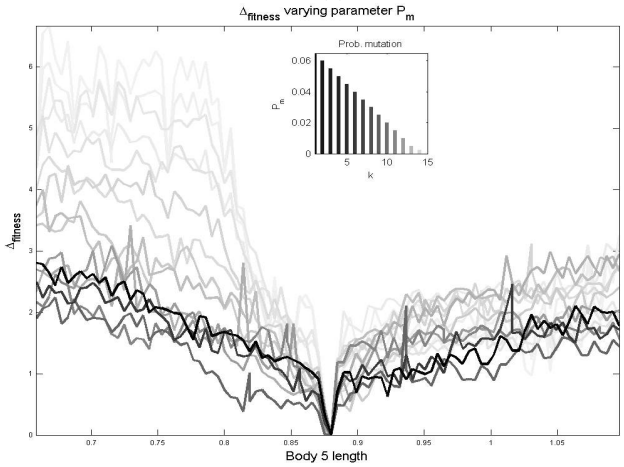
4. Even though we observe that $\Delta_{fitness} = 0 \;\Leftrightarrow\; s = s_r$, the partial behavioral discrepancies δ_k might become zero at points such that $s \neq s_r$. In fact there is a likelihood that the fitness of a particular behavior is the same in reality



(a)



(b)



(c)

Figure 3: In (a) 45 δ_k curves computed for variations of bl_5 around $s = s_r$. In (b) 45 $\Delta_{fitness}$ curves computed for different number of behaviors. Darkness increases with the number of behaviors, the subplots shows fitness values for each one of the 45 controllers in reality (left) and the starting simulation (right). In (c) 16 $\Delta_{fitness}$ curves computed for a fixed number of behaviors (15) but increasing the parameter p_m which is depicted in the sub-plot. Curve darkness increases with P_m.

and simulation even for $s \neq s_r$, but the key point is that it is extremely unlikely that the coincidence happens at the same point s with another behavior. This is a great illustration of the reason several behavioral comparisons are averaged in order to achieve useful measurement.

5. It is possible to recognize a plateau on $\Delta_{fitness}$ for values $bl_{14} \leq 1.1$ in Figure 2 (b). This can be clearly understood when looking at Figure 1 (c,d), what happens is that for such values the corresponding leg does not actually touch the ground and therefore, as can be observed, it does not have an impact in behavior until it reaches values above 1.1. A similar observation can be made in Figure 2 (c), when $bl_5 \geq 1.3$ the whole extremity is lifted away from the ground and therefore the parameter changes beyond that value are not affecting behavior.

6. We can observe as well that $\Delta_{fitness}$ monotonically increases with the identification error, i.e. when s moves away from the real value s_r. The Figure 2 (a,b) shows a strong increasing monotonic behavior of $\Delta_{fitness}$ with the superposition of small fluctuations. On the other hand, the characteristic of a single behavioral discrepancy δ_k is not monotonic with the identification error. From Figure 3 (b) we observe how the smoothness of $\Delta_{fitness}$ increases with m. From these observations we make the following hypothesis

   Hypothesis 3 *The smoothness of $\Delta_{fitness}$ as the function of a parameter of S increases with m over the range of behavioral influence of the parameter.*

7. We observe from Figure 3 (c) an increase of the smoothness and linearity of $\Delta_{fitness}$ when increasing the diversity factor p_m. Thus we make another hypothesis

   Hypothesis 4 *The smoothness of $\Delta_{fitness}$ as the function of a parameter of S increases with p_m over the range of behavioral influence of the parameter.*

Comparisons With Other Metrics

Results from applying sensor based metrics such as the rolling mean metric (Bongard and Lipson, 2004) and the Euclidean difference of sensor time series of real versus candidate simulations over the described range of bl_5 are presented in figure 5. A central parameter of the rolling mean metric is the header length h which indicates how much of the starting sensor time series are going to be compared. Corresponding metric is computed for different values of this parameter, ranging from 1.25% up to 100% of the total evaluation time. As can be seen, comparing a very short starting period (1.25%) leads to an almost perfectly linear behavior of the metric, however when increasing the length of sensor data under comparison the monotonicity of the

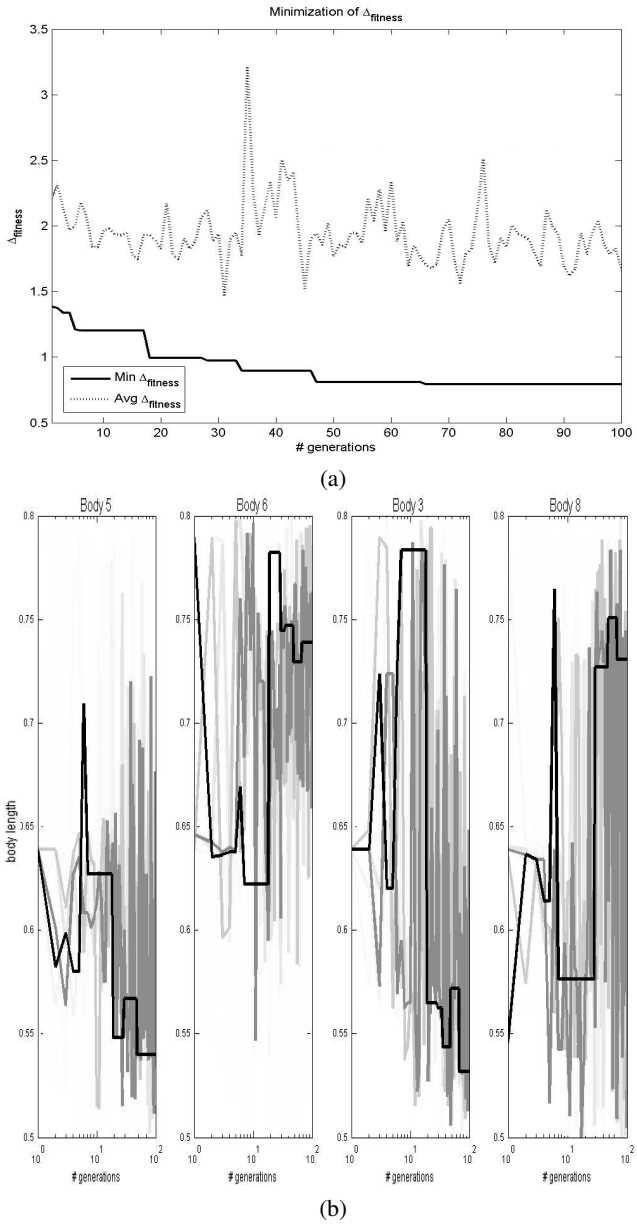


Figure 4: In (a) minimization of $\Delta_{fitness}$ carried along 100 generations. In (b) corresponding convergence to parameters defining reality is shown. Black lines represent parameters encoded by the individuals having minimum $\Delta_{fitness}$. Similarly the remaining individuals are represented by gray lines, where the darkness of curves is proportional to $\Delta_{fitness}$.

curve decreases reaching a non monotonic behavior. The metric behaves similarly as the Euclidean difference of corresponding time series when h is equal to the whole evaluation period; this is depicted with dashed line. Such behavior can be clearly understood since the decorrelation of sensor time series increases with time for slight differences of simulation and reality.

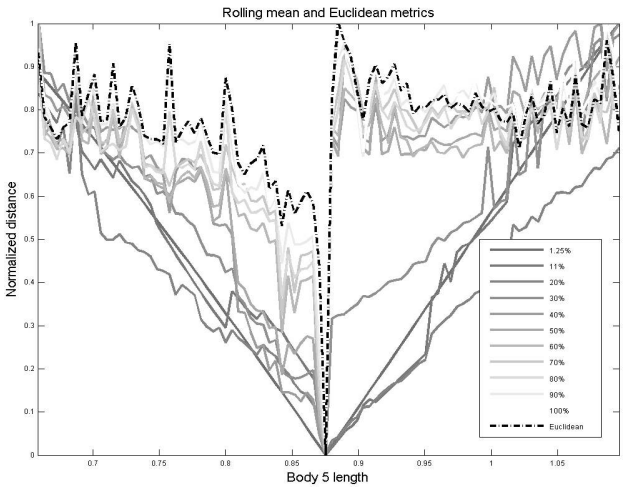


Figure 5: Rolling mean and Euclidean metrics. The rolling mean metric is computed for different values of the starting time header h, ranging from 1.25% up to 100% of the total evaluation time. In dashed line the Euclidean metric is presented considering the whole evaluation time. The figure illustrates how the monotonicity of the rolling mean metric is affected by the parameter h, behaving similar to the Euclidean metric when the whole time window is considered.

Conclusions

We have investigated the behavior of a distance function that can be used for comparing candidate simulators by measuring the average fitness of small variations of behavior. Before drawing conclusions we should remember that in the presented experiments *reality* was also simulated, having the same physical laws as the candidate simulations, but varying morphological aspects of the robots. In principle a negative effect of fitness measurement noise should be reduced with m given the linear construction of $\Delta_{fitness}$. However, performing several behavioral evaluations in a real environment is expensive.

Having this into account we can give the following answers to the main questions that motivated this work:

1. Indeed a minimization of $\Delta_{fitness}$ necessarily implies a better identification of aspects of reality. However these aspects must be related to the execution of behaviors considered under $\Delta_{fitness}$. Thus the methodology under analysis allows a robot to generate a self model of the behavior-relevant components of its interaction with the world. However the methodology would leave undetermined the value of parameters that are not relevant for the execution of behavior (like the color of the head!).

2. We have observed that the function allows a perfect match of those parameters that are relevant for the execution of behaviors to be achieved.

3. The monotonicity of $\Delta_{fitness}$ increases with the number m of behaviors that are tested in reality.

4. The monotonicity of $\Delta_{fitness}$ increases with the diversity of the controllers and it can be controlled with the parameter p_m.

We conclude that the latter two factors should be carefully considered when designing experiments on a fitness based identification of robot structures since the monotonicity of the distance function is a critical factor towards increasing the dimensionality and complexity of simulation search spaces.

Out from these experiments we observe that there is an advantage of using a sensor based time series metric (such as the rolling mean metric) when comparing small time portions of data collected during robot functioning. Moreover, since this apporach involves the collection of a higher amount of data (sensor time series versus fitness), it appears as the right move to experiments in reality given the reduction in hardware trials. However, when time increases, the sensor time series become highly decorrelated if simulation is dissimilar to reality. A fitness based comparison such as $\Delta_{fitness}$ allows us to assess the quality of candidate robot simulation over extended evaluation periods.

Simulation should allow us to reproduce real robot operation over all behavior-relevant time scales. We consider that comparisons cannot be restricted to the first instants of robot operation but must be extended until the outcome of behavior is obtained. An account of multi time scale behavioral comparisons is required.

References

Bongard, J. and Lipson, H. (2004). Once more unto the breach: Co-evolving a robot and its simulator. In *Proc. of the Ninth Int. Conf. on the Simulation and Synthesis of Living Systems (ALIFE9)*, pages 57–62.

Bongard, J., Zykov, V., and Lipson, H. (2006). Resilient Machines Through Continuous Self-Modeling. *Science*, 314(5802):1118–1121.

Hornby, G., Fujita, M., Takamura, S., Yamamoto, T., and Hanagata, O. (1999). Autonomous evolution of gaits with the Sony quadruped robot. *Proc. of the Genetic and Evolutionary Computation Conf.*, 2:1297–1304.

Jakobi, N. (1998). *Minimal Simulations for Evolutionary Robotics.* PhD thesis, University of Sussex.

Lungarella, M., Pegors, T., Bulwinkle, D., and Sporns, O. (2005). Methods for quantifying the informational structure of sensory and motor data. *Neuroinformatics*, 3(3):243–262.

Mirza, N., Nehaniv, C., Dautenhahn, K., and te Boekhorst, R. (2007). Grounded Sensorimotor Interaction Histories in an Information Theoretic Metric Space for Robot Ontogeny. *Adaptive Behavior*, 15(2):167.

Philipona, D., O'Regan, J., Nadal, J., and Coenen, O. (2004). Perception of the structure of the physical world using unknown multimodal sensors and effectors. *Advances in Neural Information Processing Systems*, 16:945–952.

Vaughan, R. T. and Zuluaga, M. (2006). Use your illusion: Sensorimotor self-simulation allows complex agents to plan with incomplete self-knowledge. In *From Animals to Animats IX, Proc. of 9th Int. Conf. on Simulation of Adaptive Behaviour (SAB06)*, Rome, Italy.

Zagal, J. C. and Ruiz-del-Solar, J. (2004). UCHILSIM: A dynamically and visually realistic simulator for the robocup four legged league. In *RoboCup 2004: Robot Soccer World Cup VII*, volume 3276 of *Lecture Notes in Computer Science*, pages 34–45. Springer.

Zagal, J. C. and Ruiz-del-Solar, J. (2007). Combining simulation and reality in evolutionary robotics. *Journal of Intelligent and Robotic Systems*, 50(1):19–39.

Zagal, J. C., Ruiz-del-Solar, J., and Vallejos, P. (2004). Back-to-reality: Crossing the reality gap in evolutionary robotics. In *Proc. 5th IFAC Symp. on Intelligent Autonomous Vehicles (IAV '04)*, Lisbon, Portugal. Elsevier Science Publishers B.V.

Ziemke, T. (2003). On the role of robot simulations in embodied cognitive science. *Journal of Artificial Intelligence and the Simulation of Behavior*, 1(4):389–399.

Ziemke, T., Jirenhed, D., and Hesslow, G. (2005). Internal simulation of perception: a minimal neuro-robotic model. *Neurocomputing*, 68:85–104.

Presented abstracts

Topological properties of evolved robot brains

Christoph Adami[1], Nicolas Chaumont[1], Jeffrey Edlund[2] and Arend Hintze[1]

[1]Keck Graduate Institute
[2]California Institute of Technology
adami@kgi.edu

The topological structure of animal brains is likely to be interesting because the computational power of brains is thought to be almost entirely due to its wiring pattern and hierarchical organization. At the same time, this pattern is not at all well understood, and the information about the wiring pattern of the nematode C. elegans, for example, is unique in the literature. A promising direction for the study of network topology in the absence of detailed biological data is the Artificial Life approach, where functional networks are evolved that determine the survival of artificial organisms in an artificial chemistry and genetics. Recently, we used this approach to understand modularity in evolved artificial metabolic networks and developed new tools to dissect their topological and functional characteristics. Here, we apply some of these tools to the study of the brains of robots that have evolved to behave in a simulated world. The robots that are controlled by these brains are simulated versions of real robots (the ATRV Jr. of the iRobot Corporation) whose properties we tested in our laboratory. Both the robot and its environment are simulated in a three-dimensional world that implements realistic rigid body dynamics via the Open Dynamics Engine (ODE). As a consequence, evolved controllers could in principle be transplanted onto the simulated robots' real-world counterparts.

Neural computational tissues ("brains") are grown from genomes that implement neural network development and function based on a set of rules ("genes") that are conditionally executed, that is, regulated, by a set of simulated proteins produced by the cells in the tissue. This system ("Simnoesis") is based on the "Norgev" platform but was completely rewritten in order to be able to evolve complex tissues that process many temporally varying input signals. We evolve neural tissues on two-dimensional grids (of up to 15x15 neurons) that control a simulated ATRV Jr with 19 sensors (17 sonars, a compass, and a sensor relaying distance to goal), controlling two motors driven by two actuators for differential steering. The evolved tissues control complex robot behavior, such as wall-following, obstacle avoidance, and goal-finding, using a complex network structure reminiscent of the C. elegans connection graph. The fitness evaluation of a genome consists of growing the network, and evaluating the behavior of the robot in a 3D environment akin to the fitness evaluation in the work of Sims). Fitness evaluation and evolution via a Genetic Algorithm is implemented within the EVO software.

We analyze the properties of evolved neural networks using standard tools (such as edge-distribution, shortest-path length, and betweenness centrality), as well as new tools that reveal robustness and modularity via clustering methods and information theory. We find that the topological properties of evolved functional networks are very different from their randomized counterparts, and characterize the "rarity" of these networks with standard statistical tests. Finally, we compare the topological properties of our evolved networks to the connection graph of C. elegans.

Evolutionary robotics and the morphological turn: an epistemological perspective

Fernando Almeida e Costa

Centre for Computational Neuroscience and Robotics, University of Sussex
F.AlmeidaCosta@sussex.ac.uk

What are the philosophical and epistemological implications of work in evolutionary robotics (ER) dealing with the evolution of morphologies and morphogenesis? So far, its theoretical consequences for cognitive science have not been fully fleshed out. Also, investigation in morphologically-based ER has not shown its affiliation in the long tradition of a *morphological thought*. Understanding theoretical implications and the phylum of thought of some line of research may be of great importance not only to the historians of science but to the future development of the research itself. I propose that the shift towards morphodynamics belongs to an old phylum of cognitive and biological thought, both naturalist and structuralist in nature, and is a component of a broader *morphological turn*. Examples of its manifestations include early investigations in A.I., e.g. the studies on morphogenesis and cognitive structures by Alan Turing, the semiophysics firstly proposed by René Thom, or the morphological, non logico-combinatorial structuralism of Claude Lévi-Strauss in cultural anthropology. More recently, work in ER addressed not only the morphodynamics of an agent's body and environment but also the morphological properties of its "perceived world" (Almeida e Costa et al, 2008, Alife XI, these proceedings). By highlighting what is common to these apparently unrelated lines of research, light is shed on the entire framework of the morphological turn. Structuralism is normally seen as the continental current of thought that developed in the 60's and 70's and had its major tenets on linguistics or literary criticism, while being of a logico-combinatorial, algebraic and static nature. But there is another phylum of structuralist thought: one that that can be traced back to the works of D'Arcy Thompson and even to Goethe. This other phylum thinks of structures as dynamical forms in development. It is a naturalist and non-formalist (in the sense of formal logic) approach that considers forms as morphodynamically self-organised wholes. The concept of "transformation" and its mathematical treatment is central to this perspective. Morphodynamically inspired robotics exploits all intrinsic and extrinsic physical properties available. This entails the denial that the "cognitive" properties are to be found at an algorithmic level that dominates the physical properties. Cognitive activity relies crucially on the agent's morphodynamics, actually implying the minimization of control at the algorithmic level. The functionalist principle of the irreducibility of the cognitive level to the physical medium is put aside. Thus, the *morphological turn* opens up the possibility of a non reductionist physics of meaning. The success of modern science, i.e. the physical mechanism that emerged in the XVII century was only possible due to the abandon of the dynamics of forms: the Aristotelian physics of qualities. This implied the impossibility of connecting the new "objectivity" with the qualities of the world has it is perceived. It is often pointed out that the recent embodied approach to cognition refuses the Cartesian mind-body dualism. It should be noted that it also refuses the divide between physics and qualitative form. On a supplementary note, this perspective is totally consistent with high-level and low-level cognitive abilities forming a continuum. The hypothesis of an evolutionary path leading from the emergence of particular human morphologies (feet and hands), hence the ability to walk, to the emergence of language, put forward by palaeontologist André Leroi-Gourhan, in the same structuralist vein afore mentioned, is entirely consistent with this orientation.

Morphodynamics and perceptual worlds: conceptual approach and an experiment in evolutionary robotics

Fernando Almeida e Costa[1], Ian Macinnes[2] and Inman Harvey[1]

[1]Centre for Computational Neuroscience and Robotics, University of Sussex
[2]RBS
F.AlmeidaCosta@sussex.ac.uk

From New A.I. came the deep conceptual insight that cognition is a consequence of the opportunistic exploitation of all morphodynamical properties of an agent's body and environment, which acts to minimize control at the algorithmic level. These properties structure the agent's perceptual world. Here, we aim at those morphological properties that structure the environment "as perceived" by the agent. Jakob von Uexküll's functional circle hypothesis, provides a general framework to understand active perception as morphologically-based. This framework must be seen within a broader *morphological turn*; this occurred in the second half of the 20th century across various fields of research, and is morphologically-based, as opposed to information-theoretic based (Almeida e Costa, 2008, Alife XI, these proceedings). For some authors, developments in dynamical systems theory opened the possibility of a nonreductionist physics of perception and meaning (e.g,. Petitot 2000, Physique du Sens, CNRS Éditions). A dynamicist approach regards organisms as being perturbed by and responding to cues they have been evolutionarily selected to respond to, rather than mirroring or extracting information from the outside world. The morphological structuring of the perceived environment is highly constrained by the particular morphologies of its body, and by the dynamics of those morphologies. To exploit this aspect for engineering and conceptual purposes, a particular method in evolutionary robotics is proposed, based on the functional circle hypothesis by Jakob von Uexküll (Macinnes et al., 2005 Adaptive Behaviour, Vol:14.2. p. 147). A functional circle is an abstract structure that describes the functional relationship between an organism, its "perceived world", and its environment. According to the functional circle hypothesis, a perceptual sign of an object (say, the smell of a mammal's butyric acid, captured by a tick) give rise to a perceptual cue, the *subjective experience* of that object in the organism's (the tick's) *Umwelt*: the word Umwelt was used by von Uexküll to describe the biologically evolved world of perceptions, as perceived by a particular organism/species, which results from the morphodynamical interaction with its environment. This leads to an effector cue which drives the animal to perform some action (say, fall down from the tree under which the mammal is passing), changing the organism's relationship to the object. After the action is performed, the perceptual cue is gone and therefore that functional circle is extinguished but may lead to another (say, dealing with the fur, finding warm skin, then biting). The proposed method consists of changing the mutational operators to evolve functional circles instead of directly evolving sensorimotor loops. The agent's morphodynamics and perceptual world are co-evolved; evidence suggests this enables a closer coupling between body, controller, and environment. The evolving functional circle hypothesis predicts that adding multiple perceptual cues produces robots more adapted to their environment than they would be otherwise. A comparative analysis of the evolved robots suggests that this is the case. An explanation is suggested: the specific positions of the sensors using mutable locations together with body morphology define spatial and temporal relationships with the environment. Co-evolving the agent's morphology, locations of its sensors, and controllers, evolve these relationships as well which implies that we are evolving perceptual cues, and therefore evolving perceptual worlds.

Smart plankton: a new generation of underwater wireless sensor network

Davide Anguita, Davide Brizzolara, Alessandro Ghio and Giancarlo Parodi

University of Genova
Davide.Anguita@unige.it

The aquatic world, which covers more than the 70% of the earth, has been largely unaffected by the WSN revolution (ignited by DARPA funded UC Berkeley "Smart Dust" project) due to the difficulty of transferring most of the knowhow, developed for terrestrial and aerial systems and devices, to their underwater counterparts. Nowadays underwater wireless networks are expensive (US$ 10k or more), sparsely deployed (a few nodes, placed kilometers apart), typically communicating directly to a base-station or sometimes based on the use of underwater manned or unmanned vehicles. Our research is aimed to develop a new generation of UWSN (Underwater Wireless Sensor Network), called Smart Plankton, by drawing inspiration from marine biology and aquatic micro-organism such as zooplankton and phytoplankton. Our target is to develop a self-organizing network composed by a relatively large number of innovative nodes, equipped with sensors for monitoring, surveillance, underwater control and many others potential applications. Inspired to the rich inventory of plankton adaptations our research is oriented to explore innovative solutions in following areas:

1. implementation of the single network node focusing on:

a)the use of reconfigurable architecture, balancing the need of computation (sense, communicate, etc.) with the survivability constraints (energy foraging and storage), with algorithms recently developed for computational embedded intelligence (e.g. kernel methods for embedded and pervasive systems [D. Anguita, A. Ghio and S. Pischiutta, Adaptive Hardware and Systems, p.571, 2007]) for acquiring or improving intelligent behavior;

b)a mobility system based on body thermal expansion of solids and liquids and compression under pressure, such as in sperm whale which uses spermaceti, a semi-liquid, waxy substance for movement and stability;

2. communication between nodes because, in comparison with ground-based sensor networks, mobile UWSNs cannot employ radio frequency (RF). The alternative and more innovative method of optical communication, suggested also by the natural world (e.g. quorum sensing through bioluminescence in plankton shoals [F.J.Jochem, Marine Biology, Vol. 135, p.721]), can allow the development of a high rate, low power, long life and low cost communication link among devices. At our Department, we are testing the use of LEDs and phototransistors for developing an underwater optical communication system (based on 802.11a protocol) considering that experimental tests have shown that the better wavelength lies around 420 nm (blue-violet wavelengths) and that the value changes in presence of turbidity;

3. energy scavenging in order to allow a long life to the network; energy can be generated by using electrochemically active bacteria [B.E.Logan and J.M.Regan, Environmental Science and Technology, 40, p. 5172] which have been recently discovered and have the property to oxidize organic matter and release the electron to an electrode. This has some definite advantages over the use of a chemical catalyst, as bacteria can sustain themselves and recover after inadvertent poisoning;

4. shoal intelligence in order to allow Smart Plankton to perform complex tasks by cooperation of the individuals; this approach can be considered as an application of Swarm Intelligence model [G. Beni and J. Wang, Proceed. NATO Advanced Workshop on Robots and Biological Systems, 1989] for dealing with the peculiarities of the harsh underwater environment.

Combinatorial optimization and self-disciplined computing by amoeba-based neurocomputer

Masashi Aono and Masahiko Hara

RIKEN
masashi.aono@riken.jp

Creating a biocomputer with its hardware incorporating biological materials, is it possible to implement some unique functions that are difficult for conventional digital computers to deal with? A living organism is a hierarchically-structured system in which a number of self-organization processes run simultaneously on its multiple levels with their characteristic spatiotemporal scales. Because a self-organization process at each level involves a certain kind of benefit optimization such as energy minimization and stability maximization, it would be sound to assume that an organism is a particular kind of concurrent computing system in which a number of computing processes to solve different benefit optimization problems run concurrently by sharing common computational resources such as the energy and substances. Despite a lack of predefined decision program, if these hierarchically-intervened optimization processes are capable of making a self-disciplined decision, for example, a decision to accept a loss in short-term benefits of its local part for the sake of long-term gains of its global body, the decision capability may be exploited for discovering some unprogrammed but reasonable optimization criteria when incorporated in a biocomputer.

With this expectation, we created a computing system incorporating an amoeboid unicellular organism, a true slime mold *Physarum polycephalum*, known to exhibit rich spatiotemporal oscillatory behavior and sophisticated computational capabilities. Introducing an optical feedback according to a recurrent neural network model, we lead the amoeba's photosensitive branches to expand or shrink within a network-type chamber in search of a solution to the traveling salesman problem (TSP).

Here we demonstrate our system's high optimization capability of solving four-city TSP. Our system reaches and stabilizes an optimal solution, as the amoeba having photoavoidance changes its shape in search for the most stable configuration allowing the amoeba to maximize its body area while minimizing the risk of being illuminated.

Intriguingly, the maintained stabilizing mode of the solution, however, spontaneously switches to the destabilizing mode without any explicit external perturbation. Contrary to the photoavoidance, the amoeba starts to destabilize the once-reached solution by spontaneously expanding its branch under illumination, and restarts the solution-searching process. Consequently, our system finds multiple solutions by repeatedly switching between the stabilizing and destabilizing modes.

As long as the amoeba maintains the photoavoidance and stabilizes the solution without changing its shape, the amoeba is stuck in a stalemated situation eliminating any possibility of nutrient acquisition. However, the amoeba spontaneously takes a risk of being illuminated locally and temporally to restart its shape change. It may be possible to view this spontaneous behavior as implying biological systems' capability of self-disciplined decision to put their resources available at present into risky investments to target resource acquisitions in the future.

We speculate that the spontaneous destabilization occurs due to the existence of chaotic dynamics capable of amplifying tiny fluctuations in a microscopic level to affect the unstable shape change in a macroscopic level. Indeed, applying several nonlinear time series analysis to the amoeba's oscillatory movements, we obtained results suggesting that an individual amoeba might be characterized as a set of coupled chaotic oscillators.

Additionally, we present a new technique that we call "autonomous meta-problem solving." In this approach, our system not only can solve a given problem but also can find new problems and then determine solutions in a self-disciplined manner, by exploiting the amoeba's unique searching ability and spontaneous behavior.

Artificial mental life

Xabier E. Barandiaran[1] and Ezequiel Di Paolo[2]

[1]Dept. of Logic and Philosophy of Science, University of the Basque Country, Spain
[2]Centre for Computational Neuroscience and Robotics, University of Sussex, UK
xabier@sindominio.net

Work in Artificial Life aimed at informing Artificial Intelligence (Steels & Brooks, 1994, *Artificial Life route to Artificial Intelligence*, Lawrence Erlbaum) has drawn inspiration from biology mainly at two levels: i) a bottom-up modelling approach conceiving cognition as the evolutionary complexification of adaptive behaviour, and ii) appeals to self-organization in the domain of behaviour and neural dynamics in analogy with self-organized chemical and biological processes. But little attention has been paid to the possibility of conceiving (and modelling) behaviour in terms of a self-maintaining organized unity in analogy with minimal forms of proto-cellular (or autopoietic) life. We propose that the behavioural counterpart of a network of self-sustaining chemical reactions should be a network of interactively maintained sensorimotor dissipative structures (*habits*) that emerge from the continuous reciprocal interaction between brain, body and world (and not, as in previous attempts, between molecular processes and neural processes, conceiving the nervous system as *operationally closed*Varela, 1979, *Principles of Biological Autonomy*, Elsevier).

Despite its popularity among pre-Darwinian biologists (such as Aristotle, Lamarck or Bichat), pragmatists and phenomenologists alike (Dewey, Merleau-Ponty) and among pre-computationalist psychologists (like James, Goldstein, Ivo Kohler or Piaget) the notion of habit has received little attention within Artificial Life. Habits posses key properties that make them extremely attractive for modelling the organization of behaviour: a) the structure of habits can be traced back to a fully operational-dynamicist framework, b) they do not presuppose a distinction or a causal priority between perception and action, c) habits are inherently situated or enactive structures cutting across brain, body and environment, d) habits are plastic and malleable, e) habits provide a concrete sense of self-maintenance (they are both cause and effect of their occurrence) potentially implying an intrinsic and a interactive teleology and f) habits can be nested or composed at different scales. This opens up the possibility for an operational notion of what might be called *Mental Life* (Barandiaran, 2007, *The World, the Mind and the Body*, p. 49, Imprint Academic) as the continued formation of a web of habits through sensorimotor interactions whose cohesive self-maintenance constitutes the identity of a *cognitive* (as opposed to barely biological) agent and the world it thereby co-defines.

We use some recent evolutionary robotic models on preference and habit formation (Di Paolo & Iizuka, 2008, *Biosystems*, 91, p. 409) to illustrate and explore the theoretical and philosophical implications of taking sensorimotor habits as the building blocks of behavioural organization. This organization takes the form of an attractor landscape whose stability is homeodynamically maintained through sensorimotor coupling. Mental Life opens up a new object of modelling in its own right, closer to the Aristotelian notion of psyche (or even the Heideggerian notion of Dasein) than to the notion of information processing, adaptive problem solving or weak conceptions of autonomy in robotics. Artificial Mental Life involves a shift from building artificial systems that satisfy externally imposed norms (engineering or evolutionary) to systems capable of generating their own norms: those required to sustain their own behavioural organization. In turn, it can become a source of new research questions to investigate the dynamics of assimilation and accommodation into an existing organization, its shaping by social interactions and institutions, or mental disorders dealing with stability, stress, identity, etc.

Ruggedness and evolvability — an evolution's-eye view

Lionel Barnett

Centre for Computational Neuroscience and Robotics, University of Sussex
lionelb@sussex.ac.uk

In both artificial and biological evolution, autocorrelation is commonly cited as a statistic which speaks to the "ruggedness" — and by implication evolvability — of a fitness landscape. But while the standard definition of autocorrelation involves *uniform* sampling of genotypes, it is a truism that evolution most decidedly does *not* sample a landscape uniformly. This is of particular significance in difficult artificial evolution problems, or indeed in natural evolution, where the vast majority of genotypes tend to be of poor or lethal fitness. On such landscapes uniform sampling is effectively biased towards precisely those (poor quality) genotypes which, from an evolutionary perspective, are of limited interest. To address this problem we suggest instead to take an"evolution's-eye" view of autocorrelation: that is, we let evolution itself do the sampling.

How are we to go about this? We note first of all that autocorrelation may be considered naturally in terms of *mutation*. Indeed, the significance of autocorrelation to evolutionary dynamics lies precisely in the (statistical) relationship between the fitness of parents and their mutant offspring. We thus propose that a more cogent and useful statistic is just the correlation between parent/mutant fitnesses *as sampled over the ensemble of evolutionary histories*. We argue that this alternative autocorrelation is both conceptually compelling and also practicable, in the sense of being amenable to finite sampling.

We note that our new statistic is no longer "evolutionarily agnostic"; rather, it is tightly bound to the dynamics of a particular evolutionary scenario. This, however, we regard as a strength. We can imagine, for example, that the same fitness landscape might "appear smoother" to one evolutionary algorithms than to another, thus providing insight into the suitability of a particular evolutionary algorithm to a particular problem in artificial evolution.

We also demonstrate how autocorrelation may be derived from the *mutant fitness distribution* — a finer-grained statistic — and we introduce the notion of *linear regressive* fitness landscapes. We illustrate our ideas with generalised NK landscapes, which are particularly tractable to analysis.

A functional account of minimal cellular life

Mark Bedau

Reed College
mab@reed.edu

Life seems to be one of the most fundamental categories in nature. But how exactly do material objects that are living differ from those that are not? Is there any a fundamental difference? And if so, is it a dichotomy or a matter of degree?

One answer to these questions abstracts away from chemical details and instead concentrates on living system's functional properties. In fact, the protocell research community more or less agrees that minimal cellular life forms are defined by chemically integrating three functionalities (Rasmussen et al., 2008, Protocells, p. 71). First, the system maintains an identity over time by localizing all its components, concentrating reagents and protecting key chemical reactions from molecular parasites and poisons. Second, it utilizes free energy from its environment to digest environmental resources in order to maintain and repair itself, to grow, and ultimately to reproduce. Third, these processes are under the control of inheritable information that can be modified during reproduction. The three functionalities mutually enable and support each other. They are collectively autonomous in the sense that they are created and sustained by the operation of the whole functional triad itself, rather than by any external governing agency.

Why should we believe that minimal cellular life is a chemically integrated functional triad of container (C), metabolism (M), and genetic program (P)? The rough consensus in the protocell community lends CMP view some weight, but not enough to convince skeptics. Other functionalities often associated with life–like reproduction, autonomous behavior, and sensitivity to the environmentcan be explained by the functional triad, which lends it further support. Going even further, the CMP view can be explained as a consequence of a more fundamental view according to which the essence of life is open-ended evolution. Elsewhere I have defended this view on the grounds that it best explains life's familiar hallmarks (Bedau, 1996, The Philosophy of Artificial Life, p. 332, Oxford UP) and puzzles (Bedau, 1998, Art. Life, 4, p. 125).

One puzzle about life concerns whether the distinction between life and non-life is dichotomous or continuous. The functional triad view implies that there is an array of thousands of different possible kinds of functional organizations, and they all more or less match the paradigm organization used to define life. Instances of some other functional organizations would be pretty clearly alive, and instances of others would be pretty clearly not alive, and a gray zone of further possible functional organizations separates those two clear cases.

One could divide the gray zone with any number of bright lines purporting to separate those systems that are "really alive" from those that are not, but I recommend not doing this. Instead, I think there is no deeper fact of the matter about the life/nonlife distinction other than the graded array of functional organizations. Attempting to find a more precise "definition" of life would be to invent a categorical distinction that does not exist in nature.

The arrow of complexity hypothesis

Mark Bedau

Reed College
mab@reed.edu

The evolution of the biosphere exhibits a trend of increasing complexity of the most complex organisms. Even though we are uncertain about the proper way to measure complexity, it is hard to deny the trend that the earliest prokaryotic cells are simpler than the eukaryotic cells that arose from them, and these were simpler than the multicellular life forms that evolved from them, and so on. But this trend is controversial to interpret and explain, and even to describe properly. Some think that the trend has for all intents and purposes already been explained. In contrast, I argue that the trend is not yet adequately explained but instead is a major remaining challenge in understanding the creativity of evolution.

Progress on this challenge is slowed in part because many people fail to realize that the explanation of life's complexity is still a mystery. Some people believe that natural selection given an infinite space of genetic possibilities will inevitably produce more and more complex adaptations. But soft artificial life models like Tierra, Avida, and Echo show conclusively that those mechanisms are in general insufficient to produce a trend of increasing complexity. The proof is simple: The models embody those mechanisms but they don't exhibit the requisite behavior. Mechanisms like natural selection in an infinite space of genetic possibilities might be necessary for explaining the trend, but they are not sufficient.

This implies that we need new concepts, theories, and models if it is to resolve the arrow of complexity hypothesis. Fortunately, soft artificial life models can be just the right tool for exploring answers to this question. But these models are not fool-proof. Some models beg the interesting questions, and others fail to produce the relevant behavior. So, proper use of these models requires care and experience. But in the right hands, they can provide a public, repeatable, and empirically grounded method for making incremental progress on the question of the creativity of evolution.

Artificial life as cancer research: embodied agent modelling of blood vessel growth in tumours

Katie Bentley, Paul Bates and Holger Gerhardt

Cancer Research UK
katie.bentley@cancer.org.uk

Tumours need to signal the growth of new blood vessels (angiogenesis) in order to obtain an oxygen supply and continue to grow. Angiogenesis in tumours, as opposed to normal tissue, generates abnormal, tortuous and leaky vessels. The vessels poor quality keeps oxygen levels low in the tumour, which keeps mutation rates high, and causes metastases to develop and spread across the body.

Angiogenesis as a process is a fascinating example of adaptive, environment-driven, morphogenesis of a spatial network; the most suitable and practical approach, and framework for simulation, was artificial life. Our multidisciplinary research aims to 1) understand the mechanisms of angiogenesis, 2) understand why the tumour environment causes abnormal vessels and 3) develop novel cancer therapies which could normalise tumour angiogenesis and thereby prevent metastases through reduction in hypoxia and increased genetic stability.

We have developed a multiscale agent-based model of a blood vessel interacting with its environment in order to investigate the effects that different environmental factors have on the initial stages of angiogenesis. The simulated endothelial cells in the vessel exist across multiple grid sites in a 3D gridded lattice. Each cell is comprised of many autonomous agents, representing sections of the cell membrane. In the first incarnation of the model, agents create new agents to change cell morphology. In the current version they move and are connected by springs, which realistically mimics membrane tension during cell migration.

Each agent communicates with its local environment, including other agents, to decide whether to activate receptors, release ligands and/or alter the cells local morphology. Overall a cell's behaviour and morphology emerge from the low-level interactions of its agents with the environment, which in turn then determines the vessel network morphology and development.

With this approach we have realistically modelled the initial stages of angiogenesis and made interesting predictions concerning abnormal endothelial cell fate determination in tumours, which are now being tested in the laboratory. The model is now being developed further to fully simulate cell migration and fusion of cells as the network develops.

Modelling spatial market dynamics with retail agents

Mark Birkin and Alison Heppenstall

University of Leeds
a.j.heppenstall@leeds.ac.uk

The power of agent-based modelling (ABM), when integrated with other AI-based and conventional approaches, can be greatly enhanced. The resulting hybrid systems offer a flexible modelling environment that exploits the benefits of the component methods. In particular, the ABM paradigm can be used to explore and understand systems that are governed by complex, non-linear relationships and self-organisation.

In earlier research, the authors have described an agent-based model of retail behaviour in which customer transactions are simulated using spatial interaction models (Heppenstall et al., 2005, Trans. in GIS, 9, p.35). The model has been used to simulate processes such as the diffusion of price changes through a retail network, and the interdependence of pricing behaviour between competing retail chains (Heppenstall et al., 2006, J. Artif. Societies & Soc. Simulation, 9).

In the research which is now presented, we draw an insight from an established method which explores the behaviour of retail provision when customer transactions are simulated by a spatial interaction model, but in which structural change is driven by a simple equilibrium-seeking mechanism. Established methods have provided useful insights into retail patterns under equilibrium-seeking behaviour, but have done relatively little to enrich our understanding of the dynamic processes and decisions from which change arises. Through the combination of these approaches, it is suggested that a much richer model architecture is possible, in which interacting retail agents produce a spatially heterogenous distribution of supply. This structure brings together coevolution in the economic and geographical variables (price and provision) through a dynamic model of competition amongst agents.

A series of numerical experiments are introduced to demonstrate how the use of agents can introduce more behaviour. The simulations are embedded in a real local retail environment. We evaluate the extent to which this work can be considered to present an improved understanding of this system.

Does coexistence solve the prebiotic information problem?

Sergio Branciamore[1], Walter de Back[2] and Enzo Gallori[1]

[1]Department of Evolutionary Biology, University of Florence, Italy
[2]Collegium Budapest, Institute for Advanced Study, Hungary
wdeback@colbud.hu

The information problem in prebiotic evolution arises from constraints on the amount of information that can be maintained in Darwinian evolution. The error threshold limits the transmissible length of a template under high mutation load. Existing solutions to the error threshold assume that the primitive genome consisted of multiple coexisting unlinked templates. Such coexistence requires a mechanism of cooperation to counterweigh competitive exclusion. Although much attention is given to ecological coexistence properties of cooperation mechanisms, little is known about the information carrying capacity of these systems under high mutation rates. Template coexistence may escape the error threshold, but it simultaneously raises the new problem of maintaining cooperation. Cooperation is threatened by the production of parasites through mutation. This results in an additional constraint on the information content in a template ensemble, which we call the 'parasite threshold'. If the parasite threshold of cooperative system is lower than the error threshold, template coexistence does not solve the prebiotic information problem.

We study the information carrying potential in a spatial eco-evolutionary model, based on the metabolic model (Czárán and Szathmáry, 2000). We define a surface in which each site is either empty or occupied by a molecule. A genome consists of d different templates of length l=i/d nucleotides, where i is the information content of the genome. We assume that replication of a template is only possible when all d functional template types are present in the local neighborhood. Mutation in a functional template results in a nonfunctional parasitic copy, with probability m per nucleotide. Growth and decay rates of of functional molecules and parasites are equal. Every reaction step is followed by diffusion of the molecules over the surface.

Genomes with varying number of fragments (d=1,...,4) are compared under high mutation load (m=0.01). We observed the parasite numbers and the maximum maintainable genomic information. Our results show that, surprisingly, the information storage capacity is highest for unfragmented, single replicator, genomes (d=1), despite their high production of parasite. While the number of parasites decreases with the number of templates, the vulnerability of fragmented genomes to parasitism sharply increases. Because the latter effect outweighs the former, the benefit of template coexistence is lost. When the different genomic strategies are put in direct competition with each other, fragmentation can out-compete the single template genome, but only in a situation with low mutational rate, and length-dependent growth rate. Close to the error threshold, however, fragmented genomes are competitively excluded by the single template strategy.

We conclude that template coexistence by itself does not solve the prebiotic information problem, because cooperative systems are limited by the 'parasite threshold'. We demonstrate that in the metabolic model, template coexistence does not increase information content and is excluded in direct competition. Although more realistic conditions concerning catalytic specificity, length-dependent neutrality and growth may refine these results, it is clear that limitations arising from cooperation must be taken into account in solving the information problem in prebiotic evolution.

How might group selection explain the major evolutionary transitions?

John Bryden

University of Leeds
johnb@comp.leeds.ac.uk

The central idea of group selection is that an individual will reduce its fitness so that the mean fitness of a group (of possibly non-related individuals) may increase. This is relevant to the major evolutionary transitions where an individual will cooperate by stopping reproduction on its own and reproduce instead as part of a group. Explaining the major evolutionary transitions should simply be a case of applying models of group selection. However, group selection may only work when there is a small difference between the fitness of a cooperator and a defector (Traulsen et al., 2006, Proc. Nat. Acad. Sci. U.S.A., 103, 10952).

A new approach (Bryden, 2008, PhD thesis, University of Leeds) sheds fresh light on this topic however. We must carefully look at what we mean by fitness and what we mean by group.

Recent perspectives on fitness (Metz et al., 1992, Trends Ecol. Evol., 7, p.198) argue that fitness should be calculated over a range of environments. This contrasts with the Hamiltonian perspective of fitness (Hamilton, 1964, J. Theor. Biol., 7, p.1) which is the number of adult offspring. By calculating over a range of environments, some traits which prosper in some environments decay in other environments. Tools are available (e.g., Tuljapurkar, 1990, Proc. Nat. Acad. Sci. U.S.A., 87, p.1139; Bryden 2008) for modelling long-run growth rates over varied environments.

To apply this long-run perspective on fitness to the major evolutionary transitions, resource allocation strategy modelling has been done by Bryden (2005, ECAL, p.551; 2007, ECAL, p.645;2008). The problem of the major evolutionary transitions is reformulated as a question as to whether an individual will invest resources in a higher reproductive process: a process of generating new offspring with two or more individuals having some genetic stake in, and contributing resources to, the new offspring. When an individual reproduces clonally, it will grow faster in favourable environments than those that contribute toward a higher reproductive process. However, reproduction can be risky and leave the fast reproducing lineage dangerously low on resources during unfavourable environments.

Analytic methods and computer simulations have shown how a strategy of collective reproduction (by sharing resources between several individuals and one offspring equally) can dominate a strategy of producing clonal offspring (Bryden 2007). When individuals are selfish and only contribute minimal resources, increases in the amplitude of environmental resource fluctuations becomes increasingly significant (Bryden 2008). The reason the shared strategy is successful is because the clonal strategy is very weak in harsh environments.

These results demonstrate that it is plausible that an individual may lower its Hamiltonian fitness (i.e., its reproductive output) to increase its long-run fitness by contributing to a higher reproductive process (such as those in the major evolutionary transitions). If I suggest a definition of a group as a lineage that is temporally spread out across several environmental eras, rather than all being present at the same time, we may then compare the long-run fitnesses of lineages to determine the most successful. In other words, the group that has the greatest long-run fitness is selected for. This theory calls for verification through scientific experiments and expansion through further modelling of the major evolutionary transitions.

Dynamical complexity of spatially embedded networks

Christopher Buckley[1], Lionel Barnett[2] and Seth Bullock[1]

[1]Science and Engineering of Natural Systems group, School of Electronics and Computer Science, University of Southampton
[2]Centre for Computational Neuroscience and Robotics, University of Sussex
clb05r@ecs.soton.ac.uk

Living systems are embedded within physical space. While this embedding can be viewed as a restrictive constraint on structure, where the prohibitive costs of establishing or maintaining interactions over long distances mitigate against certain kinds of potential organisation, it can also be seen as an enabling factor, bringing about correlations, regularities and symmetries that can be exploited by evolution. Artificial life research on spatially embedded games, ecologies, networks, evolution, and agents has shown that projecting a well-mixed system into a low-dimensional medium and constraining interactions to be local can confer interesting properties (e.g., stability, honesty, robustness to parasites) that are otherwise absent or unstable. This paper explores the question: what is the contribution of spatial embedding to the dynamical complexity of networks.

In previous work some of us have developed a general framework for characterising the impact of spatial constraints on network topology (Barnett et al., Phys. Rev. E 76, 056115, 2007), and some of us have explored the dynamical complexity of spatially embedded artificial neural networks (Buckley & Bullock, ECAL 2007). Here we combine these two threads to discover what graph theoretic properties of networks confer high dynamical complexity, and to explore the extent to which spatial embedding tends to encourage exactly these topological properties in networks that are random in other respects.

We first return to the original formulation of the dynamical complexity measure due to Tononi, Sporns and Edelman (PNAS 91, 5033, 1994) and correct an error in a widely used approximation of this measure. This correction impacts on intuitions about the structural and functional roots of dynamical complexity. However, we are able to rescue these intuitions by re-deriving the approximation for a continuous-time dynamical system rather than the discrete dynamical system used in the original formulism. This process emphasises some key differences between the dynamics of continuous and discrete dynamical systems.

We then go on to derive and extend a graph theoretical interpretation of dynamical complexity for the corrected discrete measure and the new continuous measure. This allows us to strengthen our understanding of the relationship between properties of spatially embedded structures and high complexity. In particular, we are able concretise the notion that the structural contribution of spatial embedding to high dynamical complexity results from the introduction of cycles of connectivity at many structural scales. Furthermore, we are to able to address a misconceived equivalence between the small world property and systems of high "dynamical complexity". Specifically, we find that while systems of high dynamical complexity may possess the small world property, neither property is either necessary or sufficient for the other.

Learning drives the accumulation of adaptive complexity in simulated evolution

Mikhail Burtsev[1], Konstantin Anokhin[2] and Patrick Bateson[3]

[1]M.V.Keldysh Institute of Applied Mathematics, 125047, Moscow, Russia
[2]P.K.Anokhin Institute of Normal Physiology, 125009, Moscow, Russia
[3]Sub-Department of Animal Behaviour, University of Cambridge, Cambridge, CB3 8AA, U.K.
mbur@ya.ru

Developmental plasticity and particularly learning enable organisms to cope with new environmental challenges. But if learning is costly, the same behavior could evolve through Darwinian modifications of development that substitute for the role of learning in the acquisition of that behavior, a hypothesis known as the Baldwin effect. Computer simulations have confirmed that learning accelerates evolutionary adaptation to a single problem posed by the environment. What has not been shown, however, is the way in which the driving force of learning can generate ever greater complexity in organization of evolved behavior, the one that has very small chance to appear in one step in the course of evolution. Here we report such consequences of the role of learning using a model fitted with sequentially appended adaptive systems.

A central component of our approach is that of a functional system. It asserts that each adaptive behavior is executed by a distributed system of phenotypic elements that cooperate towards organism's fitness in this particular task. Novel challenges lead to generation of new adaptive functional systems that can be added to the existing ones either by evolution of development or by learning. This allows the establishment of elaborate behavior patterns and results in increased complexity of organisms at the systems level. Selection assesses organisms by the adaptiveness of their functional systems and the more functions the individual possesses, the more competitive we assume it is. The greater complexity in our model implies more extensive repertoire of behaviors supported by greater amounts of equipment for monitoring and coping with the environment. From a biological standpoint an organism with the higher functional complexity will be better able to deal with a variety of challenges from the environment and therefore will be more likely to survive. If considered from the perspective of a single ecological challenge requiring just one functional system our model is similar to a classical single-peaked landscape simulation by Hinton & Nowlan. However, the main highlight of the model is its operation in a complex evolutionary landscape similar to the "Royal Staircase" fitness function of van Nimwegen & Crutchfield, which allowed us to examine coordinated evolution of multiple functional systems under the impact of learning and developmental plasticity.

The results of simulations demonstrate that ability to learn dramatically accelerates the evolutionary accumulation of adaptive systems in model organisms with relatively low rates of mutation. The growth of complexity is mediated through a process of allelic substitutions that simulate emergence of evolutionary predispositions for learning of certain behaviors and simultaneously release organisms capacities for acquisition of next tasks. The effect of learning on evolutionary growth of complexity is even greater when the number of elements required for adaptive system is increased. These results suggest that as the difficulty of challenges from the environment become greater, so learning exerts an ever more powerful role in meeting those challenges and in opening up new avenues for subsequent genetic evolution of complex adaptations.

Dynamically adapting parasite virulence to combat coevolutionary disengagement

John Cartlidge

University of Central Lancashire
jpcartlidge@uclan.ac.uk

Participating in an evolutionary arms-race, natural coevolutionary predator-prey and host-parasite systems often exhibit accelerated evolution. Competitive Coevolutionary Genetic Algorithms (CCGAs) attempt to harness this evolutionary acceleration by engaging (multiple) evolving populations in competitive self-play.

By evaluating individuals competitively, CCGAs afford the possibility of tackling problems that are ill-defined, open-ended and lacking in formalism. This offers CCGAs a potential advantage over more traditional Genetic Algorithms (GAs) when fitness evaluation is difficult to operationally define. Analogously, one imagines that it is much easier to formally define the rules of the game of tennis than it is to define tennis playing ability. In practice, however, defining an *appropriate* game is often non-trivial.

Competitive evaluation leaves CCGAs susceptible to some adverse evolutionary dynamics. One such hindrance is "disengagement". This occurs when one coevolving population gets the upper hand and begins to easily outperform the other. Since it becomes impossible to discriminate between individuals according to ability, the selection gradient disappears and the coevolving populations begin to stagnate. The result is a stymied system that is left to flounder aimlessly.

To prevent disengagement, the author has previously introduced the "Reduced Virulence" technique (Cartlidge & Bullock, 2004, Evol. Comp., 12, p.193). This technique helps avoid disengagement by reigning in a population that inherits an advantageous bias. Rather than reward individuals that maximally damage a competitor, Reduced Virulence favors individuals that give opponents a chance. Perhaps counter-intuitively, Reduced Virulence enables accelerated evolutionary progress by disadvantaging a population's most successful individuals.

In this work, Reduced Virulence undergoes a rigorous sensitivity analysis in the Counting Ones domain (introduced by Watson & Pollack, 2001, GECCO, p.702, Morgan Kaufmann); an analytically tractable substrate designed to highlight the dynamics of coevolution. Following intuition, it is shown that for optimal performance, virulence should be increasingly reduced as the asymmetrical bias (and thus likelihood of disengagement) between coevolving populations increases. Interestingly, even when coevolution is unbiased, "Maximum Virulence" — equivalent to the canonical fitness evaluation of "reward all victories" — is shown not to be ideal. Thus, results suggest that (in the Counting Ones domain) when population sizes are small, it is never the case that the canonical coevolutionary setup should be favored. The generality of this result, however, is an open question.

Utilizing this information, a novel "Dynamic Virulence" algorithm is introduced. This algorithm adapts population virulence over time as populations evolve. It is shown that Dynamic Virulence is able to cope with varying bias better than fixed virulence and allows the discovery of optimal solutions under a much wider range of conditions than any individual fixed virulence setting.

Finally, it is discussed how analyzing the role of virulence in artificial systems may allow us to better understand virulence in nature. For instance, perhaps there is potential for a "Reduced Virulence" approach to tackling infectious diseases. Rather than killing mosquitoes to eradicate malaria, one could alternatively encourage malaria-resistant strains that are better able to survive.

A measure for natural selection's contribution to the origins and maintenance of organismal complexity

Alastair Channon

University of Keele
alastair@channon.net

The aim of Artificial Life research into Open-Ended Evolution is, initially at least, to develop artificial evolutionary systems in which new adaptive traits continue to evolve and the maximum complexity of organisms, ecosystems or behaviours continues to increase. The main proponents of this approach have presented systems that invoke natural or biotic selection, as opposed to artificial or abiotic selection, as the drive for both the generation of new adaptive traits and (potentially) a sustained increase in complexity. However, within both Biology and Artificial Life, doubts have been raised as to natural selection's role as the drive for increasing complexity (Lynch, 2007, Proc. Natl. Acad. Sci., 104, p.8597; Miconi, 2008, Artif. Life, 14, in press), with the suggestion put forward that nonadaptive evolutionary forces (such as mutation, recombination and genetic drift) or mathematical/statistical constraints may be the primary drives, through either a passive increase in variance of complexity in the presence of a lower bound, or a constraint-driven drive toward complexity. The question therefore arises, how to determine natural selection's contribution to increases in complexity?

This work introduces first a measure for the phenotypic complexity of an individual, based on the class of components previously used in the population-level analysis of an Artificial Life system classified as exhibiting unbounded evolutionary dynamics (Channon, 2006, Genet. Program. and Evolvable Machines, 7, p.253): components that are approximately equivalent to the biological notion of a gene or coding DNA; and second a measure for the contribution made by selection to increases in complexity, based on the mechanism and methodology used in the development and application of component-normalised activity statistics to that system. These measures enable us to address the question posed above, about a fundamental aspect of evolution in general, in a way that would not be possible given the biological world alone. They also provide a mechanism for detecting increases in phenotypic complexity and attributing them to either adaptive or nonadaptive forces.

Results from the application of the measures to evolution in runs of the above system suggest that, according to these definitions, natural selection initially (but only briefly) opposes the level of increases in complexity (new active/coding DNA) that would be brought about by the nonadaptive forces alone, presumably because the new active/coding DNA would be nonadaptive; but that as evolution progresses, natural selection maintains and drives the increase in adaptive complexity with remarkable consistency: natural selection can (and does, in this system at least) provide a sustained drive toward increasing complexity.

An adaptive model of marine biogeochemistry in the Archaean

James Clark and Hywel Williams

University of East Anglia
h.williams@uea.ac.uk

Here we describe ongoing work that applies adaptive modelling techniques from artificial life to open questions in Earth system science. Understanding the Earth system in terms of global chemical cycling is of critical importance for the interpretation of Earth history and the prediction of climate change. Many of the key chemical reactions that facilitate biogeochemical cycling occur during the metabolism of organisms. The range of metabolic reactions present in the biota has changed significantly over evolutionary timescales, often with dramatic effects on the Earth system. Yet models of biogeochemistry have traditionally only included static representations of the biotic components of the major nutrient cycles. More recently, some models have begun to include a number of functional types of organism, each with different prescribed biogeochemical properties. However, none of these models address the dynamic adaptation of the biota over time, which can change the way in which different species interact and alter their effects on biogeochemical cycling. Here we present a new adaptive individual-based model of the marine ecosystem in the Archaean period of Earth history. We use a simplified version of the major physical chemical processes and metabolic functions that are known to have existed at this time; the marine ecosystem during the Archaean is in any case simpler than the modern ecosystem, being solely based on microbial life. We specifiy a number of microbial guilds containing species that have similar metabolic reactions and biogeochemical functions, e.g., photosynthesisers, chemoheterotrophs, etc. Within each guild, multiple species may coexist and compete on the basis of various physiological traits. Individual microbes each have a genotype that specifies various metabolic traits and thus determines their growth rate as a function of their environment. We consider each model individual to represent an aggregation of many genetically similar real world individuals; this assumption allows us to study phenomena on a global scale. Successful microbes (i.e., those that are well suited to their environment) grow and reproduce, while unsuccessful microbes starve and die. Mutation can occur during reproduction, allowing the creation of new species. Competition for nutrients drives ongoing adaptation that dynamically changes the chemical environment in a coevolutionary loop of interaction. Our model considers a vertical column separated into three compartments representing the deep and surface ocean layers and the atmosphere. We find that diverse self-sustaining ecosystems emerge over time without being prescribed, and that the distribution of nutrients and organisms between the three compartments is qualitatively similar to that believed of the Archaean. In particular, photosynthesisers tend to dominate the sunlit surface ocean, fixing inorganic carbon (CO_2) into organic forms that support other populations, while the deep ocean ecology is dominated by methanogens, which are able to survive in the dark and anoxic deep ocean conditions. Adaptation of nutrient uptake and light sensitivity traits creates species within each guild that are optimally suited to their environment. Recent work has looked at evolutionary trade-offs and the possibility of a biological trigger for the Great Oxidation event. This work represents the first step in a greater program of study that will seek to model evolutionary adaptation in marine biogeochemistry.

Phenotype-based evolution of complex food webs

Walter de Back[1], Sergio Branciamore[2] and George Kampis[1]

[1]Collegium Budapest, Institute for Advanced Study, Hungary
[2]Department of Evolutionary Biology, University of Florence, Italy
wdeback@colbud.hu

Explaining the relation between structure and dynamic of food webs remains one of the most daunting challenges in ecological theory. The trophic structure is a fundamental property of ecological communities that relates their diversity to productivity and stability. Understanding the generic properties of food webs requires insight in the way these networks are assembled. Although ecological communities are often augmented by invasion rather than adaptive speciation, the diversity in an ecological network can ultimately only be explained from an evolutionary perspective. Recently, several aggrerate population models for the evolution of food webs have been proposed. Here, we present an individual-based model in which we show the evolutionary emergence of complex trophic networks from the bottom-up.

We use a spatial individual-based model with a nonlinear mapping between genotype and phenotype. An individual is specified by a genotype that determines its phenotype in a redundant ("many-to-one") and epistatic ("one-to-many") fashion. All behavioral/ecological properties of the individual (e.g. reproduction/mortality rate, auto/heterotrophy, prey preference, etc.) are derived from specific aspects of the phenotype, such that trade-offs in ecological function are inherently introduced. Autotrophs only consume abiotic resources, while heterotrophs can consume individuals in their spatial neighborhood. The outcome of the consumption interactions depends on prey preference and the distance between both phenotypes. Mortality and (asexual) reproduction are based on energy that decreases linearly, and is increased by consumption. Mutations occur by substitutions in the genotype with a low probability per locus.

The system is inoculated with a single genotype coding for a random autotroph. After the appearance of heterotrophs, coevolution between auto- and heterotrophs causes phenotypic diversification. The number of species (i.e. individuals with same phenotype) varies around 40-60 for the given system size (+/- 75000 individuals), of which approximately half are heterotrophs. The evolved species show a large variation in life history and consumption patterns, balancing various trade-offs. The structure and composition of the network can continually change, although evolutionary stability increases over time. We assess the ecological stability of the evolved food webs by canceling mutation. After a partial collapse, the truncated food web (with typically 10-15 species) persists over an indefinitely long period, during which it shows chaotic population dynamics.

We demonstrate the evolutionary emergence of ecological stable food webs in an individual-based model. The genotype-phenotype mapping provides efficiency and robustness in the exploration of phenotype space, while spatial interactions stabilizes population dynamics and preserves diversity. By encapsulating the ecological properties and their trade-offs in the phenotypes, rather than defining them as global state variables, allows for the niche differentiation by which food webs are assembled.

Life in time: the missing temporal dimension in autopoiesis

Ezequiel Di Paolo

University of Sussex
ezequiel@sussex.ac.uk

There is a widespread view in the artificial life community that life is not so much about materiality but about organization. However, one of the favourite candidate theories that explain what this organization should be like has an ambivalent position with respect to materiality. This is the theory or autopoiesis (Maturana and Varela, 1980, Autopoiesis and cognition, D. Reidel). Accordingly, autopoiesis is the self-production of a unity in the domain of processes of material construction, transformation and destruction. The identity of the living system, in this view, is sustained in its form in spite (or rather because) of its material flux.

This notion seems to imply an inherent temporal dimension to autopoiesis. It is, intuitively, a dynamical concept. However, because of its insistence on a set-theoretic focus on conservation of autopoiesis, the theory says little about the temporality of life (leaving relevant phenomena such as stress, fatigue, pathologies and development untouched).

Efforts have been made to bring thermodynamic material constraints into the theory of autopoiesis (Ruiz-Mirazo and Moreno, 2004, Artificial Life, 10, p. 235). The implications are not trivial, nor are they foreseen in the theory as postulated in the original literature and interpreted purely in formal terms. One immediate consequence of realising autopoietic organizations in dissipative physical structures far from equilibrium is that there is an obvious time arrow introduced into the process of life: the arrow of thermodynamics. Bare autopoiesis, paradoxically, does not present us with a similar time arrow (a time-reversal thought experiment leads to this conclusion).

But this temporality belongs to the nature of dissipative processes and is in some sense only inherited by life because such material processes constitute it.

I shall argue that living systems enjoy a different kind of temporality, given by their own interactive and teleological organization. This temporality is richer and different from that of the background time's arrow — it is a consequence of expanding the theory of bare autopoiesis with the notion of adaptivity, (Di Paolo, 2005, Phenomenology and the Cognitive Sciences, 4, p. 429). This temporality is characterized by intentional direction, minimal granularity, rhythmicity, and the presence of historical transitions (in behaviour and development). It belongs to the organization of an adaptive autopoietic system under precarious circumstances.

It is conceivable that this temporality could emerge even if the system were not subject to thermodynamical constraints since it is a consequence of the higher order relations between interactive and constitutive aspects of self-maintenance. It is also conceivable for aspects of the temporality of life to contradict the temporality of the thermodynamics time's arrow, since the temporality of life is inherently related to its intentionality. Even at a minimal level, a living system may retroactively alter the virtual possibilities of the past through its sense-making activity in the present.

Chemical organizations in living systems

Peter Dittrich[1], Florian Centler[2], Christoph Kaleta[3], Naoki Matsumaru[3] and Pietro Speroni di Fenizio[4]

[1]Friedrich Schiller University Jena
[2]UFZ Leipzig
[3]FSU Jena
[4]DCU Dublin
dittrich@minet.uni-jena.de

Complex dynamical reaction networks consisting of many components that interact and produce each other are difficult to understand, especially, when new components may appear over time. In this talk, I outline a theory, which has been inspired by artificial chemistry research, to deal with such systems. It has been sucessully applied to regulated metabolic networks, virus-immunsystem dynamics, chemical information processing, chemical evolution, and planetary athmosphere photochemistries. I will show how the approach can be used to predict growth phenotypes and to evaluate the quality of large bio-models. The theory consists of two parts. The first part introduces the concept of a chemical organization as a closed and self-maintaining set of components. This concept allows to map a complex (reaction) network to its set of organizations. The theory provides a new view on the system's "organizational structure", which is fundamentally different from a pathway-oriented view. The second part of the approach connects dynamics with the set of organizations, providing a link to classical dynamical systems theory, e.g., by mapping a movement of the system in state space to a movement in the set of organizations. It is shown that every dynamically stable state must be an instance of an organization.

Strategies for maintaining large robot communities

Stephen English[1], Jeffrey Gough[1], Alexis Johnson[1], Robert Spanton[1], Joanna Sun[1], Richard Crowder[1] and Klaus-Peter Zauner[2]

[1]School of Electronics and Computer Science, University of Southampton, United Kingdom
[2]Science and Engineering of Natural Systems group, School of Electronics and Computer Science, University of Southampton
kpz@ecs.soton.ac.uk

The confluence of progress in micro-actuators, power sources, and mixed-signal microelectronics have recently moved swarm robotics and robot communities from simulation to reality. Swarms of 20 to 100 robots are in use already, implementations with several hundred robots are practicable, and communities exceeding a thousand robots are certainly conceivable. Such large robotic communities provide platforms for numerous exciting research directions including collaborative swarms and self-reconfiguring structures.

Maintaining hundreds of robots, however, poses significant practical challenges. The literature on strategies for maintaining software and hardware in large robot communities is sparse, even if applicable concepts from wireless sensor-networks are included. Crucial for the viability of any such strategy is its impact on cost per robot.

To provide a realistic setting we introduce a robot platform designed to be fabricated in full on standard printed circuit board (PCB) assembly lines. In this context we introduce a framework for on-line testing and calibration based on code pieces, termed plasmids, that migrate among the micro-controllers of the robots. The proposed approach allows the robots access to a larger library of code then what could be stored locally.

The robot consists of a single PCB that doubles as chassis and contains no custom mechanical components. Inexpensive motors (mass produced to vibrate mobile phones) are directly soldered to the circuit board and used in direct drive. Our prototypes use a 200 mAh rechargeable lithium polymer battery giving the robot over an hour of autonomy while moving at a speedy 1 m/s. An MSP430F2131 microcontroller controls the robot and communicates with neighbouring robots via infrared light. The simplicity of the design allows the entire robot to be assembled with low-cost PCB manufacturing techniques and is well suited for small-scale mass production of several hundred robots.

While this design significantly reduces the current cost barrier to obtaining a robot swarm, it also shifts the attention to the practical problem of maintaining hundreds of robots. Recharging batteries, sieving out robots with worn tyres or accidental damage is one aspect. A second aspect is testing and calibration. It can not be performed in the PCB assembly process and cost considerations prevent proprioceptive sensor. Collaboration among robots to verify performance and provide feedback (e.g., drift direction during a run and return) provide a scalable alternative. A third aspect is the maintenance of software in the robot community.

Our plasmid framework addresses all three aspects with a design that is lightweight enough to run on the microcontrollers. Pieces of code and associated attributes (version, target number for redistribution, lifetime, conditions for transmission) are propagated among robots that meet. For example, the code may perform a test on the robot and require to be forwarded to four other robots that have not encountered this test, before it is deleted. Such test plasmids traverse the robot community which, in its collective memory,contains and executes more code than would fit within the program memory of a single device.

Abstract organization and material parts in ALife and synthetic biology

Arantza Etxeberria

University of the Basque Country
arantza.etxeberria@ehu.es

Two decades after the initial steps of ALife another field, Synthetic Biology (SynBio), defends the need of synthetic methodologies in the life sciences. Both have used similar arguments: the epistemological principle that to understand how something works, we must know how to build it, and the instrumental goal of producing useful things. But they come from different traditions and constitute different scientific communities; probably ALife's closest ancestor is cybernetics, whereas SynBio's is molecular biology.

In the case of ALife, the main goal is to study living organization. For Langton (1996 Artificial Life, MIT Press.), synthesis makes possible to explore life-as-it-could-be in order to understand the necessary and the contingent of living organization in principle in a materiality different from the carbon-based. For others, the stress is in characterising life as an autonomous organization. ALife, like SynBio, is very diverse, and although it has predominantly pursued to create "life-like behaviours within computers and other artificial media", occasionally ALife models have been developed in vitro with biochemical components. Also, although the main goal was to construct new forms of life, in practice the field has produced more models (with scientific purposes) and tools (with instrumental purposes) than real instantiations.

In SynBio, however, the goal is not organisation, but design. The field intends to construct engineered organisms (biofacts) out of the components of existing life, by changing specific parts. Like ALife, SynBio is very diverse; O'Malley et al. (2008, BioEssays, 30, p. 57) have distinguished three different approaches: DNA-based device construction, genome-driven cell engineering, and protocell creation. Because it is difficult to see what they have in common, we may consider that the first is the most characteristic so far, or at least we could say that it is the one sharing less with ALife (whereas there is certainly some overlapping in the case of the third).

This paper intends to analyse some epistemological similarities and differences of both fields, especially in what concerns their views on living organization and the importance of materiality. In what concerns the first, in ALife living organization has been considered as an invariant to be found/constructed, whereas in SynBio, the goal is to engineer or to create life (sometimes close to the field of the origins of life). One idea has been that "nature is imperfect and should and can be revised and improved" (cited in Morange, 2008, Unpublished manuscript on Synthetic Biology). In what concerns the second, ALife aims to understand parts, composition and function as emerging properties, thus avoids fixed parts and aims at construction; in contrast, SynBio uses existing parts to change the design of life (one major effort is to build an open-access library of presynthesized biological parts and devices, the *Registry of Standard Biological Parts*), aiming at intervention.

Developing a methodology for social network sampling

Daniel W. Franks[1], Richard James[2], Jason Noble[3] and Graeme Ruxton[4]

[1]University of York
[2]University of Bath
[3]University of Southampton
[4]University of Glasgow
df525@york.ac.uk

Researchers are increasingly turning to network theory to describe and understand the social nature of animal populations. To make use of the statistical tools of network theory, ecologists need to gather relational data, typically by sampling the social relations of a population of animals over a given time-period. Due to effort constraints and the practical difficulty involved in tracking animals, these sampled relational data are almost always a subset of the actual network. Measurements of the sample – such as average path length, clustering, and assortativity – are assumed to be informative as to the structure of the real-world social network. However, this assumption is problematic. Due to artefacts of the sampling process, the various network measures taken on the sample may be biased estimators of the true values. For example, just as we would get a biased estimate of mean human height by selecting for a sample those people who stood out in a crowd, we will get a biased estimate of a measure like mean connectivity if we sample individuals who are socially prominent.

This problem can only be solved by developing a qualitative theory of network sampling, answering questions such as what proportion of the whole network needs to be sampled before a given level of accuracy is achieved, and what sampling procedures are least biased? To develop such a theory, we need to be able to generate networks from which to sample. Ideally, we need to perform a systematic study of sampling protocols on different known network structures. But currently available data on animal social networks are unsuitable as these networks were themselves sampled.

The simulation methods of artificial life provide the way forward. We have developed a computational tool for generating artificial social networks that have user-defined distributions for network properties (such as the number of nodes, and the density) and for key the measures of interest to ecologists (such as the average degree, average path length, clustering, betweenness, and assortativity). This tool allows us to perform the required systematic analyses of the biases inherent in different sampling regimes (e.g., snowball sampling) applied to different network structures. We will present details of this system, and show we are using it to develop robust sampling methods for social network data. We see the system as the first in a series of works that will allow us to develop a qualitative theory of social network sampling to aid ecologists, and eventually social scientists, in their social network data collection.

Utility and experimental testability of the Gaia hypothesis

Andrew Free, Amanda McNeil, Olivia Mozley and Rosalind Allen

University of Edinburgh
Andrew.Free@ed.ac.uk

Gaia theory describes the life-environment system of the Earth as stable and self-regulating. It has remained at the fringes of mainstream biological science owing to historical definition problems and its apparent incompatibility with individual-level natural selection. However, various bodies of ecological and evolutionary research suggest ways in which the biosphere might tend towards stability and self-regulation. Here we review this research, relate the results to a plausible and informative formulation of 'Gaia theory', and ask how the theory extends the perspectives offered by these disciplines.

We then address the question of how Gaia theory might be tested experimentally. Such tests require the (reasonable) assumptions that life, where it evolves, will exploit essentially all thermodynamically-feasible forms of metabolism, and that Gaian regulation should be possible with a purely microbial biosphere. The biosphere is a closed system driven by solar radiation, and we describe here a laboratory microcosm which is an appropriate analogue of such a system. We then describe our preliminary experimental results from characterisation of this system, and discuss how we will use advanced molecular techniques employed by modern microbial ecology, in conjunction with computer simulations of inter-species interactions, to study the system and answer questions of relevance to Gaia theory. We also describe how this combined experimental-simulation approach can be applied to many questions of evolutionary and ecological interest which lie within the research areas bounded by Gaia.

CAT: a market design competition

Enrico Gerding[1], Peter McBurney[2], Jinzhong Niu[3] and Simon Parsons[3]

[1]University of Southampton
[2]University of Liverpool
[3]City University of New York
eg@ecs.soton.ac.uk

In the past decade the market paradigm has been adopted as the prima facie candidate for achieving coordination between self-interested, autonomous software agents. In this context, the area of mechanism design has been increasingly applied in order to achieve desired global outcomes based on local interactions, by developing rules of interaction that align the individual needs with social desiderata. Crucially, most of the current literature considers these mechanisms or markets in isolation and ignores any competition between them. In practice, however, we see similar markets competing, including Internet auction sites such as eBay and Amazon, and stock exchanges from around the world. Similarly, we argue that competition needs to be considered within multi-agent systems when designing mechanisms for them.

Against this background, we have developed an international market-design competition called CAT (short for catallactics, the science of exchanges) where each entrant to the competition sets the rules of interaction, as well as a fee structure. Each market consists of a double auction exchange where, similar to a stock exchange, the role of the market is to match buyers with sellers. Our competition framework includes the set of trading agents who participate in these markets, and who will choose the market which has proven the most profitable. Given this, the goal of the market designer is to attract profitable traders and make profits by charging appropriate fees. This can be done by changing the market rules and fees dynamically as a response to changing market condition. Overall, the objective of the competition is to see whether dynamic markets outperform static ones, and to study the types of markets that emerge from such complex interactions.

The CAT market design competition is an international competition and is part of the trading-agent competition (TAC). The first competition was successfully held in conjunction with AAAI 2007, and this year's competition is going to be held in conjunction with AAAI 2008.

The gene-function relationship in the metabolism of yeast and digital organisms

Philip Gerlee[1], Torbjorn Lundh[2], Bing Zhang[3] and Alexander Anderson[4]

[1]Niels Bohr Institute
[2]Chalmers and University of Gothenburg
[3]Vanderbilt University
[4]University of Dundee
gerlee@maths.dundee.ac.uk

Many natural and artificial systems form structures which can best be viewed as networks consisting of a set of nodes and links connecting the nodes. This perspective has been helpful in elucidating the organisation of a variety of systems ranging from power-grids and the internet to protein interactions, gene regulation and cell metabolism. Many of these networks exhibit a scale-free degree distribution and therefore deviate from the classical description of complex networks which predicts a Poisson degree distribution, which for degrees larger than the average degree scales as an exponential distribution.

We have studied the metabolic gene-function network in yeast and digital organisms from the artificial life platform Avida. The gene-function network is a bipartite network in which a link exists between a gene and a function (pathway) if that function depends on that gene, and can also be viewed as a decomposition of the more traditional functional gene networks, where two genes are linked if they share any function. We show that the gene-function network exhibits two distinct degree distributions: the gene degree distribution is scale-free while the pathway distribution is exponential. This is true for both yeast and digital organisms which suggests that this is a general property of evolving systems. One possible explanation for this structure is that in the network the genes acquire new links according to preferential attachment while the pathways receive new links independent of their degree.

This hypothesis was tested in Avida by tracking the evolution of the gene-function network in repeated simulations and measuring the rate of link attachment. Here the single commands takes the role of genes and the functions are the evolved boolean functions for which the organisms are rewarded. The results show indeed that the a gene is more likely to become involved in new functions (i.e. increase its degree) the more links it already has. The link attachment of the functions on the other hand occurred independent of the degree. In real cells it is known that gene duplication is the main mechanism by which new genes are created. If the two genes would retain similar functionality then we would expect pathways which involve many genes to increase their degree. This is contradicted by the exponential degree distribution and the observations in Avida and suggests that the rate of the gene divergence in yeast must be high. The duplication of pathways/functions could on the other hand explain the scale-free distribution of the genes, and this mechanism has already been observed in Avida. Measuring the overlap between different pathways in terms of the genes which constitute them, showed that this also is a likely mechanism in yeast evolution. In conclusion we have presented a new way of analysing the gene-function dependence which sheds new light on the evolution of genes and functionality, and suggests that function duplication could be an important mechanism in evolution.

In silico evolution of chemotaxis

Richard Goldstein[1] and Orkun Soyer[2]

[1]NIMR
[2]CoSBi
richard.goldstein@nimr.mrc.ac.uk

Bacteria are able to sense their environment and move towards sources of attractant and away from sources of repellent through the process of chemotaxis. The best understood model for chemotaxis comes from Escherichia coli, where the biochemical pathways have been extensively studied. In E. coli, the bacterium switches between swimming and tumbling based on the changes in the local concentration of attractant. It is unclear, however, how similar the behaviour and biochemical mechanisms are for other organisms. Work is proceeding on evaluating the chemotaxic behaviour of a number of different bacteria, indicating substantial differences with E. coli. Even in E. coli, the fact that bacteria 'gutted' of most of the chemotaxic machinery still displays effective chemotaxis indicates that there are still unanswered questions even in this organism. Finally, there are issues about how this particular strategy and implementation have evolved. Was this only one of a number of possible strategies? How did the strategy depend upon the environment, the properties of biochemical networks, or on the evolutionary process? Would similar strategies result under different conditions?

Ideally we would like to take non-chemotaxic bacteria and evolve them to perform chemotaxis under a variety of different conditions, a daunting and lengthy experiment. In contrast, we can do this easily in a virtual world. In addition, we can keep complete records of the evolutionary process as it occurs. We create a population of digital organisms that move in a virtual world of attractant. The organisms contain a rudimentary set of biochemical elements, that is, sensors of the external attractant concentration, a reversible motor that can cause the bacteria to tumble or swim, and a set of proteins that can be activated, and while activated, have the potential to activate or deactivate other proteins. We then allow the biochemical network to evolve, selecting those digital bacteria that better co-localise with the attractant to reproduce for the next generation. This allows us to combine molecular-level evolutionary dynamics with phenotype-level selection of a relevant fitness parameter in an exactly specified environment. We find that these digital organisms quickly are able to display effective chemotaxis. Interestingly, the dominant mechanism is one that is very different from that observed in E. coli, but is similar to that observed in other bacteria as well as in gutted E. coli. The required network can be extremely simple, and can be fulfilled by coupling the bacteria's metabolic network to the regulatory network, suggesting an explanation for the behaviour of gutted E. coli as well as suggesting a possible evolutionary route to how chemotaxis first arose.

Immune-inspired networked service delivery

David Gowans, Paul Marrow and Richard Tateson

BT Group plc
david.2.gowans@bt.com

The expansion in distributed computing capabilities has led to a need to deliver services at different locations in networks, where demand is unpredictable. Giving all nodes in a distributed system the potential to deliver any service is likely to be a waste of resources, while extreme specialization on the part of individual nodes is likely to incur large overheads in terms of messaging and service transmission across the network. Other Computer Science researchers have drawn inspiration from one of real life's most adaptable distributed systems, the vertebrate immune system, to attempt to solve problems in a variety of different application areas (de Castro et al., 2003, Soft Comp., 7, 526). We describe an immune system-inspired method for service management in a distributed network services scenario. Each node in the network runs our system and the actions of all these local instances mesh together to provide overall service management.

The self/non-self theory of activation of the vertebrate immune system suggests how it can generate a response to non-self entities, protecting the organism concerned. Our system uses an analogous activation sequence system to respond to requests arriving unpredictably at different nodes in a network, and enables the efficient delivery of responses to the requesting node without requiring complete specialization of capabilities in each node. In our system we have focused on the network of stimulatory interactions amongst antigens, antibodies, T-Cells and B-Cells. Requests for services are represented by antigens which interact with elements (nominally representing T- and B-Cells) in a two stage activation process to produce fully activated B-Cells. The fully activated B-Cells are monitored by Service Runners to indicate the level of demand in the system. The activation of B-Cells also releases antibodies which act as adverts for services. We believe that reproducing the logic of this interaction network, along with biologically plausible parameters for longevity and diffusion rates of the various cell types, give us an artificial system which can adapt to service demand patterns in the same way that the immune system adapts to patterns of antigenic challenge. A key issue is the level of complexity in our system. It aims to be useful, with some of the advantages of the analogous natural system, without attempting to model it slavishly. Our simulations show that an adequate level of complexity was chosen, as simpler incarnations lost some desirable properties, whereas more complex implementations would have led the system more towards modelling the immune system than focusing on service delivery.

In the simple network simulations described here some of the major issues for distributed systems are successfully counteracted. Unevenly distributed demand (which would otherwise result in excess demand on particular nodes) is balanced across suitable processing nodes. We see a memory effect whereby nodes can build up a more rapid response to requests based on their history of responses, allowing nodes to become specialists at dealing with particular request types. The diffusion of 'cells' deals with the issue of locating a suitable node to satisfy a request, even across several network hops. The large number of cells in the network originating from many different nodes, provides a level of fault tolerance, directing requests to alternative nodes in the event of node failure. We look forward to testing the system in larger, and more realistic, distributed network services scenarios.

From market to non-market: an autonomous agent approach to central planning

Dan Greenwood

University of Westminster
D.Greenwood2@westminster.ac.uk

In the longstanding debate in political economy about the feasibility of socialism, the Austrian school of economists have argued that markets are an indispensable means of evaluating goods, hence a prerequisite for productive efficiency. From an Austrian perspective, the prices generated by markets neatly encapsulate, in terms of a single numerical unit, highly complex information about the relative levels of demand and supply for different goods. Furthermore, it is emphasised that prices enable both consumers and producers to discover new economic knowledge about more efficient ways of attaining their particular ends. The Austrians contend that, such is the complexity of economic choices facing producers, no adequate level of economic efficiency could be achieved in the absence of markets, even on the assumption that a set of production objectives has been agreed. This problem of productive calculation is referred to as the 'economic calculation problem' for socialism.

Socialist advocates of a non-market economy have yet to provide a satisfactory response to this Austrian argument for the indispensability of markets. Some have sought to develop computational solutions to the economic calculation problem using techniques such as linear programming. Yet the computational models proposed are strongly influenced by the equilibrium model of neoclassical economics. From an Austrian perspective, these models overlook the essence of the calculation problem by assuming the availability of knowledge which can only be acquired through the market process itself.

The debate in political economy about the feasibility of a computational solution to the problem of non-market calculation has not yet considered the recent emergence of agent-based systems and their applications to resource allocation problems. Agent-based simulations of market exchange offer a promising approach to fulfilling the dynamic functions of knowledge encapsulation and discovery in a decentralised way, as the Austrians show to be performed by markets. Further research is needed in order to develop an agent-based approach to the economic calculation problem. Given that the macro-level objectives of agent-based systems can be easily engineered, it is suggested that such an approach holds the potential to become a desirable alternative to the real markets that the Austrians favour.

How epigenetic evolution can guide genetic evolution

Graeme Groom[1], Rob Mills[2] and Richard Watson[2]

[1]School of Electronics and Computer Science, University of Southampton
[2]Science and Engineering of Natural Systems group, School of Electronics and Computer Science, University of Southampton
raw@ecs.soton.ac.uk

The expression level of a gene in future generations can be modified both by genetic mutations and by the attachment of methyl groups to the DNA. Since the DNA methylation pattern along a genome is inherited, methylation patterns constitute a significant epigenetic inheritance mechanism that is subject to evolution by natural selection. The variation rate of methylation patterns is generally higher than that of DNA which suggests that evolution of methylation patterns might be more rapid than that of genetic evolution. But, common consequences of methylation, such as reduced expression of methylated genes, could also be produced by genetic changes and these would have higher heritability. The question we address in this work is how the evolution of epigenetic methylation-dependent phenotypes might interact with the evolution of genetic DNA-determined phenotypes.

There is no biological mechanism known to directly transfer methyl groups into equivalent DNA changes. However, in principle an indirect mechanism could cause evolved methylation patterns to enable the subsequent evolution of equivalent genetic patterns in a manner analogous to the Baldwin effect (Baldwin, Am. Nat., 30:441-451, 1896; Jablonka et al, TREE, 13:206-210, 1998). The Baldwin effect describes how non-heritable acquired characteristics can influence the evolution of equivalent genetic characteristics without any direct Lamarckian inheritance of acquired characters. This occurs because the ability to acquire or learn a new behaviour changes the selective pressures acting on genetic changes. Specifically, genetic changes that support this behaviour, e.g. by reducing learning time by making a small part of the behaviour genetically innate, may be selected for when the learning mechanism is present even though these same genetic changes may not be selected for when the learning mechanism is absent. Over generations, the modified selection pressures so produced can cause genetic assimilation of a phenotype that was previously acquired, even to the extent of making the acquisition mechanism subsequently redundant. Thus a learned behaviour can guide the evolution of an equivalent innate behaviour (Hinton & Nowlan, Complex Systems, 1: 495-502, 1987).

In the Baldwin effect a rapid mechanism of lifetime adaptation guides the relatively slow genetic evolution of the same behaviour. By analogy, Jablonka et al have suggested that "genetic adaptations may be guided by heritable induced or learnt phenotypic adaptations". Here we hypothesise that "inherited epigenetic variations may be able to 'hold' an adapted state for long enough to allow similar genetic variations to catch up", as they put it, even if the epigenetic variations are not induced or learnt but simply evolved by natural selection on methylation patterns. We assume that an individual may only express one phenotype in its lifetime, but that a given genome will persist relatively unchanged on a timescale that allows its methylome to adapt by natural selection. Thus, in contrast to the Baldwin effect, in this case two mechanisms of evolution by natural selection are coupled — one acting at a different variation rate from the other. We present a simple model to illustrate how a rapidly evolving methylome can guide a slowly evolving but highly-heritable genome. This is used to show that methylome evolution can enable genetic evolution to cross fitness valleys that would otherwise require multiple genetic changes that were each selected against. This finding suggests that the relatively rapid evolution of methylation patterns can produce novel phenotypes that are subsequently genetically assimilated in DNA evolution without direct transfer or appeal to induced phenotypes. This can enable the genetic evolution of new phenotypes that would not be found by genetic evolution alone, even if methylation is not significant in the ultimate phenotype.

Chemical basis for minimal cognition

Martin M. Hanczyc[1] and Takashi Ikegami[2]

[1] University of Southern Denmark
[2]University of Tokyo
ikeg@sacral.c.u-tokyo.ac.jp

We have developed a simple chemical system capable of self-movement in order to study the chemical-molecular origins of movement, perception and cognition. The system consists simply of an oil droplet in an aqueous environment. The aqueous phase contains a surfactant that modulates the interfacial tension between the drop of oil and its environment. We embed a chemical reaction in the oil phase that reacts with water when an oily precursor comes in contact with the water phase at the liquid-liquid interface. This reaction not only powers the droplet to move in the aqueous phase but also allows for sustained movement. The direction of the movement is governed by a self-generated pH gradient that surrounds the droplet. In addition this self-generated gradient can be overridden by an externally imposed pH gradient, and therefore the direction of droplet motion may be controlled. Also we noticed that convection flow is generated inside the oil droplet to cause the movement, which was also confirmed by simulating the fluid dynamics integrated with chemical reactions (Matsuno et al., 2007, ACAL 07, Springer, p.179, Springer). We can observe that the droplet senses the gradient in the environment (either internally generated or externally imposed) and moves predictably within the gradient as a form of primitive chemotaxis (Hanczyc, M., et al., 2007, J. Am. Chem. Soc., 129, p. 9386).

By creating a pH gradient and concomitant convection flow, the droplet behaves as if it can perceive the environment. We believe that the geometry of the interface shape can control sensitivity to the environment (Ikegami et al., 2008, BioSys., 91, p.388). This geometry-induced fluctuation is the source of fluctuation of motion, which we think is tightly linked with the idea of biological autonomy. There is empirical evidence to support the above ideas. Some form of internal bias is necessary for breaking symmetry to cause self-movement and the bias may be the result of perception of the environment.

Such simple oil droplet systems show autonomy in the sense that the droplets move in response to the self-generated pH and the environmental gradient. In our modeling, we demonstrated that a computational autopoietic cell could move by continuously self-repairing the membrane, but in this case failed to show any gradient-climbing behavior (Suzuki et al., 2008, Artificial Life, in press). This may be due to the fact that the autopoietic cell can only survive in the narrow range of environments that support a certain substrate density. Compared with that autopoietic cell model, our oil droplets are more stable and they strive to maintain their boundary structures. We hypothesize that the pH gradient around the droplet results in an unbalanced interfacial tension at the interface. The droplet then responds by motion in order to maintain a balanced interfacial tension. Once the tension forces around the droplet are balanced the droplet would stop moving. In this way, we contend that a kind of homeostasis is a basis for self-movement. Different from the mere physical-chemical process, any life system preserves its own identity and consistency with respect to the environment. This homeostasis, rooted on the sensory motor couplings, will organize minimal cognition (see also, Ikegami , T. et al., 2008, BioSys., 91, p.388]

Homeostasis via chaos: implementing the uniselector as a dynamical system

Inman Harvey

University of Sussex
inmanh@gmail.com

Ashby's Homeostat (Ashby, 1952, Design for a Brain, Chapman and Hall) was a demonstration of how an extended form of homeostasis, defined by him as ultrastability, could be achieved with a relatively simple mechanism. Homeostasis refers to the process whereby an organism, or a machine, actively maintains certain 'essential variables' (EVs) within the critical bounds of viability. The simplest form is negative feedback, but a higher order of homeostasis can sometimes be observed when an EV, on approaching a critical value, triggers one or more periods of *positive* feedback that reorder the dynamics until a new stable equilibrium (based once again on negative feedback) is found. This ultrastability can be viewed as an interaction between two coupled dynamical systems (DS): the primary DS comprises the EVs, and their direct, parameterised interactions; the secondary DS only kicks in temporarily when the EVs of the first are threatened, and then it alters the parameters of the first DS until some equilibrium is found that no longer threatens the EVs. Hence this is a form of selection between multiple possible steady states.

In Ashby's Homeostat, the secondary DS was implemented by the 'Uniselector'. Under normal circumstances it maintained a fixed set of parameters for the first DS. When it was triggered, it picked a different set of parameters (in practice drawn from a lookup table of random numbers), and continued doing so until the triggering factor ceased. In Evolutionary Robotics one common method for designing an artificial 'nervous system', coming from the DS perspective on cognition, is to evolve the parameters (weights, biases and time constants) for a Continuous Time Recurrent Neural Network (CTRNN; Beer, 2006, Neur. Comp. 18(12). p. 3009). One way of implementing an Ashbyan ultrastability mechanism would be to incorporate the Uniselector as an add-on to the CTRNN. An alternative approach proposed here is to incorporate the Uniselector-effects within the CTRNN, rather than as a separate add-on.

We require a very large number of different attractors (corresponding to different sets of random numbers in the Uniselector); and a trigger mechanism that initiates random or chaotic jumps to a new attractor. This can be done with a core of just 3 interconnected variables, equivalent to 3 nodes of a CTRNN if we extend the class of transfer functions at each node to include sine waves as well as sigmoids. Drawing on a result of Thomas (Kaufman et al., 2003, C. R. Biologies 326, p. 205), we show how this can be implemented; we can switch between chaotic 'search' and settling into one amongst many possible attractors. These attractors are cyclic or strange, but can be used to set parameters for the remaining part of the CTRNN that comprises the 'primary DS'. There remain practical issues, somewhat glossed over by Ashby, in orchestrating how long is spent 'evaluating' each attractor visited before abandoning it for another one.

This approach demonstrates the possibility of composing a Homeostat entirely of such an (extended) CTRNN, with the Uniselector-substitute as a distinct hand-designed sub-circuit or module. Further evolution can maintain the desired ultrastable characteristics, whilst relaxing these architectural constraints of modularity.

Animatronic model of a human tongue

Robin Hofe and Roger K. Moore

Department of Computer Science, University of Sheffield
R.Hofe@sheffield.ac.uk

Contemporary speech technology struggles with the variation that occurs in natural speech. In order to explain such variation, researchers frequently refer to processes of energy optimisation that govern speech production. Although there is plenty of qualitative evidence, quantitative data about speech energetics are still sparse because it is difficult to acquire them from human subjects. To overcome this problem, an animatronic model of a human tongue and vocal tract (AnTon) was designed. Human anatomy provided the guideline for its construction; functional considerations were made only when an approximation using available technology proved impossible or infeasible. Thus, the behaviour of the model derives from, and is grounded in, its anatomy. The tongue model presented here was developed using these 'biomimetic' principles. The human tongue consists almost completely of interwoven muscle fibres whose topology allows for complex movements. The incompressibility of muscle tissue is an important prerequisite, a property it shares with water; such structures are therefore called 'muscular hydrostats'. The soft silicon that forms the artificial tongue body approximates incompressibility. Muscles are represented by filaments that run along paths resembling real muscle fibre orientation. Wherever muscle fibres follow a curved path, regularly spaced glass beads prevent filaments from cutting into the silicon. The filaments connect to meshes that are embedded into the tongue body and serve both as an attachment point and to distribute force evenly. The current tongue model comprises four of the main tongue muscles, represented by eleven filaments that are attached to servo motors. It is connected to a movable jaw and a hyoid bone; the latter is a horseshoe-shaped bone that supports the tongue root and is situated directly above the Adam's apple. AnTon is able to imitate a range of oral gestures and will be used for sound production as soon as the anterior part of the vocal tract is completed. Apart from studying speech energetics, AnTon has the potential to become a general tool for speech research, education, and speech therapy. A video is available on the AnTon project's website: http://www.dcs.shef.ac.uk/ robin/anton/anton.html.

Simulating active touch with a simple embodied agent

Hiroyuki Iizuka[1] and Takashi Ikegami[2]

[1]Osaka University
[2]Tokyo University
ezca@sacral.c.u-tokyo.ac.jp

In the present paper, we simulate the differentiation of sensory- and motor-like interfaces, and their dynamics in an agent of novel architecture. This paper reconsiders the boundary between an agent and its environment. In the standard framework, we tend to assume that the physical boundary between the agent and its environment exists due to the independent physical devices of its sensory and motor interfaces. The boundary between agent and environment thus appears as a static and rigid boundary. We contest this view by creating a simple simulation model in which this boundary can vary dynamically. Thus we argue that an agent's perception is an outgrowth of the complex interference between the efferent and afferent copies of its action patterns.

The agent consists of a body with two straight arms that move freely from -90 to 90 degrees. Those arms are controlled by a continuous-time neural network. The neurons connected to the arms sometimes receive body states from the arms (the afferent copy) and then will move the arm in order to explore the environment (efferent copy). The role of sensory system and motor system are thus switching temporally, and there is no explicit sensory-motor flow.

There are some neuro-psychological experiments supporting the above dynamic view. For example, Yamamoto and Kitazawa (2001) demonstrated that with their arm-crossing experiment in which the perceived temporal ordering of haptic stimuli on a subject's hands is reversed when successive stimuli are temporally very close together. This experiment implies that we may be able to explain our perception by something based on body image rather than just by reactions to sensory input.

The agent in this simulation is trained via a standard genetic algorithm. The agent was intended to learn to distinguish between fans with differing numbers of wings which the agent can touch and manipulate using its arms. From this experiment we draw two primary conclusions:

1) The two arms become differentiated into sensor-like and motor-like arms, and the agent can successfully distinguish different fans by touching them

2) The agent's active and passive touching of the fans is driven by the stability of dynamic attractors associated with those touching behaviours. By placing time delays into the neural connection from the arm neurons to the body neurons, we notice that motor-like arms appear more fragile than sensor-like arms, and the attractor associated with active touch is more fragile than that associated with passive touch.

Our simulation demonstrated that the differentiation of sensory and motor functions emerged by mixing free and constrained arm motion conditions. When there are no obstacles, the arm moves freely and the neural network activations maintain coherence. This coherence could be taken as a sensation of body image. When there are obstacles present, however, this coherence is lost, and this interruption is transmitted as sensory information. By investigating the coherence/decoherence events of these internal neural dynamics, we will argue that the interference between efferent and afferent sensory copies drives sensation.

Adaptive fault tolerance in wireless sensor networks

David Irons[1], Fabrice Saffre[2] and Chris Cannings[1]

[1]University of Sheffield
[2]British Telecom
d.irons@sheffield.ac.uk

Systems including wireless sensor networks and amorphous computing involve scattering a large number of nodes (sensors / processors) over a geographical area and then creating local connections to carry out some global objective. For example, monitoring, positioning and reporting back on changes in environmental features; such as heat, light, sound and / or motion. Geographical, cost or size constraints may also imply that the nodes need to be dispersed semi-randomly, implying a great deal of uncertainty about their relative positions and the exact size of the area being covered.

Various research challenges exist in these systems, such as creating strategies for self organisation, energy management, fault tolerance, maintaining security and adapting to cope with geographical features (e.g. walls, hills, buildings). Moreover, these challenges become increasingly important as technological advances lead to smaller and smaller nodes, making them (potentially) more error prone and difficult to position accurately. In this study, we introduce a fault tolerance / energy management strategy that allows the system to robustly self re-organise, when nodes break or run out of power. We then test this strategy on different geographical landscapes.

In order to model these sensors networks, we use random geometric graphs. Here, n nodes are randomly placed on a surface and edges can form between nodes if they are within distance R of one another. Additionally, we assume that each node has a randomly assigned lifespan corresponding to time taken for the node to run out of power / break down.

We then test the following strategy. Suppose each node can take one of two states, 'active' or 'hibernated', with the hibernated state consuming less power. Additionally, suppose any active node is capable of sending all other nodes within distance r ¡ R into hibernation. Now, beginning with every node in hibernation, the system evolves by individual hibernated nodes (asynchronously) trying to become active (i.e. turning active if there are no existing active nodes within radius r). Whenever the lifespan of an active node is exceeded, it disappears from the system, allowing the opportunity for hibernated nodes to take its place.

In order to score the integrity of the system, we compare the evolution of such networks to an equivalent system where all nodes are active. In particular, we compare various statistical properties of the network at regular time intervals; including (1) the proportion of active nodes in the largest component, (2) diameter and (3) area covered by the largest component. Such performance measures were chosen since these systems may need to adequately monitor a geographical area (over a period of time), with network connectivity being essential for communication of data / results. We find that, for a large range of parameters, the new strategy presented here is more robust, in that the integrity of the system is maintained over a longer period of time. In particular, robustness increases as the initial number of nodes (n) increases. We believe that as nodes become cheaper to mass produce, having a large number is a feasible strategy for these systems.

As well as testing our strategy on flat two dimensional surfaces, we also extend our approach to more complicated three dimensional geographical landscapes with the additional constraint that nodes can only communicate if they are in line of sight of one another.

Solving the division of labour problem using stigmergy and evolved heterogeneity

Emyr James, Jason Noble and Richard Watson

Science and Engineering of Natural Systems group, School of Electronics and Computer Science,
University of Southampton
ej07r@ecs.soton.ac.uk

Evolving cooperative teams is a research area with applications in the fields of robotics and software agents. Progress on this problem could also help us to understand the evolution of cooperation in natural systems such as the social insects. The overarching question is how cooperative teams should be represented in order to promote efficient evolutionary search. More specifically, what should serve as our basic unit of selection — the individual or the team? — and how can the division-of-labour problem be solved? In order to answer these questions we have taken a benchmark problem from the genetic programming (GP) literature, the artificial ant problem, and extended it so that teams of ants must cooperate to complete the task.

In this model, the ants are centrally placed in a bounded grid with each square containing food. The goal of the team is to harvest all the food in the environment in as few moves as possible. In the initial version of the problem, the members of the team are all clones, each having exactly the same GP controller program. Many solutions will have poor performance as the team members will all behave in the same way, and will therefore fail to cover the grid efficiently. To perform better, the ants must evolve to take advantage of stigmergic interactions to break the symmetry of the problem and clear the world of food efficiently. This division of labour through stigmergy is indeed what is seen to evolve during the simulations.

A further extension is made by assigning each member of the team an identity tag, and adding the ability to execute different subtrees of the cloned controller based on this tag. When these operations are allowed, higher fitnesses are achieved than with the purely stigmergic situation above.

During evolution, selection acts at the team level. We can therefore view the members of the team as being equivalent to cells in a multicellular organism. The identity branching operation is analogous to cell differentiation within this abstract organism. Using this scheme, the degree of differentiation is not specified a-priori and is controllable through evolution. This allows the full continuum from purely homogeneous teams to entirely heterogeneous teams to be expressed. There is also the potential to use this method as a way of measuring the degree to which a task demands heterogeneous solutions.

The relative importance of stigmergy and innate heterogeneity in achieving the necessary division of labour were compared with a third experimental manipulation. The ability to influence each other stigmergically was removed by placing each ant in it's own world and tallying the pieces of food consumed by the team as a whole. In this scenario, the most efficient way to tackle the problem is for the team to evolve complete heterogeneity.

We conclude that the division-of-labour problem in the evolution of cooperative teams can be solved by both stigmergic communication and innate heterogeneity. Furthermore, the technique of allowing the level of heterogeneity of the team to be open to selection shows promise for future work.

Nonconsumable resources facilitate complex evolution

George Kampis, Laszlo Gulyas and Walter de Back

Collegium Budapest
gkampis@colbud.hu

Open ended evolution (OEE) is a problem closely related to the "arrow of complexity", that is, the temporal emergence of complex and complicated structures and organisms in a progressive, never-ending fashion. A straightforward, and biologically proven means to approach this is to integrate organisms in complex environments, and to make them parts of an evolving food web where various aspects of complexity may increase. Our recent work, reported in a series of papers, addresses OEE via the emergence of food webs and niche differentiation in a simple agent based population that starts from a seed such as a single species of producers (consuming nonreplicating resources and forming the basic trophic level of an ecosystem).

A notorious problem of such systems (as exeplified by recent work like DOVE, Webworld, etc) is the questionable ecological stability of the evolving trophic structure. To stabilize even a 3-species system by parameter tuning is an egregious task, let alone in systems of higher complexity and with the frequent introduction of newer species. A typical problem is that either a newly evolved consumer cannot grow (thus cannot become part of the system) or depletes the resources until both die out. Consequently, few systems currently can handle this problem. To breed restraint e.g. by selecting for moderate predators is a slow process which already presupposes the coexistence of species in a food web. But how do we get there?

Density dependent feedback (such as functional response) could be a viable solution at this point, but that typically requires complicated agents or an alien hand. An additional mechanism is to consider genotype-phenotype maps with tradeoffs in consumption, replication, and other items of life history. This is a road we have reported in another paper submitted to this conference.

Here we introduce and study a different idea, that of nonconsumable resources (NCRs), an idea which is usually underestimated in the evolutionary modeling of food webs and OEE. NCRs are factors required for the life history but not destroyed by the agents; NCRs can range from space to nesting ground to other abiotic factors to sexual partners to environmentally inherited properties to other species entering mutualism or (mild) parasitism with the given species. In natural systems NCRs are found everywhere and they pose important, yet little understood feedbacks on otherwise destabilizing dynamics. Using our agent ecosystem we show that phenotype to phenotype interactions in a consistently agent based ("fully embedded") system naturally introduce NCRs dynamically. This, in turn, helps stabilizing the emerging ecosystems and permits complexification steps otherwise impossible due to kinetic instabilities.

This work is part of our development of the FATINT system and is part of a research project reported in "Feedback Self-Organization", Springer, to be published in 2009. This work was supported by the EC grant QosCosGrid IST FP6 #033883. The authors thank Collegium Budapest for their hospitality. L.G. acknowledges the partial support of the GVOP-3.2.2-2004.07-005/3.0 (ELTE Informatics Cooperative Research and Education Center) grant of the Hungarian Government.

Intentional and causal urges: when to ascribe life or mind to artificial systems?

Fred Keijzer

University of Groningen
f.a.keijzer@rug.nl

How do we decide whether or when Artificial Intelligence systems are cognitive or Artificial Life models are alive? Despite the many different ways in which these decisions are made and specific criteria are used, in both cases the final verdict has a strong intuitive component: The Turing Test is a way to harness the decision in a particular way; but why would making a specific impression on a human observer be a good criterion for deciding whether an AI system is cognitive? Similarly, why would a CA model of growth be considered a possible form of life? In both cases, many of us have tendencies to interpret mind-like or life-like phenomena as being actual example of mind or life. Still, many of us also have tendencies to question the step from looking like mind or life to actually being a mind or alive. The result has been a wavering attitude in which both tendencies coexist, while a more definite view on this issue remains elusive. Research on the automatic tendency to detect causality and animacy provides a way to make sense of this wavering attitude. The upshot of this research is that we have an inbuilt intentional urge: a prereflective tendency to categorize certain objects or displays as alive and even intentional. We also have a causal urge: a prereflective tendency to interpret events with specific visual features in causal terms. Both urges seem to be mutually exclusive. These automatic reactions are subsequently elaborated and scrutinized in a more reflective mode, which can lead to either the acceptance or rejection of the first automatic reaction. This research can have important and wide-reaching implications for discussions on the status of artificial cases of life and mind. Foremost, it would explain the wavering attitude when it comes to formulating criteria for making decisions either way. Strong intuitive judgements and reflection do not necessarily coincide. It seems clear that we should be more critical of our own intuitive judgements in this area. At the same time, it is less evident how exactly we can come to a more critical and ultimately sounder judgement on these matters. At present, it seems best just to start out with trying to sort out the options and issues. For example, a way out would be by differentiating between design research in which AI and ALife would fall and descriptive and explanatory research where empirical cognitive science and biology belong. Another important issue would be whether the intuitive mutual exclusiveness of causal and intentional interpretations is really to be trusted and should be taken to imply a serious dichotomy in the natural world. In this talk, I will start out with the problem of ascribing life and mind, introduce the literature on the intentional and causal urges, and focus on the possible implications for artificial life and mind.

The effects of local information and trading opportunities in a network constrained economy

Daniel Ladley[1] and Seth Bullock[2]

[1]University of Leeds
[2]Science and Engineering of Natural Systems group, School of Electronics and Computer Science, University of Southampton
danl@comp.leeds.ac.uk

Work within the field of artificial life has as history of exploring the ways in which locally constrained interactions between the elements of a system can give rise to organised behaviour at the level of the ensemble. Here we study the effect of constraining co-operative, competitive and communicative interactions within a market by embedding it within a network. We are particularly interested in how these different kinds of interaction are influenced by the structure of the market network. The paper aims to examine the effect of limited trading opportunities and information availability on the behaviour of individuals and of the market as a whole. It examines how a trader's ability to make profit is influenced by their location within a trade network and how trader strategy must be adapted to cope with this constraint. To this end we employ an agent-based model of trader interaction in which the actions of each trader are governed by individual behavioural rules. Traders are situated on the nodes of a network and interact with potential trading partners through the ties. The networks considered in this work are constructed via preferential attachment schemes resulting in networks both with and without positive assortedness. The behavioural rules of the traders are optimised for their respective locations within the networks through the use of a hill-climbing algorithm. It is demonstrated that a trader's ability to profit and to identify the equilibrium price is positively correlated with its degree of connectivity within the market. Better connected traders are able to exploit their market position at the expense of other market participants. When the effects of constraining trade and information are separated it is demonstrated that when traders differ in their number of potential trading partners, well-connected traders are found to benefit from aggressive trading behaviour. A higher number of potential trading partners allows these traders to demand better terms as there is a higher chance of another trader being willing to trade with them. Where information propagation is constrained by the topology of the trade network, connectedness affects the nature of the strategies employed. Better connected traders attempt to learn more quickly, taking in as much information as possible at the start of the market in order to exploit possible trading opportunities. Less well connected traders learn more slowly and average over time to avoid being exploited by better connected individuals. We also demonstrate that traders are unable to exploit second order information and trade effects connected to the network. We show that it is not possible for traders to modulate their price or the way in which they weight information based on the connectedness of the potential trading partner/information source to make higher profits. When this situation is permitted all traders adopt strategies such that none benefit from the additional abilities.

Towards mental life as it could be: a robot with imagination

Hugo Marques[1], Owen Holland[1], Rob Knight[2] and Richard Newcombe[3]

[1]University of Essex
[2]The Robot Studio
[3]Imperial College
hgmarq@essex.ac.uk

"Artificial life is the study of man-made systems that exhibit behaviors characteristic of natural living systems." So wrote Chris Langton two decades ago, and most people within the discipline still seem happy with his definition. But from the human perspective, the behaviour of our own living system is only half the story, because what matters to most of us is not our behaviour, but our experience — our mental life. Perhaps it is now time for artificial life to embrace the challenge of creating man-made systems that have experiences 'characteristic of natural living systems' and giving an account of 'mental life as it could be'.

This talk will describe some of our recent work within the area known as machine consciousness. Several modern theories of consciousness, notably those of Damasio and Metzinger, stress the importance of embodiment for consciousness, and identify a body-centred self-structure as being a key element in the origin and support of phenomenal experience. In the CRONOS project, we reasoned that human consciousness — the only kind of which we have knowledge — must require both a human-like body, and a human-like body model, and so we set out to construct suitable entities. The physical robot, CRONOS (Holland and Knight 2006), is a new kind of robot: it is anthropomimetic, faithfully copying the human skeleton, and equipped with appropriately placed elastic muscles and tendons. The body model, SIMNOS (Gamez et al. 2006), is an accurate physics based simulation that behaves sufficiently similarly to CRONOS to be controlled by the same control system.

For a self-model to have evolved, it must confer some functional advantage on its host. We are investigating the hypothesis that one useful function might be the ability to support imagination by predicting the outcomes of possible actions through internal simulation, more or less along the lines suggested by Hesslow (2002). Simulating the body and its controller is only the first step; it is also necessary to create an internal representation of the real world that interacts with the body sufficiently accurately to have predictive value. A physics based internal representation meets this requirement, but the problem of creating such a representation goes well beyond the traditional AI/computer vision challenge of achieving correct geometry, since the objects in the modelled world must also have the correct physical properties. We will report on the use of SLAM techniques and cross-modal integration for dealing with this problem.

In order to obtain functional benefit from imagination, it is not enough merely to have predictively valid models of the body and of the world; it is also necessary to embed the models within a suitable architecture to identify when imagination is required, to generate appropriate candidate actions, to evaluate the success, failure, and costs of the actions, to prevent candidate actions from being acted out by the real robot, to select the preferred action, and to execute the preferred action. We have developed a task-dependent taxonomy of these architectures, and have implemented and tested the ones that seem most important (Marques and Holland 2008). For the first time, this talk will describe and demonstrate an anthropomimetic robot using an architecture for embodied imagination, including a physics based internal model of itself and the world, to select and execute an appropriate action; we believe this may be a significant step towards building a robot with a form of mental life.

Dimensions of adaptivity

Marek McGann

MIC, University of Limerick
marek.mcgann@mic.ul.ie

Artificial Life offers strong tools for making coherent and disciplined our understanding of the relationships between several fundamental concepts within Cognitive Science more generally such as cognition, value, function, learning and others. One concept which may act as an anchor for this set of ideas is that of adaptivity, in its intraorganismic sense of a system being able to change to maintain itself in the face environmental challenges. A-Life, along with many other areas in Cognitive Science, examines different forms of adaptivity, though rarely in an explicit fashion. We therefore miss many opportunities to analyse the concept of adaptivity itself, and what it might tell us about the relationship between life and mind, learning and value, and other issues within this constellation of notions. The present paper will attempt to outline some of the general characteristics of adapativity, with the particular aim of identifying what distinguishes different forms of adaptivity (for example, homeostatic regulation and operant learning) and how such dimensions might be used to help organise and direct our thinking on the matter in future. Crucial elements include the timescale over which the adaptive mechanism operates (e.g. achievement of reward in an operant learning task versus strategic play in chess), the inertia of those mechanisms (e.g. the tolerance parameters of a homeostatic mechanism) and integrative capacity of the mechanisms (basically, pattern recognition). It may be possible for these three dimensions to give us a coherent account of adaptivity and how it varies. This in turn would open new avenues of research into the relationship between cognition and value, and how that relationship changes through the operation of such adaptive mechanisms. The account proposed differs from the likes of Dennett's "Tower of Generate and Test" (1996, Kinds of Minds, Weidenfeld & Nicolson) and similar models as it is an attempt at an analysis of adaptivity per se, rather than the kinds of mechanism in a given organism that might produce different forms of competence or adaptive response.

The proposed account might therefore also offer ways in which we could codify the concept of mediacy in the interaction between an agent and its environment. In this, the framework might fit with other theorising on the matter such as Hans Jonas's (1966 The Phenomenon of Life, Greenwood Press) arguments concerning the increasing mediacy of the interaction between animals and their environments in evolution, the concept of the "recession of the stimulus" in Edwin Holt's (1915 Some Broader Aspects of Freudian Ethics p.134, Holt Company) description of learning and the variety of forms of cognition identified by Merlin Donald (1991 Origins of the Modern Mind, Harvard University Press) in his account of cognitive evolution.

An artificial chemistry towards the identification of the transition to life

Ivan Alfredo Mendoza, Patricia Muethe, Jonatan Gómez and Marcela Ewert-Sarmiento

National University of Colombia
iamendozap@unal.edu.co

A simplified model of a natural system in the form of an Artificial Chemistry is presented in order to explain the origin of life. It is intended to serve as a tool for modelling certain scenarios where conditions that could lead to the formation of the compounds, thought to have a major role in the emergence of life, are met. The "molecules" (first element of an Artificial Chemistry) in this model take the form of two-dimensional atoms modelled after those most commonly found in living creatures (with a justification for their inclusion). These atoms have specific shapes (and places for bonds) for metals, nonmetals and metalloids. Their last energy level valences, electronegativities, radii (based on the Helium radius along with all other distances in the system) and weights are also included in the model. The rules were designed with two main purposes, to move the atoms around the reactor and to make and destroy bonds. In this way, molecules are formed, moved around and eventually they give form to new molecular species too. Atoms are initially placed at random over a two dimensional space and also given an initial random velocity and acceleration. Their movement is based then in one of two alternatives, since no forces can be modelled to move the atoms in this level: the first approach is to give them velocities that change depending on their relationship (bonding potential) with others in the neighbourhood. The second alternative is to randomly accelerate them but keeping a total constant value for this acceleration, proportional to the atomic weight. For bonding purposes a novel approach is introduced. It is based on a combined strategy that involves the electronegativity difference between two atomic species (when free valences are available) and the "gross affinity", a magnitude given for two specific atomic species that was obtained from counting real bonds from a database of bioorganic molecules. This gross affinity favours the formation of compounds that can be familiar in the context of the origin of life, so the future work for this model is based on the fine-tuning of initial conditions (quantities, placing and environment division) and the modification of the affinity to make the model to move towards specific desired results. Also, the emergence of a prevalence of more complex structures and behaviours could be achieved by tracking down the processes that eventually led to them, boosting these processes with new rules can be a form of mimicking natural selection. The preliminary results are optimistic in terms of the production of basic molecules and substrates and compounds as the ones found in experiments like the ones conducted by Miller and Urey.

Adaptive units of selection can evolve complexes that are provably unevolvable under fixed units of selection

Rob Mills and Richard A. Watson

Science and Engineering of Natural Systems group, School of Electronics and Computer Science,
University of Southampton
rmm05r@ecs.soton.ac.uk

Symbiosis, the collaboration of multiple organisms from different species, is widespread amongst prokaryotes. As symbiotic associations become more absolute, an inseparable union can result (symbiogenesis). This union would typically be considered a new unit of selection once a set of symbionts became reproductively inseparable. However we consider a macroevolutionary model within which new units of selection are formed gradually from maturing symbiotic relationships, without the need for symbiogenic events to be complete. We find that these adaptive units of selection can evolve complexes that are provably unevolvable under fixed units of selection.

We perform experiments in several structured rugged landscapes (including spin glasses) in which there only few global optima amongst many local optima. Importantly, the number of initial conditions that must be sampled to find one of these global optima by individual selection is exponentially large (i.e., their basins of attraction are small).

Selection on individual species only has the ability to evolve to locally optimal configurations. Selecting on groups at the level of entire ecosystems requires an exponentially large number of ecosystems to be created to find optima of globally maximal fitness. Since this level of selection cannot utilise gradient information, global optima are only guaranteed to be found by enumerating all possible ecosystem configurations.

We model an ecosystem, in which species have variable length genotypes. Every species has the potential to develop a symbiotic association with any other species. Initially all species specify a single gene and have no associations. The ecosystem is sub-divided into several semi-independent demes, each evolving to local attractors via individual selection. Symbiotic associations between the entities that are present in the local attractor discovered by each deme are modified to reflect their frequency of co-occurrence in the attractors across all of the demes: when entities are repeatedly found to be successful with the same partners, the symbiotic relationship is strengthened. Strong associations between entities biases the likelihood of their future co-occurrence. These two phases are repeated, resulting in coalitions of increasing size, stability, and co-dependence. Selection starts at the level of the individual, when local optima are discovered. However, the unit of selection adapts as composites gradually develop. Some composites are favoured at the expense of other composites, leading to the discovery of higher fitness optima. These composites consist of co-adapted groups that are locally optimal under individual selection, and not random configurations of large sets of entities. Thus, although the groups can contain many entities, the number of competing complexes is small: the configuration space is reduced as symbiotic relationships are strengthened.

We demonstrate the adaptive significance of these processes by applying them to provably difficult optimisation problems. In general, optimisation problems frequently have some structure that can be exploited, but only with an appropriate mechanism that can recognise that structure. When locally optimal configurations contain some information that is congruent with globally optimal solutions, such as when local optima are created from the frustration of a large number of low-order fitness interactions, the process described above can provide automatic problem decomposition.

When composite entities evolve through the intensification of symbiotic coalitions, the units of selection adapt causing the configuration space dimensionality to be collapsed. We show that this can result in complexes that are unevolvable when the units of selection are static.

Considering the reconfiguration process of subjective temporal order by using recurrent neural networks

Kohei Nakajima and Takashi Ikegami

Department of General Systems Sciences, The Graduate School of Arts and Sciences, University of Tokyo
nakajima@bio.c.u-tokyo.ac.jp

Starting from the famous conception of B. Libet, how the brain orders successive events has been a matter of intense debate in neuroscience. S. Yamamoto and S. Kitazawa (S. Yamamoto et al., 2001, Nature Neuroscience, 4, p.759) revealed that subjective temporal order of successive taps to hands, or to the tips of sticks held in each hand, are easily reversed just by crossing the arms. What is astonishing is that, especially in the crossing case, when the tapping interval was less than 0.3 second, the mistake rate grew to 100%. This result could be taken as general functionality of the real brain's time-space reconfiguration process. In this session, we try to explain these phenomena and to characterize the real brain reconfiguration process of subjective temporal order, from dynamical systems perspectives. To construct the computational model, we adopted the following settings. We prepared an agent containing two arms and corresponding pairs of input nodes, consisting of proprioceptive and exteroceptive inputs, and output nodes to make the agent answer which hand had received the stimuli first. Proprioceptive nodes detect the location of the arms and exteroceptive nodes detect the stimulus applied to the hand. For the agent's internal architectures, we adopted discrete time recurrent neural networks with plasticity and trained the network by using a genetic algorithm depending on the tasks. To prepare the same settings as S. Yamamoto and S. Kitazawa's experiment, one calculation of internal networks was defined as 0.01 second, and one experiment takes 3 minutes by iterating 60 cycles of time period consisting of "a stimulus application interval", "an agent responding interval" and "a resting interval". Each interval was defined as 2 seconds, 2 seconds and 1 second, respectively. The agent receives the stimuli in "a stimulus application interval", has to respond to the stimuli in "an agent responding interval", and rests in "a resting interval". Other actions are not allowed for the agent. Stimuli are designed to create alternative time delay between both hands. For the agent's task, we set two: Task1 involved changing the locations of both arms for the entire parameterized region, and applying the stimuli to only one hand, after which the agent had to answer which hand had received the stimuli. Task2 involved changing the locations of both arms for a limited parameterized region, non-hand-crossing region, and applying the stimuli to both hands, after which the agent had to answer which hand had received the stimuli. For the experiment, we crossed the agent's arms, applied the stimuli to both hands and then analyzed the response and the agent's internal dynamics. Finally, determining the result, we discuss the implications of this model to real brain reconfiguration processes of empirical time and space.

Emergent intracellular network states and cell fate decision: a dynamically integrated study of the epidermal growth factor receptor network with an agent based model

Nuno Nene and Sylvia Nagl

University College London Cancer Institute, Centre for Mathematics and Physics in Life Sciences and Experimental Biology (CoMPLEX)
n.nene@ucl.ac.uk

In biological networks (e.g. protein-protein) component interactions are highly nonlinear. Theoretical models based on ordinary differential equation (ODE) have dominated the simulation of intracellular networks in the literature of theoretical cell biology. This approach is parameter driven and therefore motivated by the data currently available for well studied signalling systems such as the epidermal growth factor receptor (EGFR). Although these models are incredibly detailed they become unmanageable with the increase of network size and hence lose their predictive power. Simplified techniques such as discrete time Boolean networks which strip completely the dynamical system of kinetic parameters and evaluate qualitative network behaviour have been an alternative option for large networks. The network studied in this work involves the extended EGFR signalling pathway and its validated nuclear targets [Oda,K., et al.Mol Syst Biol 1:2005.0010] for which only a small part has been modelled using ODEs.The epidermal growth factor receptor (EGFR) has been implicated in the regulation of cell proliferation, survival and differentiation via activation of signalling pathways. Overexpression or constitutive activation of EGFR has been associated with in-vitro tumourigenic transformation and linked for example to non-small cell lung carcinoma (NSCLC), breast and colon cancers. Small molecule kinase inhibitors of the EGFR have been developed and two of them, Gefitinib and Erolitinib, have already been licensed for clinical use in NSCLC. These drugs have been found to have positive impact — reduction in cell proliferation and induction of apoptosis — in patients with mutated EGFR. Nevertheless, the action of the inhibitors has many non-specific interactions [Fabian, M, et al., (2005), Nat. Biotechnol, 23, 329]. Both pro-cancerous and anti-cancerous kinases, in addition to EGFR, are inhibited exerting therefore additional effects on the cell. To study the possible consequences of an EGFR signalling network perturbation and devise strategic methods for cell fate decision control, through the administration of drugs, an Agent Based Modelling (ABM) approach was used. Cell phenotype is identified with particular network emergent dynamical states. An extension of a class of continuous time Boolean networks was analysed under the ABM paradigm. Each agent, seen as a control gate, represents a 'molecular species' (signalling protein or transcription factor) with a normalized concentration value but with only ON/OFF output states determined by a threshold value. Its concentration evolution is represented by a piece-wise linear differential equation with time delays. The time delays stand for diffusion associated effects or other processes that escape the usual assumption of a well mixed system in cell simulations. Higher autonomy of each node is secured by the existence of an internal noise term associated with each time delay. Agents representing transcription factor nodes exhibit higher time delays which stand for the differences in time scale of signal transduction and gene transcription processes. Node targeting is performed by node removal, by modulation of the internal noise term or partial inhibition according to the binding assay data [Fabian et al (2005)]. This modelling approach is the simplest possible one retaining important properties of biological systems such as distributed control, asynchrony and noise.

Is the organism really a machine?

Daniel Nicholson

University of Exeter
dan.j.nicholson@gmail.com

Despite all its successes, modern biological science has done remarkably little to tackle the fundamental question that lies at the very heart of biology: what is the nature of the living organism? Contemporary biologists (and philosophers of biology, for that matter) seldom ask this question openly and explicitly. The reason is simple: they already presuppose the answer. The organism is a machine.

One could conceivably construe the history of biology since the seventeenth century as the story of the success of the Cartesian notion of the bte-machine. Although the dissatisfaction with this mechanistic conception is almost as old as the idea itself, most of those who found themselves in disagreement were labelled as vitalists and marginalised from the scientific discussion. Today, the organism-machine analogy is dominant in virtually every branch of biological science that studies the organism. This conception of life promotes the view that biology is a subsidiary branch of physical science within which the theories and methods of physics, chemistry, and engineering can be fruitfully applied. In this way, the organism-machine analogy serves to not only justify, but also actively encourage, a number of epistemic attitudes, such as strict methodological reductionism, that have dominated the study of life since the mid-twentieth century.

In this paper I question the conceptual coherence of the organism-machine analogy, and in so doing, I challenge some of the central presuppositions underlying contemporary biological research. Apart from promoting a misleading view of what the living organism is and how it behaves, I argue that the Cartesian notion of the bte-machine has actually more in common with a Creationist/ID theorist's conception of life than it does with a well-informed evolutionary-developmental account of the organism. This unhappy marriage of Cartesian mechanicism on the one hand, and Neo-Darwinism on the other, has led to a number of tensions which play out both at the theoretical and practical level concerning what the organism is and how it should be studied.

As a symptom of the pervasiveness of the organism-machine analogy in biological thinking, several research programmes have emerged in recent years which aspire to provide the ultimate vindication of the organism-machine analogy. Central among them is Synthetic Biology, although a number of lines of research in A-Life also appear to share this objective. Whilst not denying the legitimacy and usefulness of these new fields, I argue that the faith that has been bestowed upon these disciplines regarding their potential to substantially advance our biological understanding of life is likely to be misplaced.

Finally, I draw from the long-standing anti-mechanistic tradition in biology to propose an alternative, organisation-based conception of the organism that sidesteps the problems of the organism-machine analogy, eliminates some of the deep-rooted conceptual tensions generated by it, and provides an understanding of the organism that is more in accordance with its actual nature.

Intelligent locomotion of eukaryotic cells

Shin Nishimura and Masaki Sasai

Nagoya University
shin@tbp.cse.nagoya-u.ac.jp

Eukaryotic cells such as cellular slime molds (Dictyostelium descoideum) and other animal cells are thought to share unified mechanisms for locomotion. In this work, emergence of "intelligent" behaviors of cells is discussed by developing a simple computational model of locomotion. The model describes changes in cell shape on the two-dimensional plane by considering a cell membrane, actin filaments embedded in the membrane, and an intracellular control factor called "cortical factor". Actin filaments polymerized on the membrane push it outward to change the cell shape, whereas cortical factor suppresses polymerization of actin filaments. Cortical factor is conveyed from the leading edge to the rear of a moving cell by the intracellular flow of cortex and accumulates at the rear of the cell. This flow of cortical factor leads to the spontaneous locomotion of cells by amplifying the initial fluctuation in cell movement: If a fluctuating cell slightly moves into a direction, cortical factor begins to accumulate at the rear of the moving cell, which suppresses actin polymerization there and further promotes cellular locomotion in the initial direction. This positive feedback mechanism reproduces the experimentally observed amoeboid-like and keratocyte-like locomotion and cytokinesis B- and C-like cytofission depending on the kinetic rate and the threshold value for actin polymerization in the model, where the amoeboid-like locomotion is a repeat of stop-and-go motion, and the cell usually changes its moving direction after the stopping phase, while the keratocyte-like locomotion maintains a moving direction for long duration. Cytokinesis B-like cytofission divides a cell into two parts, and a cell is torn into several pieces in cytokinesis C-like cytofission. Based on this model of eukaryotic cells, emergence of intelligent behaviors in locomotion is demonstrated. We assume that the reception of external chemical signal suppresses the activity of cortical factor, leading to chemotaxis of a cell toward the source of the chemical signal. We consider that there exist obstacles intercepting a cell on its way to the source of signal. When signal permeates through some obstacles to attract the cell (i.e. traps), the simulated cell falls into a trap at first but it suddenly escapes from the trap to find a way to get it around. The cell finds this detour because the distribution pattern of cortical factor is flushed while the cell is trapped and the occasional fluctuation in cortical factor amplifies locomotion against the gradient of the external signal. In this way, the feedback loop between cell movement and cortical factor is a key mechanism of the emergent behavior to find a detour. We also discuss efficient food finding and cells' hunting moving bacteria. Cognitive locomotion of the model eukaryotic cells is the result of the fluctuating dynamics in which the interaction with the environment and the internal chemical reactions are coupled through the feedback between cell movement and cortical factor.

What can artificial life offer ecology?

Jason Noble[1], Donna Clarke[2] and Rob Mills[1]

[1]Science and Engineering of Natural Systems group, School of Electronics and Computer Science, University of Southampton
[2]School of Biological Sciences, University of Southampton
jn2@ecs.soton.ac.uk

Artificial life is the simulation and synthesis of living systems, and ALife models show how interactions between simple entities give rise to complex effects. Ecology is the study of the distribution and abundance of organisms, and ecological modelling involves fitting a linear model to a large data set and using that model to identify key causal factors at work in a complex ecosystem. We are interested in whether the individual-based modelling approach of ALife can be usefully employed in ecology.

ALife models are "opaque thought experiments" (Di Paolo et al., 2000, Proc. ALife VII, p.497). They show that a phenomenon can arise from a given set of assumptions in cases where the implication is not clear from intuition alone: e.g., that spatial structure in a population can lead to altruistic behaviour. This type of modelling can be useful to ecology by showing the plausibility of a novel concept or process, which in turn suggests new natural experiments and new forms of data to collect. However, we argue that ALife models can go beyond this "proof of concept" role and serve as a direct account of data in the same way that statistical models do.

We focus on a typical problem from ecology: the effect of clearing powerline corridors through a forest on the local wildlife populations (Clarke et al., 2006, Wildlife Research, 33, p.615). The real data set in this case is complex and, of course, we don't know the true effects that underlie it. We therefore generated a fictional data set that reflects aspects of the original problem while allowing complete control over the simulated environment. The idea is to construct a test case for looking at the relative success of different modelling approaches. We know the true picture because we generated the data, but which modelling approach will get closer to the truth? The fitting of generalized linear models as is conventional in ecology, or the use of individual-based simulations as in ALife?

Statistical models are fitted using some variant of the method of maximum likelihood: given the data, which of the models in the family we're considering (e.g., a linear regression) makes the observed data most plausible? When dealing with simulations, however, it is difficult to establish that one model is a better fit to data than another. Simulations have many parameters, it may be difficult to determine a level of granularity at which the simulation output is supposed to "match" the data, and there will be no analytically tractable likelihood function. These problems are solved by the method of indirect inference (Gouriéroux et al., 1993, J. Applied Econometrics, 8, p.S85) in which an auxiliary model is fitted to both the real data and to the output from competing simulation models. The best simulation model is the one producing the closest match to the data in terms of fitted parameter values in the auxiliary model.

Using indirect inference with our fictional data set we demonstrate that ALife simulation models can be fitted to realistic ecological data, that they can out-compete standard statistical approaches, and that they can thus be used in ecology for more than just conceptual exploration.

Information-theoretic characterization of relative and fluctuating system–environment distinction

Takayuki Nozawa and Toshiyuki Kondo

Tokyo University of Agriculture and Technology
tknozawa@cc.tuat.ac.jp

Defining a system in distinction from its environment is a fundamental but elusive problem in artificial life as well as in real-world complex systems. While many notions of closure gives a qualitative and absolute criteria for the system–environment distinction, the concept of "informational closure" proposed by Bertschinger et al. (Bertschinger et al., 2006, Proc. GWAL-7, p.9, IOS Press) gives gradual and relative evaluation of closure (or closedness). There, a system is tentatively defined in distinction to its environment, and the validity of the definition is judged according to how causally closed the system is, being quantified by information flow (transfer entropy) from the environment into the system. This quantitative approach for the characterization of closedness is expected to bring rich description of "relative" systems on a wide range of dynamical models.

In this study we proceed one step further in the direction of relativizing closure: for the evaluation of closedness we also utilize information-theoretic measures, such as the transfer entropy and difference of Boltzmann-type and KS-type entropies, but instead of evaluating closedness of a system with its elements fixed in time, we evaluate the closedness for the system's specific states which are dissociated from the history of interaction with the environment. This dissociation excludes from the system–environment correlation the components which are realized by the system modeling or controlling the environment. Therefore, the measures evaluate solely how a state can prevent the invasion of uncertainty from the environment. This setting can be effective in describing partial closures which appear transiently and fluctuate in uniformly structured degrees of freedom or in a directed flow of information processing, while the original setting of informational closure would be more efficient when the meanings of a system and its environment are clear, their boundary is fixed, and advanced notions such as cognition, learning, self-reference, etc. are of immediate interest.

We apply the method to discrete dynamical networks and a cellular automata model which simulates physico-chemical self-organization (molecular aggregation). The spectrum of closedness is shown to depend on the dynamical properties of each model. (The investigation of the spectra has some similarity with the exploration of characteristic structures in the phase spaces of chaotic systems.) We will also discuss how reversibility of the models and introducing dissipative irreversibility, that is, disregarding information flow into the environment as a heat-bath, can influence the evaluation of the closedness.

Personalities in fish without genetic differences: a model

Johanneke Oosten and Charlotte Hemelrijk

Centre for Ecological and Evolutonary studies, University of Groningen
j.e.oosten@rug.nl

Differences in personality between individuals are usually attributed to genetic differences, and seldom to differences in experience. In this study, we investigated to what degree personalities may result from differences in experience, that are caused by self-organisation of behaviour and chance. We use an individual-based model to generate such an explanation for the experimental findings on personalities in perch (Magnhagen & Staffan., 2005, Behav Ecol Sociobiol, 57, p.295). In this study, small groups of individuals could either hide in vegetation, or visit an open area that contained food, but was near a predator. Individuals were attributed a personality, based on the time that they spend in the open area, and on how fast they fed there.

In our model, we mirror this experiment, with artificial individuals that are genetically identical. We test whether personalities arise as a consequence of three mechanisms: habituation, social facilitation, and competition. To this end, we study three models: a model of only habituation, a second one of habituation and social facilitation, and a third that includes all three mechanisms.

The first model focuses on habituation. Artificial individuals habituate by increasing their (initially low) tendency to enter the open area after each successful foraging event. Although this self-reinforcing effect led to personality differences, these differences disappeared as soon as all individuals were habituated.

In the second model, we added social facilitation. This implied that individuals tended to visit the open area more, if it was already occupied by group members. Social facilitation led to a to a positive correlation between personality of an individual and that of its group members. Because of this, groups more often arose, that consisted of a single personality type. Both observations resemble the empirical data. However, in this model, personality differences also disappeared because all individuals habituated.

In the last model, we added competition. This was represented by a 'residence effect': upon arrival in the open area individuals captured less prey, if group members had arrived before them. Here, personality differences appeared to be stable over long periods of time.

In sum, our models show that differences in speed of habituation may give rise to personality differences, and that these differences are reduced by social facilitation, and maintained by competition. We conclude that it is valuable to consider learning and social interactions as an explanation for the origination of personality in animals.

that can reside in either a protected area or an open area with food. However, all individuals are genetically identical, and start with a low tendency to reside in the open area. Similar to the empirical experiment, they can reside in either a protected area, or an open area with food.

What can artificial life offer the development of methodologies in the field of socio-ecological sustainability?

Alexandra Penn

Science and Engineering of Natural Systems group, School of Electronics and Computer Science,
University of Southampton
asp@ecs.soton.ac.uk

Issues of sustainability involve the dynamics and interactions of multiple complex adaptive systems at a variety of scales: climatic; ecological; economic; technological; political and social. Some of the most pressing challenges for society are inherently concerned with gaining a better ability to understand and manage the interacting systems upon which we rely. While there is widespread acknowledgement that the science of complex adaptive systems can provide key tools to address these challenges, there is little consensus on how to develop and apply these tools effectively. New approaches are therefore needed before effective policy-making can be informed by well-founded scientific modelling. Concepts and terminology such as complexity, complex adaptive systems, whole systems thinking, non-linear dynamics, co-evolution, autopoeisis, and self-organisation enjoy common currency in movements such as resilience thinking, sustainable systems approaches and permaculture (amongst others), which share a common ancestry with A-Life in systems theory. The use of such ideas as metaphors to guide thinking is valuable up to a point. For example, the concept of non-linear response to change, including at the extreme system tipping points, is an important understanding which must certainly guide policy in areas such as climate change. A large array of complex systems metaphors are also used as sources for design and management heuristics. However, there are presently enormous methodological leaps to be made before their full potential usefulness can be realised, and the availability of clear quantitative or qualitative measures and methodologies connecting theory with practice is extremely limited. A clear opportunity exists for the field of Artificial Life to contribute in this domain at this key time. In this talk I will give an overview of the current use of complex and dynamical systems concepts within the sustainability movement and associated challenges. I will detail practical tools being developed to measure the qualitative or quantitative behaviour (or health) of dynamical systems such as ecosystems, and discuss how we can move from a metaphorical understanding of such systems as complex, dynamical or adaptive, towards strategic intervention in or interaction with them with the goal of sustainability in mind.

I will focus on what I consider to be three key areas in which A-Life methodology can contribute: 1) The use of modelling to predict the gross behaviour of systems, with a particular emphasis on the incorporation of evolutionary processes, network dynamics, and agent-based modelling into current resilience approaches; 2) The development of quantitative indicators of systems' "health" with regard to their ability to self-maintain; 3) The development of tools for management or steering of complex systems undergoing rapid change, including the potential for "engineering" or "programming" self-organisation of complex adaptive systems for increased resilience, robustness and "sustainability".

The aim of this talk is to initiate dialogue between theoreticians and practitioners towards practical use of A-Life methodologies in frontline sustainability.

Mechanisms for the initiation of multicellularity in bacterial biofilms

Alexandra Penn[1], Richard Watson[1], Simon Powers[1], Jeremy Webb[2], Alex Kraaijeveld[2], Tim Conibear[2] and Zoe Bigg[2]

[1]Science and Engineering of Natural Systems group, School of Electronics and Computer Science, University of Southampton
[2]School of Biological Sciences, University of Southampton
asp@ecs.soton.ac.uk

Major transitions in evolution create the prerequisite features which allow natural selection to occur at a new level of organisation. Heredity, variation and reproduction must all be produced at a new higher level in order for a new evolutionary unit to arise. The origin of these features at new levels of organisation has been problematic for both theoretical biology and in artificial life models — but several partial theories exist. The necessary features may arise under the action of adaptive processes on existing units, and/or with the potential support of self-organisation of some kind. Limited mechanisms of heredity, potentially including ecological inheritance of constructed niches, may play an important role in bootstrapping the early stages of a transition to higher-level selection on new units. But, in short, theories for the precise routes by which new biological individuals might arise remain mostly speculative.

Our programme of theoretical modelling work has been focussing on simple individually-adaptable characters that may be involved in initiating higher-level units of selection. It is known that two key determinants of the efficacy of higher-level selection, between-group variation and heredity of group characters (e.g. the species composition of a group), are significantly affected by the modification of some simple parameters such as initial group size in an aggregation and/or the size of dispersal propagules. Our models have demonstrated that if simple features such as group structure parameters, group size and dispersal modes can be affected by characters that are under individual adaptive control then conditions that effect significant higher-level selection can be selected for despite individual self-interests. These features are simple enough that we can begin an empirical programme to investigate and manipulate the relevant variables.

In this talk we describe an ongoing experimental programme to investigate parameters affecting the levels of selection using bacterial biofilms. The majority of bacteria spend most of their life cycle in single- or multi-species biofilms, complex collective structures formed when bacteria attach to surfaces, and in this form they display an extraordinary repertoire of coordinated behaviours and interactions. Bacterial biofilms have numerous high-impact application areas including bio-engineering, bio-remediation and medicine where controlling the adaptation and co-adaptation of bacteria is vital. Reproduction in biofilms may be either via shearing off of groups of cells or by the production of individual motile cells. Despite this, such groups are also able to disaggregate into individual cells which reproduce in a planktonic phase. Accordingly, these organisms, in this case Psuedamonous aeruginosa (a common opportunistic pathogen problematic in cystic fibrosis), provide an excellent model system to address questions concerning the transition to multicellularity. They are fast-growing and experimentally tractable, allowing us to perform multi-generational evolutionary experiments over a relatively short timeframe, and they naturally exhibit physical characteristics, such as microcolony formation, that implement group structure. They possess individual characters, such as siderophore production, that can be knocked-out to produce clearly identifiable 'cheats' and 'cooperators'. Crucially, there are easily identifiable individual characters, such as extra-cellular matrix production, that clearly affect the grouping parameters that are of interest, such as group size and propagule dispersal. We present our experimental methodology for manipulating these characters and thereby parameters that affect the strength of group selection; a vital first step in tackling the investigation of evolutionary transitions in real organisms.

Quantifying creative symbiosis: a lexical analysis of the evolution of technology as reflected in patent records

Noah Pepper, Charles Francis, Mark Bedau, Devin Chalmers and Bryan Head

Reed College
peppern@reed.edu

The similarities and differences between adaptive dynamics in biological and cultural evolution is an important and controversial open question about evolutionary processes in the real world. One way to address this issue is by studying empirical data from biological and cultural evolution. Technology is itself an important part of culture, and one that is rather amenable to empirical investigation, and an excellent window on the evolution of technology is to study patters in the citations among patent records (Jaffee & Trajtenberg, 2005, Patents, Citations, and Innovations, MIT Press). Each patent must describe a novel concept and must cite previous related works and each patent is filed at a distinct point in time. This provides us with a time series corpus of formally written text that records all patented technological innovations, an ideal platform for studying the creativity of one form of cultural evolution.

Bedau and Skusa (2002, ALife VIII, p. 431, MIT Press) analyzed the dynamics of adaption in the evolution of technology, by looking at patent citations and measuring evolutionary activity statistics over time (Bedau & Packard, 1991, ALife II, p. 431, Addison-Wesley; Bedau et al., 1998, ALife VI, p. 233, MIT Press). We here present a novel, complementary method of analyzing technological evolution using the textual content of patent records. Our analysis takes linguistic tokens as the unit of cultural adaptation and measures their occurrence and relations using several linguistic tools. This can reveal latent connections between conceptual or technological innovations.

We have analyzed a corpus containing thirty years of patents (over 4 million) with the WORDSPACE model for quantification of word relatedness within a corpus (Widdows, 2004, Geometry and Meaning, CSLI). This produces a fuzzy set of correlated terms based on concurrence within a text. The change in relatedness of n-grams over time provides a movie of a part of cultural evolution. We also analyze regressions on time series frequency counts of n-grams and groups of n-grams. (N-grams are linguistic tokens n words in length which represent technologies or concepts in the corpus.) Patterns in n-gram frequencies provide another window into the evolution of culture.

These two tools reveal and map the cross-temporal relationships between technologies. One can see classes of technologies share in significant evolutionary success, and then eventually decline (for example, the recent technology bubble). The tools can reveal when different technologies have symbiotic relationships (Marguilis & Fester, 1991, Symbiosis as a Source of Evolutionary Innovation). These results help illuminate the nature of cultural change and open-evolution, and in particular, whether there is a fundamental difference in the evolution of cultural and biological systems.

The group selection debate and ALife: weak altruism, strong altruism, and inclusive fitness

Simon Powers and Richard Watson

Science and Engineering of Natural Systems group, School of Electronics and Computer Science,
University of Southampton
stp05r@ecs.soton.ac.uk

Models of the evolution of social behaviour are often framed in terms of either multi-level selection or inclusive individual fitness theory. Although both of these descriptions correctly predict changes in gene frequency (where group fitness is defined as the average individual fitness of the group members), it is still a hotly contested issue as to which provides a faithful description of the underlying causal processes at work. Furthermore, the type of model analysis used reflects the philosophical bias of the author. It is important for Alife researchers to be aware of this issue when evaluating or presenting models of social evolution, for many authors simply claim as a matter of fact that their model works via multi-level or (inclusive) individual selection, without acknowledging the alternative perspective.

In this talk, two particular areas of ongoing contention between multi-level and individual selectionists will be illustrated, using examples from the Alife literature. The first of these concerns the evolution of weakly altruistic traits. These are behaviours that provide a whole-group benefit at some cost to the actor. Crucially, however, the cost to the actor is more than offset by its share of the group benefit, such that the lifetime number of offspring of the actor is increased. In a recent paper West et al. (2007, J. Evol. Biol., 20, p.415) have advocated that the evolution of such traits can be adequately explained in terms of direct fitness benefit, thus avoiding the need to invoke selection at the group level. However, this explanation hides the fact that weak altruists suffer a relative fitness disadvantage within every group. Indeed, the local attractor within any one group is the extinction of weak altruists. Therefore, the behaviour cannot spread unless groups compete and groups with more weak altruists are fitter than those with less. While the individualist methodology correctly predicts if the behaviour will evolve, it obscures the mechanistic explanation. This suggests that models couching the evolution of social behaviour in terms of individual benefit should be analysed to determine whether group structure is playing any causal role in the evolutionary dynamics.

The second issue to be addressed by this talk concerns the evolution of strong altruism, i.e., behaviours where there is a reduction in the lifetime number of offspring of the actor. For such behaviours to evolve there must be a correlation in interactions, such that the recipients of an altruist's help tend to be altruists themselves. This correlation frequently occurs in nature through the limited dispersal of kin, and is usually modelled by inclusive fitness equations that contain no notion of group fitness. However, the underlying mechanism is that kin groups with more altruists outcompete those with less. Once this is realised, it becomes apparent that other assortative group formation mechanisms can in principle produce the same effect. Appealing to kinship is therefore simply invoking one kind of assortative grouping.

This talk will further elaborate on these points, including definitions of a group, and consider claims about the strength of group selection.

A strong local dependency of developmental bias orients adaptive evolution in a model evo-devo system

Sean Psujek[1] and Randall Beer[2]

[1]Case Western Reserve University
[2]Indiana University
sean.psujek@gmail.com

The differential production of phenotypes, that is, developmental bias, has been presumed to have a strong influence on the direction of evolutionary modification. In a computational evo-devo system generating neural connectivity, we demonstrate an orienting role for developmental bias during adaptive evolution. The differences in phenotypic transitions taken during evolution are, to a large degree, due to phenotypic accessibility being strongly dependent on the interactions of the developmental process as directed by a particular genotype. We define developmental bias as the differential production of phenotypes given uniform genetic variation. In our gene-based developmental system, we approximate the range of phenotypic variation possible by creating random genotypes and determining the phenotypes they produce. The resultant pattern of phenotypic variation indicates an intrinsic bias (global bias) in the developmental system. We then determine the accessibility of phenotypes from a given genotype (local bias) by determining the phenotypes generated through each and every single-base substitution. We find that the local bias patterns vary strongly with the genotype, even among phenotypically-neutral genotypes. These patterns also differ from the global bias pattern indicating local biases depend more on the dynamics of the developmental process than on the overall mechanisms of the developmental system.

During evolutionary simulations toward a target phenotype, the local dependency of bias dictates the phenotypic transformations that occur. For example, in two simulations at the generation preceding an increase in the populations' best fitness (the populations have approximately the same average fitness), the target phenotype is produced in one population but not in the other. The average of the local bias patterns for all individuals in the second population (population bias) shows the target phenotype is completely inaccessible through mutagenesis of the population. Other fitness-increasing phenotypic transitions show a similar result; the particular phenotype produced is dictated by the phenotypic variants accessible from the population. This results in multiple phenotypic pathways to the target phenotype across simulations.

Because local bias has such a strong dependency on the dynamics of the developmental process as determined by the regulatory structure of the genotype, bias patterns often change dramatically during evolution through the accumulation of mutations (neutral or otherwise). Phenotypic variants that are possible, as indicated by local bias patterns, occasionally are not able to be generated in subsequent generations. More importantly, phenotypes previously inaccessible often become available after multiple rounds of mutation. Mutations change the developmental context in which subsequent mutations operate. In one example, a mutation previously selectively-neutral eventually becomes a beneficial mutation, resulting in a change to a higher-fitness phenotype.

These results indicate that developmental bias has a strong influence on the direction of evolutionary modification. More generally, there are features of the genotype-to-phenotype and phenotype-to-fitness mappings that affect evolvability, the capacity to vary in phenotypic availability over time.

Evolutionary robotics models in the interdisciplinary study of embodied time perception

Marieke Rohde and Ezequiel Di Paolo

Centre for Computational Neuroscience and Robotics, University of Sussex
m.rohde@sussex.ac.uk

A recent study by Cunningham et al. (Cunningham et al., 2001, Psychological Science, 12, p. 532) has shown that human subjects adapt to delayed visual feedback in a visuomotor task both behaviourally and experientially, i.e., the behaviour is altered in such a way that successful performance on the task relies on the presence of a visual delay (negative after-effect) and that the experience of simultaneity is re-adjusted to incorporate the visual delay. This adaptation effect is similar to those observed in experiments with visual displacements, but contrasts with earlier experiments with sensory delays, in which no such adaptation occurred (e.g., Smith and Smith, 1962, Perception and Motion, Saunders).

This discrepancy (i.e., adaptation in some situations but not in others) suggests that adaptation to sensory delays does not proceed automatically, on the basis of statistical properties of sensory inputs, but is contingent on the performed behaviour and the associated sensorimotor dynamics. Artificial Life and Evolutionary Robotics simulation models are proven tools in the study of non-linear sensorimotor dynamics, which are difficult to understand intuitively. In particular, our earlier work (Di Paolo et al., 2008, New Ideas in Psychology, forthcoming; Rohde, 2008, PhD Thesis, University of Sussex; Rohde and Di Paolo, 2007, ECAL 2007, p. 193, Springer) argues and demonstrates how Evolutionary Robotics simulation models can contribute to the scientific study of human sensorimotor adaptation.

In a combined experimental and evolutionary robotics modelling study, we have tested the (unconfirmed) hypothesis put forward by Cunningham et al. (Cunningham et al., 2001, Psychological Science, 12, p. 532) that adaptation to sensory delays occurs if there is time-pressure on the task (Rohde, 2008, PhD Thesis, University of Sussex; Rohde and Di Paolo, 2007, ECAL 2007, p. 193, Springer). On the basis of data analysis of both the artificial model agents' and the experimental subjects' sensorimotor recordings we revised our hypothesis: We now believe that, apart from time pressure, the task needs to feature a systematic link between present motion and future sensation over a longer time span in order to make the task predictable.

This new hypothesis will be tested using a combined Evolutionary Robotics modelling and experimental psychophysics approach proposed and applied in (Rohde, 2008, PhD Thesis, University of Sussex) that aims at formalising and explaining the sensorimotor invariances associated with perceptual experience of time and simultaneity. We argue that the contingent relation between function and underlying mechanisms inherent in Evolutionary Robotics simulations helps to identify general dynamical principles and fundamental sensorimotor invariances across viable solutions. In this aspect, the approach taken is more general and less biased, even though also less transparent than related approaches like robotic forward model learning (e.g., Tani, 1996, IEEE Trans. SMC (B), 26, p. 421). This novel methodological framework, which is characterised by a close match between simulation model and minimalist empirical experiment, can be applied to other problems of perceptual experience and opens up new powerful avenues for interdisciplinary research that uses Artificial Life methods to study of human perception and cognition in the closed sensorimotor loop.

Selective attention in artificial organisms

Fabio Ruini[1] and Domenico Parisi[2]

[1]University of Plymouth, School of Computing, Communications and Electronics
[2]Institute of Cognitive Sciences and Technologies, National Research Council, Rome
fabio.ruini@plymouth.ac.uk

Behaviour has two levels, which we will call strategic and tactical. At the strategic level the organism has to choose which particular activity to pursue at any given time (e.g., looking for food, finding a sexual partner, escaping from dangers, sleeping). At the tactical level it must implement the particular sequence of actions that makes it possible to achieve the goal of the chosen activity. The strategic level has important consequences for attention. Since an organism receives many different stimuli at the same time, it has to selectively attend to the ones that are relevant to the current activity while ignoring those which are not.

A population of organisms lives in an environment with randomly distributed food elements in which a predator appears from time to time. The organism's behaviour is controlled by a neural network with input units encoding the location of the nearest food and other input units encoding the location of the predator when it is present. Both sets of input units are connected to a single layer of internal units, which in turn are connected to the output units that control the organism's movements. The connection weights of the neural architecture are evolved using a genetic algorithm where the organism's fitness depends on both the number of food elements eaten and the organism's ability to avoid being reached by the predator. We contrast two populations of organisms, one with the basic neural architecture and the other one with an architecture which includes an additional set of units which receive connections from the input units encoding the predator's location and send connections to the internal layer. Both populations are able to evolve the appropriate behaviour which consists in looking for food when the predator is absent and flying away from the predator when it is present, ignoring food. However, the population with the additional units reaches higher levels of fitness compared with the population with the simpler architecture. Two additional control simulations show that higher levels of fitness are not obtained if we simply increase the number of internal units or if we connect the additional units directly to the output layer.

To better understand why the additional units yield a better performance we compared the activation patterns of the internal units when the locations of both food and predator are encoded in the input units and when only the latter is encoded and there is no food. This comparison shows that the contribution of the additional units consists in making the activation patterns more similar in the two conditions, with the activation patterns becoming even more similar as the predator comes closer to the organism. In other words, the additional units allow the organism's nervous system to better filter out the information from food when the predator is present and, therefore, they might be considered as functionally equivalent to the modulatory influence of subcortical structures on frontal cortex in real organisms.

Organizational requirements for 'open-ended evolution'

Kepa Ruiz-Mirazo, Jon Umerez and Alvaro Moreno

Dept. Logic and Philosophy of Science, University of the Basque Country, Spain
kepa.ruiz-mirazo@ehu.es

In this contribution we will review different conceptions of open-ended evolution and propose our own (Ruiz-Mirazo et al., 2008, Biol. & Philos., 23, p.67). Then, we will consider what are the general conditions that would allow such an evolutionary process to take place, with a specific focus on the type of organization that the systems involved should have. It will be argued that a strong 'dynamic decoupling' is necessary, making possible the long-term maintenance of those systems, in which the individual (self-constructing) and collective (ecological and historical) spheres become deeply intertwined. Particular attention in the discussion will be given to bottleneck cases, like a hypothetical prebiotic 'RNA-world', which –according to our account– would not meet all the requirements. We will also reason why the evolution of a prokaryote world is already open-ended, even if the transition to higher levels of complexity (eukaryotes, multicellular organisms, cognitive agents,) would imply further organizational bottlenecks and the fulfilment of additional conditions.

Self-organizing heterogeneous swarms designed through evolutionary methods

Hiroki Sayama

Department of Bioengineering, Binghamton University, State University of New York
sayama@binghamton.edu

Self-propelled particle swarm models are computational models of many particles capable of autonomous acceleration and local kinetic interaction. Their dynamics have been extensively studied in physics, theoretical biology, and computational science communities because of their useful implications for the understanding of collective behavior of various autonomous agents (e.g., bacteria, fish, birds, pedestrians) as well as their potential of application to practical problem solving.

Earlier studies mostly focused on homogeneous swarms, assuming that the same (or quantitatively similar) set of kinetic rules uniformly apply to all the particles. Some literature also assumed intra-specific variations among particles (such as in body size or velocity) but none of them systematically considered interactions between kinetically distinct types of particles. In real biological/ecological systems, however, there are cases where multiple distinct types of organisms interact to form nontrivial patterns. In a herd of animals, for example, males and females, or parents and offspring, occupy different ecological positions and therefore adopt distinct behavioral rules. A unique formation may arise within the herd from interactions between those different types of organisms. Such self-organization of heterogeneous swarms could also be useful for engineering design purposes.

We therefore extend our scope to heterogeneous self-propelled particle swarm systems in which more than one type of particles can co-exist and interact with each other in the same space. Our model, "Swarm Chemistry" (Sayama, ECAL 2007, p.675, Springer), assumes self-propelled particles moving in a two-dimensional infinite continuous space. Each particle can perceive only the local center of mass and the average velocity vector of other particles within its local perception range, and change its velocity in discrete time steps according to kinetic rules similar to those of Reynolds' Boids (Reynolds, 1987, Computer Graphics, 21(4), p.25). Each particle is assigned with its own kinetic parameter settings that specify preferred speed, local perception range, and strength of each kinetic rule. Particles that share the same set of kinetic parameter settings are considered of the same type.

Using this model, we computationally studied what kind of patterns/motions could emerge out of the mixtures of multiple types of particles. In the first experiments testing the effects of two-type interactions, we found that heterogeneous particle swarms usually undergo spontaneous mutual segregation, often leading to the formation of multilayer structures. Driven by their own endogenous self-propulsion forces, the aggregates of particles may additionally show more dynamic macroscopic behaviors, including oscillation, rotation, and linear or even chaotic motion. Moreover, to explore the possibilities of more than two type interactions, we developed an interactive simulation tool with which a human experimenter can select, perturb, mix, and mutate heterogeneous swarms using an interactive evolutionary method. The second experiments using this interactive tool and human participants further revealed unexpected possibilities of more complex, mechanical, and/or even biological-looking structures and behaviors when several different types are mixed appropriately. Specifications of those patterns were indirectly and implicitly woven into a list of different kinetic parameter settings and their proportions, which would be hard to obtain through conventional design methods but can be obtained heuristically through evolutionary design methods. These results suggest a novel direction of understanding and engineering collective behavior of physical agents, such as distributed robotic systems.

The interactive simulation tool is implemented in Java and available at http://bingweb.binghamton.edu/ sayama/SwarmChemistry/ . Readers are invited to participate in the ongoing exploratory efforts of this project.

Redefining robustness

Eric Silverman and Takashi Ikegami

University of Tokyo
erics@sacral.c.u-tokyo.ac.jp

The concept of robustness in the context of most work in Alife and complex systems implies that the results of a given model remain consistent despite unexpected variation. For example, homeostatic coupling between an animat and the environment is one possible simple form of robustness which we have demonstrated in simple simulation models (Ikegami et al., 2008, BioSystems, 91, p. 388), in which we defined robustness as a dynamic that sustains variability. However, these models still were situated in simple virtual environments. We are currently using this simulation as a basis for developing a real-world robot experiment with "virtual" sound scape setups. Developing this link between real and simulated methodologies has led us to an examination of robustness in a broader sense.

We argue that robustness as commonly defined in Alife is no longer adequate for producing real insight into the functions of biological life. Robustness in one methodology or virtual world does not imply robustness in another, and likewise does not imply that we can develop a robust explanation of the behaviour of interest.

Robustness analysis as a concept is credited to Richard Levins who was the first to truly address pragmatic concerns important to biological modellers (1966, Conceptual Issues in Evolutionary Biology, p. 18, MIT Press). Levins argued that the construction of robust theorems from models involves studying similar but conceptually different models of the same phenomena and attempting to discern the common structures between them. Levins' pragmatic concerns about modelling illuminate similar tradeoffs made by modellers in the Alife community, leading some models to become mired in modelling for its own sake, creating simulations with little relation to the natural world (Silverman et al., 2008, forthcoming).

With this perspective in mind, a reevaluation of the concept of robustness within Alife is needed. While Alife can contribute to the search for common structures in biological systems which can drive behaviour, producing robust theorems about those behaviours also involves confirming that such structures are instantiated in the system of interest (Weisberg, 2005, Phil. Sci., 73, p. 730). A unified framework under which to search for common structures is central to these concerns. Without a clear common relationship between conceptually related models, performing Levinsian robustness analysis becomes an impossible task.

Thus, we argue that finding robust theorems in Alife which demonstrate common structures is made difficult by the lack of common environments between models. A more critical analysis of what constitutes a useful environment for simulation and robotics is needed, and without such analysis, our concept of robustness falls short of Levins' requirements for developing true robust theorems about the natural world. In essence, crafting a robust explanation of a behaviour using a model requires a robust demonstration of that behaviour through a suitable combination of modeling and experimentation. We contend that combining simulation and robotics with an approach using common methodologies and related environments as described above will allow us to develop a new definition of robustness in Alife.

Dancing robots: form, environment and context in human-robot interaction

Eric Silverman, Takashi Ikegami, Yuta Ogai and Jean-Julien Aucouturier

University of Tokyo
erics@sacral.c.u-tokyo.ac.jp

Most dancing robots to date have used patterns of preprogrammed motions or hard-coded interaction rules to produce this behaviour. In a departure from this approach, recently we used a form of embodied chaotic itinerancy (Ikegami, 2007, J. Consc. Studies, 14, p.111; Kaneko et al., 2003, Chaos, 13(3), p. 926) to generate motor movements for a robot in real time. We used the robot's sensors to analyse audio input, processing it at regular time intervals to find the appropriate tempo, and used this information to send input pulses to a FitzHugh-Nagumo neural network model (Aucouturier et al., 2007, Proc. of the 14th ICONIP, Springer-Verlag). The dynamic properties of these neurons allow for interesting chaotic behaviour, as some inputs will produce entrained periodic states, while others produce chaotic or aperiodic responses.

The output neurons of this network drove the motors of our chosen robotics platform Miuro, a simple two-wheeled vehicle robot manufactured by ZMP (Tokyo, Japan). Despite the deterministic nature of the mechanisms driving Miuro, the resultant motions are heavily dependent on the music being played, and thus the robot displays complex transitions between quasiperiodic states of motion. The robot is able to demonstrate both synchronization and autonomy in its reactions to the music.

Currently we are organizing a workshop together with art students to invent a new type of environment for robots. In particular, we are aiming to generate "natural" sound environments in which mobile robots can generate complex and interesting dancing patterns. The workshop will also investigate the role of physical form in driving interaction between autonomous robots and human observers. The robot will interact with human observers in the same method as above, through sound, but our investigation of radical and novel physical forms for the robot will allow us to investigate new varieties of agent-environment couplings.

Ikegami and colleague Keiichiro Shibuya have started a series of sound installations, each of which uses ideas of Artificial Life and complex systems science to make unique soundscapes (http://sacral.c.u-tokyo.ac.jp/index.php?Third%20Term%20Music). Using robots, we can further develop this enlightening cooperation between science and art, which we think is a promising future avenue of artificial life study. Our collaboration with people from Art University will also encompass this issue, examining the concept of open-endedness in artificial life studies and developing new methods for generating sound arts.

In both cases, our future work hinges on developing a new understanding of the robot-environment-human relationship. Through analysis of our work thus far, and discussion of the multitude of conceptual issues we have investigated with the artistic community, we will demonstrate new ways in which to examine these relationships. The interplay between form and function, between observer and performer, and between context and action will all influence the development of both the robot's morphology and its control structure.

The virtual rocky shore — linking A-life with ecological and pedagogical research

Richard Stafford[1], Mark S. Davies[2] and Gray A. Williams[3]

[1]University of Gloucestershire
[2]University of Sunderland
[3]University of Hong Kong
rstafford@glos.ac.uk

The design and analysis of manipulative experiments is a key skill for undergraduate ecologists to learn. Despite this, students rarely conduct a true manipulative experiment with appropriate control treatments in the field, because of practical constraints such as time. Meaningful ecological experiments typically require months to obtain results, as well as constant maintenance and attention; and often require the use of large areas of undisturbed habitat. One solution to this problem is to allow students access to a virtual ecosystem in which they can conduct a possibly unlimited range of experiments which will quickly provide realistic data for subsequent statistical analysis and interpretation.

Currently, virtual ecosystems fall into two basic categories: those that are vast oversimplifications of real ecosystems, displaying simplistic and pre-programmed behaviours; and those that bear little resemblance to real ecosystems, made entirely of interacting digital organisms. While the latter are of more interest to the A-Life community, they are not user-friendly to biology students who are generally not computer literate beyond the basics of word processing, spreadsheets and internet technologies.

"The Virtual Rocky Shore", is grounded in a variety of A-life techniques, including agent-based modelling, self-organisation, evolutionary algorithms and cellular automata. The present version of The Virtual Rocky Shore is based on the high intertidal region, a simple consumer / resource ecosystem consisting of grazing snails and a photosynthetic mat or biofilm of lichens, diatoms and bacteria. Although the system is simple, research underpinning the system's models has demonstrated that these intertidal snails show many similar behavioural rules to those displayed by classic A-Life inspirations such as ants. The intertidal snails, for example, exhibit self-organisation as a result of trail following. The dynamics of the photosynthetic components of the virtual shore are also suitable to being modelled in space and time by use of cellular automata and computer-based optimisation processes including evolutionary algorithms.

Current development of The Virtual Rocky Shore using a user-friendly interface has already provided novel insights into the functioning and evolution of intertidal communities; with many of these insights backed up by empirically derived data from real shores and published in peer reviewed journals.

The first implementation of the Virtual Rocky Shore allows experiments to be designed and analysed in a matter of minutes, rather than the many months traditionally required, facilitating the active and potentially deep experiential learning of experimental design by students. The results obtained from the simulation are also similar, and result in comparable statistical analysis, to those obtained from experiments on real shores.

The full potential of The Virtual Rocky Shore, however, lies in its expansion to cover the mid and lower shore systems. The complexity found at these shore levels will allow many opportunities for further research at the interface of ecology and computer science, as well as the development of a wide range of potential experiments beyond simple grazer / biofilm interactions.

A minimal approach to modular assembly

Gregory Studer and Inman Harvey

University of Sussex
gms21@sussex.ac.uk

In this work, we propose a new research direction into minimal assembling agents. Our goal is to use very simple, inflexible assembling units to form complex and flexible assemblies (or meta-modules), guided by global environmental signals. Instead of the focus in modular robotics and self-assembly on creating maximally flexible and programmable assembling units (Yim et al., 2007, IEEE Rob. Aut. Mag., p.43), we suggest a different, complementary approach in which assembled structures maintain or enhance the range of assembly behaviors atomic agents are capable of. Replacing the idea of complex autonomous modules which are able to build arbitrary structures, like cells building organisms, we are beginning to simulate robotic platforms which themselves have rather limited assembly behavior but stochastically form structures or meta-modules with more complex interactions, like proteins built from interactions of a few amino acids. This is inspired by stochatic assembly results in the real world at any scale (Krishnan et al., 2007, Proc. ASME IMECE, ASME.org), (Winfree et al., 1998, Nature, 394, p.539), with an emphasis on understanding how function develops in these semi-controllable environments.

As a proof-of-concept and to gain intuition into how such units might look, we use a microbial genetic algorithm (MGA) to evolve the logic placed on simulated assembling agents. The agents are modeled as very simple units containing male (M) and female (F) assembly ports, as well as an input sensor, each of which may be in one of two states: enabled or not-enabled. Logic (in the form of Petri Nets) is generated by the MGA and identical copies placed on each agent, which are then allowed to assemble into chains in a well-mixed stochastic environment. Limited communication can occur between assembled agents' M and F ports. Instead of a traditional fitness function, however, where we might evaluate a logic as highly fit if it performs a particular assembly task, our fitness function rewards logics that maintain assembly behavior as the units assemble. In particular, we reward logic that maintains pairing behavior in response to a "start" signal. First, we enable the input sensor on all the agents, which may then form assembled structures including pairs which add to the fitness. If there are pairs, we then send a second "start" signal, and pairs may form of the pair structures themselves, and so on until no more pairing occurs. Higher level pairing was rewarded more than lower level pairing.

By limiting the complexity of the generated logics, and comparing the maximum fitness given these limits, there appears to be a lower complexity bound for our particular assembling units to maintain their assembly behavior as they grow orders of magnitude in size. This demonstrates that our initial proposal of designing simple assembling units which build functional assemblies themselves is feasible, at least in some cases. The successful controllers generated are interesting in that they function similarly alone or when linked together in groups of any number of agents: the *behavior* scales. In future work, we hope to expand this result and demonstrate assembly controller designs which generate more complex assembly (and other) behavior as they grow. Our eventual goal is to discover designs for very simple, inflexible units which create programmable and controllable meta-modules in response to global environmental signals.

Evolution of complexity in RNA-like replicator systems

Nobuto Takeuchi and Paulien Hogeweg

Theoretical Biology and Bioinformatics, Utrecht University
takeuchi.nobuto@gmail.com

Background: The evolution of complexity is among the most important questions in biology. The evolution of complexity is often observed as the increase of genetic information or that of the organizational complexity of a system. It is well recognized that the formation of biological organization—be it of molecules or ecosystems—is ultimately instructed by the genetic information, whereas it is also true that the genetic information is functional only in the context of the organization. Therefore, to obtain a more complete picture of the evolution of complexity, we must study the evolution of both information and organization.

Results: Here we investigate the evolution of complexity in a simulated RNA-like replicator system. The simplicity of the system allows us to explicitly model the genotype-phenotype-interaction mapping of individual replicators, whereby we avoid preconceiving the functionality of genotypes (information) or the ecological organization of replicators in the model. In particular, the model assumes that interactions among replicators—to replicate or to be replicated—depend on their secondary structures and base-pair matching. The results showed that a population of replicators, originally consisting of one genotype, evolves to form a complex ecosystem of up to four species. During this diversification, the species evolve through acquiring unique genotypes with distinct ecological functionality. The analysis of this diversification reveals that parasitic replicators, which have been thought to destabilize the replicator's diversity, actually promote the evolution of diversity through generating a novel "niche" for catalytic replicators. This also makes the current replicator system extremely stable upon the evolution of parasites. The results also show that the stability of the system crucially depends on the spatial pattern formation of replicators. Finally, the evolutionary dynamics is shown to significantly depend on the mutation rate.

Conclusions: The interdependence of information and organization can play an important role for the evolution of complexity. Namely, the emergent ecosystem supplies a context in which a novel phenotype gains functionality. Realizing such a phenotype, novel genotypes can evolve, which, in turn , results in the evolution of more complex ecological organization. Hence, the evolutionary feedback between information and organization, and thereby the evolution of complexity.

[The original article is published as Takeuchi & Hogeweg, 2008, Biology Direct, 3:11]

How to know without having been there? — Investigating communication channels in the nectar collecting system of a honeybee colony

Ronald Thenius, Thomas Schmickl and Karl Crailsheim

Department for Zoology, Karl-Franzens-University Graz, Universittsplatz 2, A-8010 Graz, Austria
ronald.thenius@uni-graz.at

Honeybees collect nectar from flowering plants in the environment to accommodate their energetic demands. In a honeybee colony a temporal caste, "foragers", is collecting nectar. These foragers bring their harvest into the colony, where they unload their nectar loads to one or more specialised "storer bees", another temporal caste in the colony, responsible for the next step of nectar processing. Natural selection has shaped the foraging-related processes of honeybees, like the communication between foragers via dances, in a way that a colony can react to changing environmental conditions in an adaptive way. To investigate this complex dynamic social system and the information and nectar channels we developed a multi-agent model of the nectar flow inside and outside of a honeybee colony. This model allows us to investigate the nectar collection process and nectar processing pathways on the colony level, as well as from the point of view of a single bee during the foraging trip and during the nectar proceeding inside the colony. The simulation includes near-natural environmental factors, like scattered nectar sources with variable distances between flowers and a near-natural model of the honeybee metabolism. The inside of the colony (the so called "dance floor") was simulated as two one-dimensional transfer zones for foragers and storers, what enabled us to simulate the unloading-procedure in a highly abstract and defined manner. Our model predicts that a cohort of foragers, collecting nectar from a single nectar source, is able to detect changes in quality (e.g., the nectar flow) in other food sources they have never visited, by analysing side-effects of the nectar processing system of the colony: We identified two novel pathways of forager-to-forager communication by analysing the results predicted by our model. Foragers can gain information about changes in the nectar flow in the environment via two ways: Firstly, foragers can detect changes in their mean waiting time for unloadings, which are performed by the storer bees. Secondly, the foragers can detect changes in the number of experienced multiple unloadings after returning from a foraging trip. The amount and quality of information available to the single forager about the environmental situation is increased, what enables the forager to modulate its individual decisions. The sum of this modulated forager decisions can lead to an optimisation of the foraging behaviour in an unsteady environment. This way, two distinct groups of foragers, that forage on different nectar sources and that never communicated directly, can share information via a third cohort of worker bees. We show that the communication channels within this noisy social network allow the colony to perform collective information processing. Simulation runs with fluctuations in the environmental nectar flow revealed, that the honeybee foraging system is even more adaptive (by exploiting the before mentioned communication channels) than was previously thought.

Towards language acquisition in autonomous robots

Vadim Tikhanoff[1], Angelo Cangelosi[1], Jun Tani[2] and Giorgio Metta[3]

[1]University of Plymouth, UK
[2]RIKEN Brain Science Institute, Japan
[3]Italian Institute of Technology
vadim.tikhanoff@plymouth.ac.uk

In this paper we present a novel cognitive robotic model to study language acquisition in autonomous robots through the grounding of words in sensorimotor representations. The aim of this new model is to extend previous work on language grounding in simulated cognitive agents to the new robotic platform iCub. The iCub is an emerging open platform for cognitive robotic research that will allow research groups to exploit a common hardware and software infrastructure in order to advance knowledge of natural and artificial cognitive systems. The language learning model is based on the use of artificial neural networks controllers. The model is based on a series of interconnected modules to gather and integrate visual and linguistic information for a language comprehension task. The model comprises of a vision module, a sound perception and feature extraction module and a language integration and recognition network. The vision acquisition module takes input from the robot's cameras and applies approximation techniques for the purpose of detecting shapes, size and colour features of individual objects. The classification of a spoken word is based on the sequence of the most activated neurons of a self organizing map (SOM) with a 10 x 10 topological 2D grid. The SOM model has been trained on 112 English words and 544 syllable utterances both from two different speakers, for determining the ability of the system to distinguish between all words. The language integration module is based on a Recurrent Neural Network with Parametric Biases (RNNPB). This network is particularly suitable for online learning of behaviour in robots. Two experiments were carried out to test the language learning model. The first consists in the recognition and classification of the speech signals as an imitation task without the integration of the vision module. This experiment has been based on the use of 20 words. Each training patterns (words) consists of a sequence of x/y coordinates of the SOM map. During the interaction phase of RNNPB training, the system learns to imitate the SOM word feature outputs pattern by predicting their next pattern. The network successfully learns to recognize spoken words with a final mean square error of the output nodes of 0.082. The second experiment consists of the integration of the vision and speech modules for learning and grounding of the names of objects. This experiment uses as input stimuli the combination of the features extracted from the visual module and the SOM output patterns. The output units predict the SOM sequence for the object name shown in the picture. The final square error of the output nodes was 0.003 over all the learning results. The model was able to categorize and name two objects which share some features (e.g. shape) but differ in other dimensions (e.g. colour). This preliminary work demonstrates the successful integration of a SOM network to classify spoken words with the RNNPB network capable of on-line learning and naming of visual objects. This model is being extended to include the learning of motor responses to be associated to the visual input of different objects and the capability to combine groups of words to describe visual scenes involving multiple objects. Although the current model primarily focuses on the naming aspects of language, our future plans include work on linguistic and communication capabilities.

Complex consumer preferences from rudiments of visual processing

Colin Tosh[1], Jens Krause[1] and Graeme Ruxton[2]

[1]IICB, University of Leeds
[2]University of Glasgow
c.r.tosh@leeds.ac.uk

Humans and many other consumer animals such as predators tend to select similar salient, pop-out, resource items from a visual scene. This observation hints at a common, simple mechanism underlying visual attention and driving the evolution of visual apparatus. Using simple artificial neural networks, we demonstrate that when information degrades early in a neural apparatus, and this degradation is compensated for in higher layers, many of the distinctive behaviours of consumer organisms emerge. These include preference for odd-looking resource items, resources that are spatially isolated, and resources that are on the edge of groups. We also observe evolution of a primitive visual fovea. While visual attention is structurally and mechanistically complex in humans, the fundamental mechanisms driving the evolution of visual apparatus across different animal species may be simpler.

Self-organising synchronisation in a robotic swarm

Vito Trianni and Stefano Nolfi

Institute of Cognitive Sciences and Technology, CNR, Italy
vito.trianni@istc.cnr.it

Synchrony is a pervasive phenomenon: examples of synchronous behaviours can be found in the inanimate world as well as among living organisms (Strogatz, 2003, Sync, Hyperion Press). The synchronisation behaviours observed in Nature can be a powerful source of inspiration for the design of swarm robotic systems, where emphasis is given to the emergence of coherent group behaviours from simple individual rules. Much work takes inspiration from the self-organised behaviour of fireflies or similar chorusing behaviours. Here, we present a study of self-organising synchronisation in a group of robots based on minimal behavioural and communication strategies. We follow the basic idea that if an individual displays a periodic behaviour, it can synchronise with other (nearly) identical individuals by temporarily modifying its behaviour in order to reduce the phase difference with the rest of the group. In other robotic studies, synchronisation is based on the entrainment of the individual internal dynamics through some form of communication (see for instance Wishmann et al., Adaptive Behaviour, 14(2), p.113). In this paper, instead, we do not postulate the need of internal dynamics. Rather, the period and the phase of the individual behaviour are defined by the sensory-motor coordination of the robot, that is, by the dynamical interactions with the environment that result from the robot embodiment. We show that such dynamical interactions can be exploited for synchronisation, allowing to keep a minimal complexity of both the behavioural and the communication level. In order to define a robot controller able to exploit such dynamical agent-environment interactions, we use artificial evolution (Nolfi and Floreano, 2000, Evolutionary Robotics, MIT Press). The obtained results are analysed under a self-organising perspective, evaluating their scalability to large groups of robots.

The main contribution of this work consists in the analysis of the evolved behaviours, which is brought forth exploiting a dynamical systems approach: We introduce a dynamical system model of the robots interacting with the environment and among each other. This model offers us the possibility to deeply understand the evolved behaviours, both at the individual and collective level, by uncovering the mechanisms that artificial evolution synthesised to maximise the user-defined utility function. Moreover, we show how the developed model can be used to predict the ability of the evolved behaviour to efficiently scale with the group size. We believe that such predictions are of fundamental importance to quickly select or discard obtained solutions without performing a time-demanding scalability analysis, as well as to engineer swarm robotic systems that present the desired properties.

Mammalian decisions

Pete Trimmer, John McNamara, Alasdair Houston, James Marshall, Liz Paul, Rafal Bogacz and Mike Mendl

University of Bristol
trimmer@compsci.bristol.ac.uk

We study a simple scenario in which optimal decision-making requires a trade-off between speed and accuracy. Our analysis is set in the context of a mammal deciding whether to forage or invoke anti-predator action following an ambiguous cue. We assume that the brain has two systems which can make decisions. Thalamic decisions are fast but are less accurate than cortical decisions (which take longer).

We idealise the analysis by assuming that: 1) Thalamic decisions are made immediately, based upon a single piece of information (represented using Signal Detection Theory). 2) Cortical decisions are made by gathering information continuously until a confidence-threshold is reached (represented by applying a single-boundary version of the Sequential Probability Ratio Test to Brownian Motion with drift).

Following the analysis of each process in isolation, we examine how such decision systems might best be combined and used in the brain, discussing results in the context of information flow and the phylogeny of the mental architecture. We find that in some circumstances, if one system is weakened, the other system can largely compensate, thereby producing similar overall performance but with a different likelihood of response and decision timing.

The work may help to open areas of research on several topics, such as selective attention, mental stress and how learning affects decision-making (and vice-versa).

Modelling the social coordination of behavior with public symbols

Ulas Türkmen and Radomir Zugic

University of Osnabrück
utuerkme@uos.de

In the recent decade, there has been an upsurge in models of language evolution. These models are based on the idea that multi-agent dynamics coupled with certain simple capabilities can lead to basic linguistic behavior. The general organization of these models, however, betrays an idea common to the cognitivist paradigm, namely that the linguistic symbols used by humans are labels for inner representations. The experiments in which these models are tested consist of two separate stages, which are (1) generating labels for sensory data and (2) using these labels for communication. The models also reflect this division, and consequently contain separate modules (1) for creating representations and (2) for creating and transmitting vocalizations that correspond to these representations. An alternative to this division between perception/categorization and symbolic representation is using situated representations. The main idea behind situated representations is that the symbols human beings use in communication serve not only to carry meaning, but also to coordinate their embodied interaction. Inherent in this view is the social shaping of embodied activity through linguistic symbols. Traditionally, human cognition is divided into contrasting classes of high- and low-level processes, which then share inner representations among them in the mediation of thinking and action. Situated representations implicate that the relationship between high- and low-level processes, traditionally posited to be linear, rather is a dialectic one, where the social being of the agent affects, but is at the same time formed by, its embodied activity.

In order to model situated representations with multiple agents, the microworlds methodology used in the early days of artificial intelligence is ideal. In such a setup, it is possible to study in a task environment the necessary components of symbolic intelligence, namely social situatedness and ecological relevance. In the case of the model presented here, a robotic approach was chosen in order not to abstract away from the sensory-motor aspects of intelligence. Our approach to categorization is called "categorization without categories", which refers to avoiding internal representations which do not have a linguistic function. In order to implement it, we have used an exemplar-based mechanism which relied on a simple similarity measure and the storing of whole sets of sensory data.

The principle idea of our experimental setup is providing the robot with an environment with a number of choices. In the first part of the experiment, the agents learn picking one of the choices, using solely the possibilities of the environment as cues. The second part starts with teaching one of the agents making a certain choice. Afterwards, a language game is played in which this agent instructs the other one. By developing a mechanism to instruct and to be guided by instructions, one agent can profit from the learning experience of the other and can directly choose the best out of a number of possible behavioural alternatives without having to go through the same learning experience again.

Using the RNA sequence-to-structure map for functional evolution of ribozyme catalyzed artificial metabolisms

Alexander Ullrich, Christoph Flamm and Lukas Endler

University of Vienna
aullrich@tbi.univie.ac.at

We introduce a novel genotype-phenotype mapping based on the relation between RNA sequence and its secondary structure for the use in evolutionary studies. The inspiration for this particular mapping emerged from the modeling of RNA enzymes within a simulation framework for the evolution of metabolic reaction networks. In our simulation we allow individuals, containing a genome and a metabolism, to evolve. The genome contains a number of RNA genes which then give rise to RNA enzymes acting on metabolites and thus shaping the metabolic network. Individuals are selected based on measures of this network and new individuals with mutated genomes are created. The use of our mapping allows not only for a more realistic study of the evolution of the entire system, but also enables us to observe the behavior of our enzymes itself and therefore possibly gain some insights about the evolution of catalytic molecules in general.

Enzymes typically have an active site where only few amino acids or bases determine its catalytic function and the remaining structure has mostly stabilization function. Accordingly, we extract structural and sequence information only from a restricted part of the fold. We decided to focus on the longest loop of the folded RNA. The idea for mapping the extracted information to a specific chemical reaction was encouraged by the fact that many enzymes catalyze a reaction by stabilizing its transition state. Recent work on hairpin ribozymes and other catalytic RNA support that as a common strategy for RNA enzymes. Given the definition of Fujita's imaginary transition structures (ITS), we developed a unique index for all possible pericyclic chemical reactions, describing the constitution of the reaction's transition state. Every RNA molecule is assigned such an reaction ID based on the information from its fold. The length of the longest loop specifies the number of involved atoms and the sequence within the loop determines the atom types. The bond types are derived from structural characteristics of the loop, such as the length and position of contained stems. Thus, a mapping from RNA sequence (genotype) to a chemical reaction (phenotype) is produced.

For many years it is known that neutral mutations have a considerable influence on the evolution in molecular systems. The folding of RNA sequences to secondary structures with its many-to-one property represents a mapping entailing considerable redundancy. Various extensive studies concerning RNA folding in the context of neutral theory yielded insights about properties of the structure space and the mapping itself. We intend to get a better understanding of some of these properties and especially of the evolution of RNA-molecules as well as their effect on the evolution of the entire molecular system.

Besides using the mapping in several simulation runs which yielded realistic metabolic networks and connectivities, we performed several statistical tests commonly used in neutral theory, such as the number of visited phenotypes and the average discovery rate during a random neutral walk. We compared it with results of approaches using cellular automatons, random boolean networks and other mappings based on RNA folding. It exceeds all non-RNA mappings in extent and connectivity of the underlying neutral network. Further, it has a significantly higher evolvability and innovation rate than the rest. Especially interesting is the highly innovative starting phase in RNA-based mappings.

Modelling resilience of agro-ecosystems: the co-evolution of regimes

Dirk van Apeldoorn, Kasper Kok, Marthijn Sonneveld and Tom Veldkamp

Wageningen University, Land Dynamics
dirk.vanapeldoorn@wur.nl

Understanding how natural resources co-evolve with management practices is fundamental to all questions concerning their sustainable management. In the search for new pathways of sustainable development we seek to understand how comparable initial conditions in agro-ecosystems lead to fundamentally different resource conditions. Agro-ecosystems are characterized by a high diversity, path dependence, self organization, cross-scale interactions and non-linear feedbacks. The resilience perspective claims it offers insight in these complex systems attributes. For exploring these dynamics we used a summary model approach. A summary model approach aggregates processes based on detailed knowledge, while allowing for integration of multiple scales and the identification of thresholds.

We analysed an area characterised by dairy farming in the Netherlands, within which two different farming systems can be found. One system is characterised by the modernisation paradigm with benefits of scale, intensification and specialisation, while the other system is characterised by low external input and less intensive farming. For a long time it was thought that the latter, alternative, system represented the laggards of the adoption rate of innovations. Scientific attention to the development of this region was attracted by the persistence of the alternative farming system, the increasingly valued effect on the landscape by these farms, and their mismatch with "modern" environmental regulations.

We developed a summary model that integrates the soil, feed, and animal compartments of the farming system. The model enables us to simulate the effects of farm management decisions on the key natural resources soil carbon and nitrogen. When characteristic farm management of both the intensive and the alternative farming systems are used as model input, the systems evolve to different regimes. I.e., the different management systems lead to natural resources that respond differently to external drivers.

The two management systems evolved to different natural resource regimes and traverses to the other regime are slow, highly non-linear and involve large costs. Policies aiming at social-ecological regime change are currently not aware of these dynamics. We are now discussing with farmers strategies for regime change employing the non-linear dynamics in their system. With researches in the area we are formulating new hypothesis on system functioning. Meanwhile at the governance level we show how the environmental regulations, although initially successful, will actually drive the systems to an unwanted regime.

Diet learning and the evolution of cultural inheritance

Daniel van der Post, Bas Ursem and Paulien Hogeweg

Utrecht University
d.j.vanderpost@uu.nl

Social learning can give rise to cultural inheritance which forms an additional inheritance system next to genetic inheritance. Its evolution can be seen as a major transition in evolution. Using a spatial individual-based model we study the evolution of social learning and therewith the emergence of culture. We focus on diet learning in group foragers as a context in which cultural inheritance could have evolved. We model a rich environment in which foragers learn what to eat and focus on how environmental complexity can structure behavioural opportunities and lead to self-organizing processes. Our results show that social influences on learning arise as obligate side-effects of grouping. In patchy environments this can give rise to both traditional inheritance and cumulative cultural processes. Cultural phenomena therefore arise "for free" as soon as individuals learn by trial-and-error in groups. This shows the role of self-organizing processes in generating novelty in evolution. These self-organized processes set the context in which more sophisticated forms of social learning can evolve. By including copying behaviour in our model, we studied its adaptive influence and evolution. Results show that copying is not a fixed strategy and its adaptive value depends on resource distributions in the environment. On the one hand copying leads to collective problem solving within lifetimes. On the other hand it generates cumulative cultural diet optimization over lifetimes. Preliminary results of evolutionary simulations show that copying behaviour evolves because it allows for these adaptive processes. However copying also tends to reduce variation in groups and thus reduces the efficacy of natural selection. We conclude that self-organization plays a large role in the transition to cultural inheritance by means of generating obligate social influences on learning as side-effects of grouping. Moreover, this self-organized baseline affects the evolution of cognitively more sophisticated forms of social learning.

Reaction-diffusion spots as a model for autopoiesis

Nathaniel Virgo and Inman Harvey

University of Sussex
nathanielvirgo@gmail.com

We analyse pattern formation in reaction-diffusion systems from an autopoietic point of view, emphasising the commonalities between living organisms and a certain class of so-called dissipative structures, namely those (such as spot patterns or hurricanes) in which there are more-or-less clearly defined unities, or individuals, which arise from the system's dynamics.

Previous authors have used cellular automata as a basis for studying the emergence of autonomous agent-like structures, but the continuous nature of reaction-diffusion systems gives them a substantial advantage over discrete cellular automata as it enables systems to be perturbed by an arbitrarily small amount. Since reaction-diffusion systems are simulations of physical/chemical systems the resulting model agents must obey the relevant thermodynamic constraints, an aspect of living systems that has generated a lot of recent discussion in the autopoietic literature.

The Gray-Scott model is perhaps the simplest reaction-diffusion system that can create complex patterns; it models a single type of autocatalyst feeding on a 'food' chemical that is continually added to the system; both are able to diffuse on a two-dimensional surface. One of the patterns that can be formed consists of blurred but individuated "spots" of autocatalyst separated by regions in which the autocatalyst is absent. We take a single spot as the basis for our model agent.

With the autopoietic description in mind we perform three experiments. Firstly, we put these spots into situations where there is a spatial gradient of the food molecule and find that they tend to move along it, usually away from areas where the level of food is too low for their survival. The relationship between constitution and behaviour is fundamental to the autopoietic theory, and this result opens the possibility of studying the interface between the two empirically.

Secondly we vary the rules of the system, allowing a different set of chemical reactions, which can result in agents with a more complex anatomy than just a single spot, and even a very limited form of heredity.

Finally we find that individuated spots are very likely to arise when there is a negative feedback between the whole system's activity and its overall supply of food. This situation is common in natural systems, and our result suggests a direction for further research into the conditions under which individuated unities are likely to occur in general.

Can we rebuild a cell? Bryopsis — an experimental model!

Alexandru Vladimirescu

National Institute for Research and Development for Microbiology and Immunology, Bucharest , Romania
AlexandruVl@yahoo.com

Generally speaking a violent mechanical treatment applied on a living cell or a unicellular organism destroys its structural and functional integrity and leads to death; any tentative to re-build the destroyed cell from the remaining cellular fractions turned into failure. There are few exceptions; one of them is the coenocytic seaweed Bryopsis. This widespread seaweed has the ability to be restored from its cellular fractions beginning with the spontaneous aggregation of cytoplasm and organelles (in the presence of seawater) and continuing with the formation of a temporary polysaccharide membrane surrounding the cytoplasm aggregates, formation of a lipid-based membrane and restoration of the cell wall; the result is a cell that has the ability to growth and form a new Bryopsis thallus. In the experimental approach to re-build the coenocytic alga Bryopsis, in early events after the mechanical destruction, cytoplasm and organelles can be mixed with biological (E. coli living cells caring the gfp gene for the Green Fluorescent Protein) or inorganic particles (Fe3O4 nanoparticles for Ferrofluids) so that the new particles would be incorporated in the re-constructed Bryopsis protoplast/cell. The behavior of Bryopsis protoplasm and particles will be presented and discussed using Optical and Transmission Electron Microscopy investigations. The preliminary experimental data support the belief that the reconstruction of a designed Bryopsis cell, including artificial or natural foreign elements, will become a reality in the nearest future.

Can individual selection favour significant higher-level selection?

Richard Watson, Rob Mills, Alexandra Penn and Simon Powers

Science and Engineering of Natural Systems group, School of Electronics and Computer Science,
University of Southampton
raw@ecs.soton.ac.uk

How do new evolutionary units, supporting higher levels of functional organisation, arise from existing evolutionary units? The adaptive transformation of co-adapted species into new units, as in the major evolutionary transitions, is centrally implicated in the evolution of complexity but has proved very problematic for current evolutionary theory and understandably elusive in ALife. We investigate the evolution of new evolutionary units via individual adaptation in a multi-species ecosystem by modelling symbiotic associations that cause interaction probabilities to deviate from a freely mixed condition. It is well known that assortative grouping supports group selection in a well-defined sense, thus it is no surprise that enabling such associations will introduce some group selection effects. However, what form will this take when the control of such grouping is under individual adaptation? We tackle this by comparing the ecosystem attractors of the initial freely mixed system to those of the same system given the evolved association probabilities. In general we find that self-interested adaptation of associations tends to only reinforce species combinations that were already stable before the associations — which seems rather uninteresting. However, if the species densities of the ecosystem are occasionally perturbed whilst associations are developing this causes the system to visit different attractors and allows multiple, possibly incompatible, associations to be selected for in different contexts. Under these conditions, even when the attractors of the final system already existed as attractors in the freely-mixed system, competition between different combinations of species enlarges basins of attraction that lead to fit combinations at the expense of those that lead to less fit combinations. Thus, after the associations have evolved, a fit combination of species may be favoured in the niche that is constructed by the action of its association preferences, even if each species involved would be individually unfit if the system were freely mixed.

These findings show that evolved higher-level selection can have significant effects even when the new units result from the self-interest of the constituent sub-units. They also suggest that evolved complexes observed naturally may appear to be merely the result of individual selection because they are supported by individual self-interest, but in fact the reason that this complex persists and not some other is due to competition among species combinations.

Nonetheless, in small systems these mechanisms do not produce higher levels of complexity than those which occurred without evolved associations because the configurations that result were already visited in the initial freely-mixed system. However, we find that in large complex ecosystems with many local attractors, evolved associations naturally generalise over the relatively few attractors that are visited, enlarging attractors for fit species combinations even before they are visited. An idealisation of these processes has been shown to be far superior to conventional evolutionary algorithms on a fairly general class of difficult optimisation problems. This self-modification of ecosystem attractors therefore illustrates a mechanism that produces high-fitness biological complexes despite the fact that their evolution would seem highly implausible given the very small size of the basin of attraction that leads to this configuration under selection on the original units.

Autopoiesis, enactivism, and the extended mind

Michael Wheeler

Department of Philosophy, University of Stirling
m.w.wheeler@stir.ac.uk

Two strands of recent embodied theorizing about cognition that are commonly held to be in harmony are actually in tension. This tension arises, in part, from the different ways in which the two positions in question — the extended mind hypothesis (EMH) and enactivism — conceive of the relationship between life and mind.

The history of enactivist ideas, plus a recent presentation of the view by one of its architects, Evan Thompson, suggest that the autopoietic theory of Maturana and Varela is a non-negotiable component of enactivism. An autopoietic system is a self-organizing autonomous system that, through its own endogenous activity, produces and maintains a physical boundary that distinguishes that system as a material unity in the space in which it exists. According to Maturana and Varela: (i) any living system is an autopoietic system; (ii) any autopoietic system is a living system; (iii) cognition is viability-maintaining activity in a domain of interactions defined by an autopoietic system's organization; (iv) enaction is the process by which significance is brought forth through the viable structural coupling of an autopoietic system with its environment.

Against this background, one striking claim made by autopoiesis theorists is that living is cognition. A natural way of hearing this claim (one that Maturana and Varela themselves often seem to recommend) is as asserting that the living system is identical with the cognitive system. If we add to this identity assertion the independently plausible thought that the living system (the organism) will be bounded by its skin, the implication is that, for the enactivist, the cognitive system is bounded by the skin. However, according to EMH it is possible for things and processes located beyond the skin sometimes to count as the proper parts of a cognitive system, which means that the boundary of the cognitive system may sometimes extend beyond the skin. So, it seems, the enactivist cannot endorse EMH.

The enactivist might reply that I have painted an impoverished picture of the relationship between life and cognition, as she understands it. Varela, in later work, depicted cognition as a process of sense-making. Di Paolo has argued that to explain sense-making, raw autopoiesis (autopoiesis as described above) must be supplemented with a capacity for adaptivity, itself established on the basis of an autopoietic organization. As I understand him, Di Paolo holds that being a raw autopoietic system is necessary but not sufficient for being a cognitive system (for realizing sense-making). However, since being a raw autopoietic system remains necessary and sufficient for being a living system, being a cognitive system remains sufficient for being a living system. So if our reconstructed enactivist did try to sanction EMH, she would be claiming (a) that an extended cognitive system is an autopoietic system, and (b) that an extended cognitive system is itself (it does not merely contain) a living system. Claim (a) is debatable and claim (b) violates our highly plausible thought that living systems don't extend. The enactivist still cannot endorse EMH. Enacted minds are not extended minds.

Environmental regulation by higher level selection in a simulated network of microbial ecosystems

Hywel Williams and Tim Lenton

University of East Anglia
h.williams@uea.ac.uk

The Earth possesses a number of regulatory feedback mechanisms involving life. In the absence of a population of competing biospheres it has proved hard to find a robust evolutionary mechanism that would generate environmental regulation. It has been suggested that regulation must require altruistic environmental alterations by organisms and would therefore be evolutionarily unstable. This need not be the case if organisms alter the environment as a selectively neutral by-product of their metabolism, as in the majority of biogeochemical reactions, but the question then arises: why should the combined by-product effects of the biota have a stabilising, rather than destabilising, influence on the environment? In certain conditions selection acting above the level of the individual can be an effective adaptive force. Here we present an evolutionary simulation model in which environmental regulation involving higher level selection robustly emerges in a network of interconnected microbial ecosystems. The Flask model simulates an evolving microbial community suspended in flasks of liquid with prescribed inputs of nutrients. The system is seeded with a clonal population of 'microbes' that are subject to mutation on genetic loci that determine their nutrient uptake patterns, release patterns, and their effects on, and response to, other environmental variables. Nutrient recycling loops robustly emerge from local adaptation, but populations are vulnerable to crashes caused by 'rebel' mutants which push abiotic conditions away from habitability. In previous work we have demonstrated a community-level response to artificial selection at the level of a single flask. Here we show that spatial structure in a network of interconnected flasks creates conditions for a limited form of higher level natural selection to act on the collective environment-altering properties of local communities. Local communities that improve their environmental conditions achieve larger populations and are better colonisers of available space, while local communities that degrade their environment shrink and become susceptible to invasion. The spread of environment-improving communities alters the global environment towards the optimal conditions for growth and tends to regulate against external perturbations. This work suggests a new mechanism for environmental regulation that is consistent with evolutionary theory. Interestingly, the system appears to be ultrastable — a term originally introduced in cybernetics by W. Ross Ashby — in that its stability requires the maintenance of key variables within bounds. We speculate that the biosphere may also be ultrastable.

Timing of critical periods in development

Oliver Winks and Luc Berthouze

University of Sussex
L.Berthouze@sussex.ac.uk

Critical periods are specific periods in the development of a living organism during which there is an increased sensitivity to external perturbations. Such perturbations result in a developmental trajectory significantly different from what is considered the norm. This study is concerned with the question of whether the presence and timing of a critical period can be predicted from the developmental profile without perturbation. We frame this question in the context of Waddington's epigenetic landscapes (Waddington, 1943, Am. Midl. Nat., 30, p. 811) and put forth the hypothesis that bifurcations are more likely to take place when the system is undergoing rapid developmental changes, i.e., critical periods will occur when the rate of change is greatest.

To test this hypothesis, we developed a simulation of the early stages of embryonic development, specifically, the development of cellular structures and cell differentiation. The model was formed of two components, the genetic component and the cellular component. The genetic component simulated gene expression and genetic regulation where artificial transcription factors and proteins were synthesised that excited and inhibited genes. The cellular component simulated several cell functions that were controlled by proteins and that made it possible to grow cellular structures composed of cells of different types. The model existed in 3D space allowing for complex 3D structures to emerge over a fixed time period through the dynamics of differential gene expression and cellular functions. The amount of energy available to the system was kept constant so that energy consumption was a limiting factor that stopped the physically impossible scenario of infinite growth. Artificial evolution was used to create genomes capable of growing into organisms of a specific structure, and genome size was varied to allow for organisms to develop differently into their final structure. The presence of critical periods was tested by systematically depriving each developing organism of varying amounts of an extra-cellular signalling protein at different times between runs and by locating variations in fitness of the organism after development of more than two standard deviations. All organisms were found to exhibit critical periods, and the timing of these critical periods was found to correlate strongly with greatest rates of change in the energy profile of the organism developing without perturbation. Interestingly, these periods of change were linked to discontinuities in the consumption profiles of various signalling proteins, an observation that is consistent with a recent finding in developmental biology that morphogenetic variables are not monotonous in time (Cherdantsev et al., 2005, Ontogenez, 36(3), p. 211). The ability to predict the critical periods of a developing system has broad implications not only in the clinical domain – in particular, the study of teratogens (Wilson, 1973, Environment and Birth Defects, Academic Press) – but also in the study of artificial developmental and adaptive systems.

Time and space in neuronal networks: the effects of spatial organisation on network behaviour

Stephen Womble and Netta Cohen

University of Leeds
womble@comp.leeds.ac.uk

Complex networks in nature are often characterised by nontrivial spatial organisation. In such networks, the topology and spatial organisation are often tightly coupled to the activity on the network. In particular, spatial topologies frequently influence time delays through the network. In many real world networks, from transport systems to brains, increased distance (or wiring length) increases time delays in physical or information flow. Here we investigate this role of space on neural network dynamics.

Time delays are well recognised to influence dynamics in neural networks. It would not be surprising, therefore, if neural networks exploited spatial organisation to tune their time delays and obtain desired patterns of activity. However when we model such networks, either to study or to apply their properties to engineering problems, it is important to distinguish between two cases: either the effects of spatial organisation in the network are merely temporal, in which case the complex and computationally expensive spatial organisation can be succinctly abstracted out, or — more interestingly — spatial organisation may lead to behaviour that cannot be captured solely by temporal effects.

We investigated this question in biologically constrained spiking neural networks operating close to a critical bifurcation between stationary behaviour and population-wide oscillatory behaviour. In particular, we used a network of leaky-integrate-and-fire neurons and conductance based synapses with alpha-function kernels, operating in the so-called balanced regime, where each neuron receives similar excitatory and inhibitory drive. Specifically, we compare the transition from stationary to oscillatory network behaviour as a function of spatial organisation. The two modes of behaviour are found in biological neuronal networks and are important for qualitatively different information coding schemes (e.g. rate coding in the stationary regime and phase or temporal coding in the oscillatory regime).

First, we demonstrate that time delays play an important role in this type of network. We found that the region of parameter space for which stationary behaviour is found increases with shorter time delays. We compared this performance to that of an otherwise identical network but with explicit spatial organisation. In our networks, spatial organisation was implemented by clustering inhibitory cells in the midst of a homogeneous population of excitatory cells, in keeping with dendritic length scales of inhibitory cells in some cortical areas. In addition to introducing a spatial patterning, the central clustering of inhibitory cells also reduces the mean inhibitory-to-excitatory time delays in the network. Thus, a purely temporal effect of clustering would yield similar results to shortening the time delays in the system. In fact, with this spatial network model we found a robust reversal of the network behaviour compared to that of the non-spatial network: the shorter the time delays, the more oscillatory the network becomes. This counter-intuitive result may suggest that the gradient in time delays imposed by spatial organisation may dominate the effect of reducing mean time delays in these networks. In fact, the role of space could be further enhanced (or suppressed) by introducing spatially graded connection probabilities (e.g. with closer cells more likely to be connected).

In conclusion, we have described a novel spatial effect. While the spatial organisation described is plausible, it and the corresponding effect on network dynamics would be interesting predictions to test in cortical networks.

Canalization and environmental engineering

Hajime Yamauchi

Japan Advanced Institute of Science and Technology
hoplite@jaist.ac.jp

Canalization is an umbrella term a leading C. H. Waddington introduced in 1960's. The term describes the ability of a population to buffer disturbances in phenotypic development. There are two major types of canalization based on the sources of disturbances: genetic and environmental. Genetic canalization describes the ability to express the same phenotype from different genotypes, while environmental canalization is to stabilize development under different environmental conditions.

With a graph of norms of reaction, evolutionary process of canalization is expressed as a squashing process of the pattern of phenotypic expression across a range of environments. It is also known that learning (or plasticity, in general) has a similar effect: it would normalize the end state of phenotypic development. However, it has not been thought of a case of canalization as it also enables a genotype to express two distinctive phenotypes (as the term "plasticity" implies).

In this work, we will challenge this view, and examine how learning would potentially provide a third type of canalization. Specifically, this work will show if phenotypic traits are partially inherited via cultural transmission (i.e., learning), and if the function of the trait is to do with social conformities among individuals in a population, it would possibly stabilize the learning environment itself so as for different genotypes to be able to express the same phenotype. This effectively narrows the range of trait variation not by (evolutionarily) modifying the reaction norm itself, but by manipulating the variation of environment the majority of the population would encounter during its learning period. As such, this type of canalization should be categorized neither genetic nor environmental canalization.

We use a multi-agent model which simulates evolution of language. In the model, the population consists of 200 agents allocated on a horizontal space. Each agent acquires her linguistic knowledge with inputs provided from adults' linguistic activities (i.e., communicative activities). Learnability of given linguistic knowledge is sensitive to 1. learner's genetic information, 2. learning ability, and 3. consistency of inputs across different adults. The fitness of an agent is measured by the number of successful communication with her neighbor peers.

With the condition that there is no evolution on the learning ability, the result shows that initially, selection works on genes so that agents' genetic information conforms with the linguistic knowledge dominating the population, as this eases the burden of learning —a classic example of the Baldwin effect. However, as doing so, the population more and more converges into a single linguistic community. This paves the way to smoothing the learning environment (i.e., the collective state of learning inputs) from which later generations learn. Consequently, the environment is now confined itself to provide coherent inputs. Under this circumstance, the population can tolerate a certain degree of genetic disturbances as they can be absorbed by the plasticity which created such an environment itself.

As noted above, this type of canalization is primarily not a process of genetic evolution. Instead, it is an environmental engineering process which modifies the frame of norms of reaction itself.