

THE ANIMAT APPROACH

Simulation of Adaptive Behavior in Animats: Review and Prospect

Jean-Arcady Meyer * and Agnès Guillot **, **

* Groupe de BioInformatique
Ecole Normale Supérieure, 46 rue d'Ulm
75230 Paris Cedex 05 - France
Université Paris 6 - URA 686 CNRS
meyer@frulm63.bitnet

** Université Paris-X
200 avenue de la République
92000 Nanterre Cedex 01 - France
guillot@frulm63.bitnet

ABSTRACT

Following a general presentation of the numerous means whereby animats -- i.e. simulated animals or autonomous robots -- are enabled to display adaptive behaviors, various works making use of such means are discussed. This review cites 176 references and is organized into three parts dealing respectively with preprogrammed adaptive behaviors, with learned adaptive behaviors, and with the evolution of these behaviors. A closing section addresses directions in which it would be desirable to see future research oriented, so as to provide something other than proofs of principle or ad hoc solutions to specific problems, however interesting such proofs or solutions may be in their own right.

1. INTRODUCTION

In a changing, unpredictable, and more or less threatening environment, the behavior of an animal is adaptive as long as the behavior allows the animal to survive. Under the same conditions, the behavior of a robot is considered to be adaptive as long as the robot can continue to perform the functions for which it was built. Now, the survival of an animal is intimately involved with its physiological state and the successful operation of a robot depends upon its mechanical condition. Under these circumstances, it is obvious that one can associate with an *animat* -- whether the term indicates a simulated animal or an autonomous robot (Wilson, 1985, 1987a) -- a certain number of state variables upon which its survival or successful operation depend, and that each of these state variables is characterized by a range of variation within which the animat's continued survival or operation are preserved. Such variables were referred to as *essential variables* by Ashby (1952) long ago. Their variation ranges describe a viability zone inside the given state space, and the animat can be referenced at any instant by a point within this zone. Under the influence of environmental or behavioral variations affecting the animat, the corresponding reference point moves and may at times come close to the limits of the viability zone. In this case, the animat's behavior can be called adaptive so long

as it avoids transgressing the viability boundary (Figure 1) (Ashby, 1952; Sibly & McFarland, 1976).

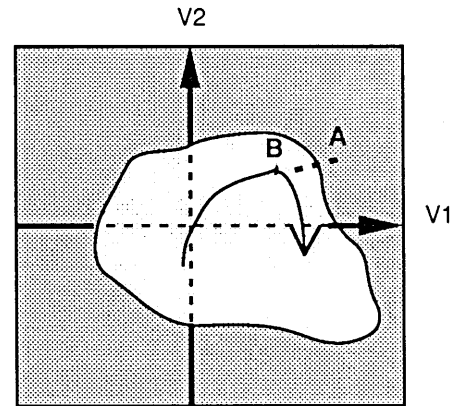


Figure 1. Viability zone associated with two essential variables, V1 and V2. The animat's behavior is adaptive because corrective action has been taken at point B, so as to avoid crossing out the corresponding viability zone at point A.

Such behavior can be generated by means of several different or complementary abilities and architectures. For example, the laws governing the animat's operation may rely upon various homeostatic mechanisms thanks to which, if the reference point alluded to earlier moves away from an adapted point of equilibrium -- adapted because it is suitably located within the viability zone --, this process tends to return it to its original position, thereby decreasing the risk that it will pass outside the limits of the zone. Other ways in which to lower this risk involve the use of high-quality sensory organs or motor apparatus that allow the animat to detect as early as possible that it is approaching these limits and/or to move away from them quickly and effectively. In this line of reasoning, it is obvious that the equivalent of a nervous system is mandatory in order to connect the animat's perceptions with its actions and that reflex circuits activated as quickly as possible increase the adaptive nature of its behavior. It is

likewise clear that additional adaptive potential is afforded to any animat capable of responding with more than simple reflexes, particularly when it is able to choose from among several possible reactions the one that proves best adapted to the situation at hand. Such a behavioral control can in particular result from changes in the animat's motivation brought on by this situation. Lastly, it is understandable that the capacity of memorizing the perception/action sequences that have shown themselves to be useful or harmful in the past is of considerable adaptive value for any animat obliged to decide what to do in a given situation, and that this adaptive value is enhanced should the animat also be capable of more or less sophisticated forms of planning.

The intent of this article is to review significant approaches that make use, separately or in combination, of these various adaptive solutions, whether they were proposed by ethologists, computer scientists or roboticists. We hope that such a variety of concepts and techniques will encourage a dialogue between specialists with different scientific perspectives and that, from an interdisciplinary approach of adaptive behavior, both a better understanding of animal behavior and the conception of really autonomous robots will arise. This review will be concerned with computer simulations and real robots and will deal with innate -- because they are programmed or wired in -- and acquired -- because they result from learning or evolution -- adaptive solutions. However, a simple inventory of these approaches will be made here, as more detailed descriptions thereof are given in Meyer and Guillot (1990).

2. PREPROGRAMMED BEHAVIORS

Many simulation models exhibit adaptive behaviors because they were purposely programmed that way. The most classical models belonging to this category are cybernetic models that use suitable feedback loops to maintain an animat's essential variables inside a specific region of its viability zone. Thus Booth's feeding behavior model (Booth, 1978) is founded on the hypothesis that the initiation and termination of feeding in rats depends on the instantaneous value of the energy flow entering or exiting from lean tissues. An hysteresis loop explicitly governs the margin of variation of this flow, and therefore the animal's hunger. Likewise, in the drinking behavior model proposed by Toates & Oatley (1970), another hysteresis loop restricts the variations of the thirst signal. Numerous other instances of this type of cybernetic model are to be found in McFarland (1971, 1974, 1985) and Toates (1980, 1986).

Work by various authors in the field of biomimetic signal processing (Busnel & Fish, 1980; Nachtigall & Moore, 1988; Rivamonte & Dral, 1990) show promise as to the

possibility of increasing the adaptive capacities of an animat by providing it with sensors that work more or less like the sensory organs of animals. For example, use has been made of knowledge gained about the compound eye of the fly (Franceschini *et al.*, 1989) in the development of an on-board electro-optical system for the guidance of an autonomous mobile robot (Pichon *et al.*, 1990). Likewise, Wang *et al.* (1990) obtained insights into the vision system of the rattlesnake which they then used to carry out fusion of multisensor data in a robot.

A markedly different approach, but one that is also expected to improve the efficiency of an animat's sensory-motor system, is the one developed by Agre and Chapman (Agre, 1988; Agre & Chapman, 1987; Chapman, 1989) involving *deitic representations*. The basic idea is that it is unnecessary to equip the animat with a sensory apparatus capable, at all times, of detecting and distinguishing between all the objects in its environment in order to ensure its adaptive competence. All that is required is that it be able to register information only about objects that are relevant to the problem to be solved. That is, at any moment, the animat's internal representation should register only the features of a few key objects and ignore the rest. Also, those objects should be indexed according to the intrinsic features and properties that make them significant. Such an approach gives rise to the design of systems that actively control their sensory apparatus, since they must mark relevant objects and change their focus of attention as new objects make their appearance in the environment and/or that the objects previously marked cease to be of interest.

The efficiency of this approach is illustrated by the *Pengi* system, which is capable of attaining remarkably high scores on a popular video game. In this game, a penguin must survive as long as possible in a world containing ice blocks and bees, the ice blocks being moveable so as to serve as protection against a bee, or to smash it.

Various approaches achieve an adaptive sensory-motor integration by means of devices more or less directly suggested by natural nervous systems and notably from knowledge gained in the area of *neuroethology* (Camhi, 1984; Ewert, 1980, 1987; Ewert *et al.*, 1983). These devices -- at least in their simulated version -- implement artificial neural networks similar, for the most part, to those made popular by the Parallel Distributed Processing Group (Rumelhart & McClelland, 1986).

As an example of the use of an artificial neural network for controlling behavior, Beer (1989) has developed a model enabling an artificial insect to display a variety of behaviors -- locomotion, wandering, edge following and feeding -- ensuring its survival in a simulated environment. Considering that the same neurons are involved in the initiation and control of these behaviors, it is evident that a

proper organization must preclude the simultaneous occurrence of incompatible acts, that is those calling upon the same motor units or the same *behavioral final common path*, to use the expression of McFarland & Sibly (1975). The solution retained is a hierarchical organization where the consumatory part of feeding behavior takes precedence over the orientation-towards-food behavior, which in turn is dominant with respect to the obstacle-avoidance behavior. According to such an organization, exploration is the behavior engaged in by default, while locomotion is activated in the course of every behavior entering into this hierarchy.

This type of organization, in which a dual perception/action hierarchy is implemented, is quite similar to those proposed by Tinbergen (1951) in a purely ethological context, or by Simon (1982) in a more general perspective. Moreover, its capacity to generate adaptive behaviors has been studied at length by Albus (1979, 1981).

Other examples of hierarchical control are afforded by Jacobs (1972) and by Meyer (1986). This latter work describes how the rat *Psikharpax* makes use of a symbolic model of its world so as to survive in a familiar environment. Notably, *Psikharpax* is capable, in order to avoid dying of hunger, of fetching the keys that open the various doors giving access to its food. More generally, this work addresses the way in which artificial intelligence and ethology may contribute mutual insight, particularly in the realm of heuristic problem-solving.

Various studies of robot navigation also rely on explicit models of the outer world, these robots generally being run by appropriate planning programs to determine what path should be followed to go from one point to another (Giralt *et al.*, 1984; Koch *et al.*, 1985; Lin *et al.*, 1989; Nilsson, 1984). To solve the problems that can arise from such an approach when an unforeseen obstacle hinders the normal operation of the plan, Arkin proposes the *AuRA* (Autonomous Robot Architecture) architecture that is based both on an *a priori* world model and on mechanisms allowing rapid and effective reaction to changes occurring within the world (Arkin, 1989a; Arkin & Taylor, 1990). In particular, this architecture has recourse to *motor schemas* (Arbib, 1989a; Arkin, 1987) which are the basic units of motor behavior in *AuRA*. In Arbib and House (1987) and in Arbib (1989b) other applications of such schemas are to be found concerning models of worm acquisition by toads and frogs in an obstacle-cluttered environment that entails detour behavior.

For Brooks (1987), the use of a world model is not essential to the generation of intelligent behavior, in particular when carrying out navigational tasks. This viewpoint is illustrated by the construction of ten or so autonomous robots of highly varied sizes and missions (Flynn & Brooks, 1988), but all characterized by the same

subsumption architecture (Brooks, 1986). This architecture represents a parallel and distributed computational formalism for connecting sensors to actuators in robots. Instead of fusing sensory data into an internal world model of the robot's environment and using this model to plan intelligent action, subsumption architecture provides an alternative way of writing intelligent control programs for robots in which sensors are more directly linked to action-suggestion modules. Fusion, here, happens at the actuator-plan level, rather than at the sensor-perception level.

One writes a subsumption program by specifying layers of networks of finite-state machines augmented with various timers and registers. For example, this architecture makes it possible for the robot *Genghis* to chase infrared sources over rough terrain (Brooks, 1989). Likewise, it permits *Squirt* -- "the world's largest one-cubic-inch robot" (Flynn *et al.*, 1989) -- to act as a "bug", hiding in dark corners and venturing out in the direction of noises only after the noises are long gone.

Another parallel and distributed structure is proposed by Maes (1989) to control the behavioral sequences of an animat. Instead of being governed by a hierarchy and according to a *top-down* design, the various acts making up the sequence follow each other in an order that is an emerging property displayed by a network of *competence modules*, according to a *bottom-up* design derived from Minsky's *Society of Mind* theory (Minsky, 1986). For a comparison of such an approach with more traditional studies about behavioral sequences in animals, one should consult Guillot (1986, 1988).

The idea of using different agents that cooperate to achieve tasks without specific coordination or communication is carried to its limit in the proposition of sending to the moon a colony of small robots for building a permanent base (Brooks & Flynn, 1989). An example is to be found in Brooks *et al.* (1990) of the simple behaviors that must be exhibited by each individual robot in order for the whole colony to be able, firstly, to select a promising area in which to construct the lunar base and, secondly, to level the ground in the construction site and pile up the soil in a small number of piles. Similar parallels between artificial self-organizing systems and insect societies are mentioned in Deneubourg & Goss (1989), Moyson & Manderick (1988) and Steels (1987, 1989).

3. LEARNED BEHAVIORS

Many simulation models implement individual learning processes by means of which an animal or a robot can improve the adaptiveness of their behaviors.

The simplest sort of learning thus used is *conditioning* that modifies the behavior of the animat as a result of certain temporal relationships between events. Many models describe how such conditioning could arise at the single neuron level (Barto & Sutton, 1982; Sutton & Barto, 1987). In Sutton & Barto (1989), for example, such a model -- called the *temporal difference model* -- is compared with animal learning theories of reinforcement from Mowrer (1960) and Rescorla & Wagner (1972).

Other models make use of several neurons to simulate conditioning experiments. *LIMAX* (Gelperin *et al.*, 1985) is an example of a model enabling one to simulate Pavlovian conditioning in a slug which learns to avoid plants containing bitter or toxic chemical substances. Under these conditions, the plant's odor constitutes the *conditioned stimulus* (CS) while the bitter flavor is the *unconditioned stimulus* (US).

Other varieties of learning can occur within artificial neural networks, as exemplified by the work of Grossberg and his school (Grossberg, 1987, 1988; Grossberg & Kuperstein, 1989; Grossberg & Schmajuk, 1989) and by many other simulations.

In the work of Pomerleau (1989), for example, a *supervised learning* procedure has been used for training a three-layer neural network to govern the movements of NAVLAB, the Carnegie Mellon autonomous vehicle. Likewise, an *unsupervised learning* procedure has been used by Salu (1984) to simulate the manner in which a young cat learns to behave in its environment and how the corresponding pertinent information is memorized and recalled.

Many models call upon a different variety of learning, known as *reinforcement learning*, that prompts the animat to discover for itself which actions it must perform in order to maximize an external gain, or *reinforcement* signal.

The Barto & Sutton model (1981) pertains to this category. It allows an animat to learn to use odorous landmarks to orient itself in a two-dimensional universe. In this universe, a central landmark is surrounded by four others, each of which emits an odor the strength of which decreases with distance. The purpose of the animat is to make use of these odors to reach the central landmark. This task is carried out by means of an *Associative Search Network* that is endowed with special neurons drawn from Klopff's theories (Klopff, 1980). The point to be emphasized in connection with this is that the animat works out, in the course of its learning phase, a kind of rudimentary *cognitive map*, and that this capacity is to be compared with various experiments that may imply the use of such maps by animals (Ellen & Thinus-Blanc, 1987; Gould, 1986; Griffin, 1982; Menzel, 1978; O'Keefe & Nadel, 1978; Pearce, 1987; Roitblat, 1987; Tolman, 1948;). Two models, for example, are to be found in O'Keefe (1989) and in McNaughton (1989) that describe how spatial data might

be memorized and used by the neuronal structures of the hippocampus.

In line with this notion of cognitive map, a number of papers describe adaptive networks of neuron-like elements that use and dynamically adjust an internal world model. Sutton & Barto (1981), for example, describe a simulation effecting a kind of *latent learning* in a network. Likewise, Barto *et al.* (1983) demonstrate how a single element can be used to model the reinforcement of the environment. Such a model makes it possible then to facilitate learning under conditions where the environment affords only an intermittent reinforcement signal. When this signal occurs after the animat has several successive actions, a fundamental *credit assignment* problem arises (Minsky, 1963): that of deciding what portion of the corresponding reward or punishment is to be assigned to each of these actions. Obviously, an elegant solution to this problem is afforded if a model can be used that will predict what reinforcement the environment could have furnished in response to each action.

The possibility for an animat to have access to a world model that allows it to carry out fictitious experiments and to plan its behavior through trial-and-error learning has, for example, been examined by Sutton (1990). Thus this author has studied a class of simple architectures called *DYNA* that apply various machine learning algorithms to approximate conventional dynamic programming control.

A *DYNA* architecture relies essentially upon four interacting components:

- The real world, that changes state in relation with the animat's behavior and that distributes rewards and punishments;
- The world model that the animat elaborates for itself and that is intended to mimic the one-step input/output behavior of the real world;
- The policy function relied on by the animat to determine what action to engage in response to each possible state of the world;
- The evaluation function to which the animat has recourse in order to assess the reward associated with each state, when its object is to maximize the long-term average reward per time step.

The world model, the policy function, and the evaluation function are progressively modified as the animat experiments with the operational laws of its world. These modifications depend upon two type of experiments that the animat may alternate between: actual experiments carried out on the real world -- that bring about updates by *temporal difference reinforcement learning* -- and fictitious experiments -- that make use of the model world and give rise to updates by *relaxation planning*. In either case, the algorithm applied is related to dynamic programming

inasmuch as the credit is propagated from one step to the preceding one within action sequences.

Simulations effected on different variants of this architecture indicate that the corresponding animats are not only capable of learning how to move around in a labyrinth while minimizing their navigational errors by simple trial-and-error learning, but further that this learning is expedited when they also avail themselves of the planning possibilities afforded by their world models. What is more, certain of these architectures are easy to adapt for use in changing environments.

A similar approach has been put into practice by Thrun *et al.* (1990) on an adaptive connectionist planning method. In this work, an animat's world model is progressively elaborated in the light of interactions between this animat and its environment, and this world model is used to work out a look-ahead plan expected to maximize future reinforcements. A special *back-propagation* method has been conceived for the purpose, which effects a gradient descent in the space of all possible actions.

Another approach conceived in this line of thought is that of Schmidhuber (1990). In this work, two neural networks are used to govern the actions of an animat, with one of the networks being responsible for the sensory-motor integration, while the other models environmental changes and reinforcements resulting from the animat's actions.

Possibilities of learning through reinforcement as well as of planification are exhibited by the *AGAR* system (Travers, 1989), the architecture of which is not directly inspired by a neural network, but rather is based upon Minsky's *Society of Mind* theory cited earlier. In this system, behavior control is carried out by entities called *agents*, comparable to production rules in that they are characterized by conditions and actions. The conditions required to activate an agent may depend on the state of various sensors, as they may also depend on the activity of other agents. The actions arising from an activation can be any behaviors whatsoever, as long as they are capable of being expressed in a high level programming language such as LISP. They may, among other things, activate or suppress other agents, activate a motor function, activate a *script* (Schank & Abelson, 1977), memorize the activation status of other agents, or create or modify agents. For example, a set of instructions is to be found in the paper of Travers that makes it possible to create the agents necessary to simulate the courting behavior of the stickleback.

The works that have just been mentioned are all based on the same implicit and simplifying hypothesis that the animat's sensory equipment allows it to elaborate a model of its world the states of which correspond unequivocally with those of the real world. Such an hypothesis is first of all more or less unrealistic and secondly results in models that are generally too complex to be usable in practice, save

for certain simple academic cases. This is why Whitehead & Ballard (1990) have studied an adaptive control architecture that integrates active sensory-motor systems -- based on the *deitic representation paradigm* referred to earlier -- with decision systems based on reinforcement learning. Such an architecture learns not only how to solve a task like block manipulation, but also where to focus its attention in order to collect the necessary sensory information. In particular, it makes it possible to solve the problem of *perceptual aliasing* generated by the active perception mechanisms thus brought into play, problem that occurs because the animat's internal representation often confuses external world states with one another.

Reeke and Edelman's *Darwin III* model (Reeke and Edelman, 1988) likewise represents the mind as an unsupervised learner in which much of the clustering of stimuli into classes is not readily inherent in the stimuli and must be performed by the individual according to what is adaptive for its species and its own particular circumstances. Furthermore, the categories constructed by an organism aren't fixed, but constantly change in response to new experiences on its part of the environment. Thus, to carry out tasks involving categorization, recognition, generalization, and association with respect to moving objects, *Darwin III* uses a multi-jointed arm and a moveable head with one or two eyes to refine its perceptions by means of simple motor actions, such as saccades and visual tracking, reaching, touching, and tracing. The corresponding architecture is a direct implementation of Edelman's *neuronal group selection* theory (Edelman, 1987), whereby the individual's behavioral repertoire arises out of a selection among diverse pre-existing groups of cells -- accomplished by differential modification of synaptic strengths -- according to what is of adaptive value to it in its niche. This selection is elaborated without change in the connectivity pattern within and between cell groups, which has been fixed during the prenatal period of the brain's development.

Another organization, called a *classifier system*, was elaborated by Holland (1971) so that individual learning could be carried out. Many descriptions of this organization have been made, notably in Booker *et al.* (1989), Goldberg (1989), Grefenstette (1985, 1987), Holland (1986), Holland *et al.* (1986) and Schaffer (1989).

However, because a classifier system is generally applied in conjunction with a genetic algorithm, specific simulations that implement such a system will be mentioned in the following section only. It should however be noted that a critical review of the way in which classifier systems function is given in Wilson & Goldberg (1989) and that changes in the basic structure of these systems are proposed by Wilson (1987b) in the interest of achieving a better control of behavioral sequences by a hierarchical structure. Such a structure, in addition to the similarities it presents

with the hierarchical control models used by ethologists (Baerends *et al.*, 1970; Dawkins, 1976; Tinbergen, 1951), should materially accelerate the learning. Let us also mention that the work of Holland *et al.* (1986) interprets in terms of classifier systems the results of many classical experiments in animal conditioning and learning.

4. EVOLVED BEHAVIORS

The idea that an animal's behavior is partially determined by its genome and that this behavior is accordingly capable of evolving through natural selection implicitly underlies many optimization models proposed by ethologists and ecologists (Alexander, 1982; Clark, 1990; Krebs & McClery, 1984; Lendrem, 1986; Mangel & Clark, 1988; McFarland, 1982; Meyer, 1980), in the framework of what is known as a *functional analysis* of behavior. This type of analysis postulates that the behavior currently exhibited by an animal stems from successive improvements it has benefited in the past, from generation to generation. Provided the process has been going on a sufficiently long time, the behavior's selective value can be expected to have reached its maximum value or, in other words, to be *optimized*. In this context, the selective value of a behavior is bound up with its adaptive nature. Indeed, it estimates the extent to which this behavior influences the animal's capacity to produce a large number of fertile descendents, and this capacity obviously requires not only that the animal be able to keep itself within its viability zone, but further that it remain in a more restricted zone compatible with its reproductive needs.

In these perspectives, a static optimization model was, for example, proposed by Belovsky (1978) for predicting food selection in the moose. Likewise, a dynamic optimization model was proposed by McFarland (1977) to predict the optimum order of occurrence of three stereotyped acts in the courting behavior of the male newt: retreat displays, pursuit motions, and spermatophore transfers.

Other dynamic optimization models of behavior make use of dynamic programming techniques. This, for instance, is the case with McNamara & Houston's model (1986) which describes the relationships between short-term behavior and lifetime reproductive success for a bird, with Noller's model (1988) which describes the queen's egg-laying cycle in bees and social wasps, with Guillot's model (1988) which describes ultradian activity cycles in mice, and with Mangel & Clark's model (1988) which describes the social hunting behavior of lions.

The optimization models just mentioned all call for just one decider. The extension of this approach to the case of two or more deciders entails, from a technical point of view, using of models that are formulated in terms of game theory

and, from a theoretical standpoint, calling upon the important concept of *evolutionarily stable strategy* (Maynard-Smith, 1974).

This notion concerns a strategy which is optimized -- in the sense that it cannot be replaced by another more efficient -- and which has proven particularly fruitful in the area of behavioral ecology (Colgan, 1989; Maynard-Smith, 1982; McFarland, 1985; Parker, 1984, 1985). Houston & McNamara (1987) or Mangel & Clark (1988) are to be consulted for its extension to dynamic models.

If these optimization models make it possible to test the hypothesis under which a given animal's behavior is the result of a long evolutionary process -- in other words that, if it had been different, it would have been eliminated by natural selection -- it is clear that these models do not purport to explain by what means this behavior evolved. In particular, these models are unable to handle genotype-phenotype relationships that are instrumental in the evolutionary process thus hypothesized (Jamieson, 1986).

In this connection, the approach favored by Holland (1975) -- in search of an adaptive plan making it possible to improve the adaptation of any given system -- appears to be entirely complementary with respect to the preceding approaches, for it enables optimization problems to be solved by using explicit genotype-phenotype relationships. More specifically, this approach calls for a *genetic algorithm* which consists in the parallel management of a population of chromosomes such that each one of them codes -- generally in the form of a chain of binary symbols -- a possible solution to a particular optimization problem. Each of these chromosomes can therefore be assigned a fitness that assesses the corresponding solution. The application of the genetic algorithm accordingly consists in causing this population to evolve from generation to generation while rendering the probability of reproduction of each chromosome proportional to its fitness and using genetic operators such as mutations and crossing-over to give rise to new solutions in the population. Under these circumstances, this type of evolutionary process causes chromosomes of ever-increasing fitness to be generated until the optimum value is reached, or sufficiently nearly so for all practical purposes.

Many applications of algorithms of this kind are to be found in Goldberg (1989), Grefenstette (1985, 1987), or Schaffer (1989) and a critical review is available in Schaffer & Grefenstette (1990). A special application, designed for the study of the behavior of a population of interactive animats, has been elaborated by Sannier & Goodman (1987). In particular, this work shows that it is possible to evolve, through hybridization, a composite genome capable of producing and organizing a distributed system of independent animats which work towards a common goal. The model of Packard (1989) is also intended to simulate

the evolution of animats in interaction in the framework of a spatial ecosystem. In this model, each animat's genome contains two genes, one coding the food threshold required for reproduction, and the other coding the number of offsprings. The results obtained indicate that the dynamic characteristics of the evolution of the simulated population strongly depend on the mode of driving the system with food, and notably on the size of the corresponding fluctuations.

Other applications of genetic algorithms, notably to the study of animat behavior, are presented in Koza (1990). In particular, this text describes how to generate a plan allowing an artificial ant to traverse a trail with gaps of various types. It also shows how to find the minimax strategy for the pursuer to catch an evader in the differential game of simple pursuit played on the plan.

Still other applications of genetic algorithms make it possible to simulate various interesting *co-evolution* phenomena (Axelrod, 1987; Hillis, 1990; Holland, 1990; Miller, 1989).

Likewise in the work of Viola (Viola, 1988) is to be found an original utilization of a genetic algorithm so as to cause a modification of the internal architecture of a robot to take place.

Lastly, let us mention the work carried out by Bertin (1990) who studies the evolution of behavior in aquatic animats called *paddlers*, that must seek out and feed on other animats called *glowballs*. Each paddler's behavior depends on a neural network the purpose of which is to translate the visual inputs affecting the animat into suitable navigational responses, in this case appropriate paddle frequencies. The architecture of these networks is invariable, but the behavior they give rise to is strongly dependent on a variety of numerical parameters characterizing each animat's visual system. The evolutionary experiments deal with these parameters, that undergo random mutations from generation to generation. However, in contrast with the procedures called on in a classical genetic algorithm, these experiments do not involve any crossing-over and therefore presuppose simple asexual reproduction.

The results obtained do not manifest any significant improvement of the animats' performances with time, due to the purely random exploration of the space of the parameters implied by these experiments. This point is obviously related to the secondary role these mutations would seem to play in the evolutionary process (Holland, 1975) and to the limited results achieved by other asexual mutation experiments that attempt to solve the problems of adaptive systems by merely "saving and mutating the best" (Dawkins, 1986; Fogel *et al.*, 1966; Hicklin, 1986).

As was mentioned previously, a classifier system is generally used in conjunction with a genetic algorithm. If

an equivalent to this practice is to be sought in nature, it may be considered that -- within the so-called "Michigan approach" -- the genetic algorithm is used to recombine thoughts in a given mental model, whereas -- within the "Pittsburgh approach" -- its purpose is rather to simulate the evolution of a behavioral program coded in a genome.

Whatever the case, the first application of a classifier system to the simulation of an animat's behavior was that of Holland & Reitman (1978). In this work, an animat learns to solve a double survival problem: avoid dying of hunger or thirst. To do this, it must move around in a mono-dimensional universe containing food at one of its extremities and drink at the other. However, due to the fact that its hunger and thirst evolve at different rates, it must learn to adjust accordingly the frequency of its visits to the spots where food and drink are distributed.

The animat simulated by Wilson (1985) also makes use of a classifier system to find food and avoid obstacles on a two-dimensional grid.

Another classifier system has been developed by Booker (1982, 1988) to allow an animat called *GOFER* to learn to produce goal-seeking behavioral sequences through the use of rules that manipulate objects, goals, and object/goal associations, rather than rules that code simple stimulus/response associations. From this point of view, this work specifically calls upon a *cognitive* approach to behavior, as opposed to a *behaviorist* one (Bower & Hilgard, 1981).

In closing this review, let us mention the work of Ackley & Litman (1990) that explicitly combines the two natural processes by means of which behavior is optimized for survival: *learning* -- as the adaptation of the individual -- and *evolution* -- as the adaptation of a population. This work draws upon a strategy called *evolutionary reinforcement learning (ERL)* that makes use of genetic evolution techniques to allow effective learning based solely upon natural selection.

The corresponding simulations concern an animat population that struggles for existence in a simulated world where the only feedback mechanism is natural selection. The fate of these animats in their environment depends on their capacity to learn to find food, to avoid obstacles, and to escape from their predators. Their behavior is governed by a neural network and a learning process which relies on a special reinforcement algorithm. The network and the evaluation function used by the algorithm are coded in these animats' genome and accordingly may evolve as the simulation progresses.

Under these conditions, the *ERL* strategy displays better performance -- measured in terms of mean population survival times -- than control populations of randomly-moving animats, or than populations that employ just

evolution or just individual learning. This result is obviously to be borne in mind when considering the various recent studies addressing the *Baldwin effect* (Belew, 1989; Duchateau *et al.*, 1990; Hinton & Nolan, 1987; Maynard-Smith, 1987; Nolfi *et al.*, 1990), or, more generally, the conjunction of a genetic algorithm and a neural network (Caudell & Dolan, 1989; Harp *et al.*, 1989; Miller *et al.*, 1989; Resnick, 1990; Todd, 1988; Whitley & Hanson, 1989).

5. OVERVIEW AND PROSPECT

The first remark that comes to mind at the end of this review is that the majority of the works alluded to were performed in the course of the past five years. It is therefore quite clear that the study of adaptive behaviors exhibited by simulated animals or autonomous robots is a theme of current concern. This sudden popularity can doubtless be explained by the fact that the subject promises to be fertile in theoretical innovations -- notably as concerns the relationships between individual learning and species evolution -- and practical accomplishments -- particularly as regards the construction of robots capable of carrying out various custodial services or exploratory tasks.

Furthermore, it so happens that natural animals afford the principle proof that adaptive behaviors -- certain of which may be extraordinarily efficient -- can be elicited by systems that have been properly organized, and that a great deal of knowledge has been accumulated by ethologists since they began to take an interest in the scientific study of animal behavior (Richard, 1984, 1985).

From this new-found interest in artificial animals or from this longstanding knowledge of natural animals, it can reasonably be hoped a growing interest in an interdisciplinary approach to adaptive behaviors.

It is probable, for instance, that ethologists will profit to a greater extent from the possibilities offered by simulation models. One of the advantages afforded by these models (Meyer & Guillot, 1986a) is indeed that their functioning laws are set forth in the corresponding programs in perfectly unambiguous form, whatever the scientific or philosophical biases of the one observing the behavior of these programs. Under these conditions, it can be expected that controversies as to the role of possible cognitive processes in the determinism of animal behavior (Colgan, 1989; Griffin, 1984; Hoage & Goldman, 1986) will soon no longer be hampered by mere statements of principle and ambiguous terminology, in which they have so often become mired down in the past. The availability of simulation programs makes it possible to identify which instructions devolve from a purely ethological interpretation and which from a cognitive one. An analysis of the

program's sensitivity to these instructions and, more generally, a comparison of the program's behavior with that actually displayed by the animal should enable the validity of the corresponding interpretations to be assessed.

We are convinced that most of these controversies would automatically work themselves out as soon as interdisciplinary approaches will begin to prevail. For example, while Booker (1982, 1988) explicitly places his work in a cognitive perspective, our belief is that nothing in *GOFER's* program mentioned earlier is in contradiction with the classical concepts of ethology. Likewise, the expressions "world model" or "planning" used to describe a program's behavior often implement data structures and procedures that are probably less at variance with certain ethologists' concepts than these latter may at first be inclined to believe. After all, starting with Walter (1950, 1951) or Wiener (1961), it has long been accepted that very simple mechanisms can generate many "goal-directed" behaviors (Ackoff & Emery, 1972; Weir, 1984).

Naturally this discussion is not unrelated with those devoted to symbolic and sub-symbolic approaches to artificial intelligence (Fodor & Pylyshyn, 1988; Lloyd, 1989; Smolensky, 1988; Waltz, 1988), nor is it unconnected with the controversy that divides the proponents of reactive control from those of knowledge-based planning in the area of robot navigation (Arkin, 1989b). In all these respects, the animat approach should eventually prove enlightening, if only through the experimentation it allows and the opportunity it affords to ascertain which model or which minimal architecture results in what type of adaptive behavior being exhibited.

It nevertheless appears to us that these advancements would be all the more explicit and useful if the aforesaid interdisciplinary approaches were to center around a minimal number of anchor points, the groundwork of which has yet to be laid.

In view of the extreme diversity of the approaches reviewed in this text, it indeed seems essential to seek a typology of the problems dealt with, on the one hand, and a typology of the proposed solutions, on the other hand, in order to find out -- among other things -- whether or not simple correspondances exist between the former and the latter. In this perspective -- which is closely allied with Wilson's proposal (1990) for a theory of environments -- it would certainly be interesting to discover that one specific type of problem is analogous to another specific type for such and such a reason and that, accordingly, the said problem stands a good chance of being solved by a specific type of solution or architecture rather than by another one.

However, for it to be likely that such knowledge will be available some day, current practice will definitely have to be superseded in which each published article is just embodying a proof of principle. Besides the fact that it is

never established that the corresponding solution is minimal, nor that any given adaptive capacity is expressly due to a specific global architecture rather than to a particular operational detail, the limits of expression of these capacities are rarely explored.

Nothing short of a systematic comparison of several different implementations of a same type of solution with a range of problems as wide as possible is liable to reveal the generic properties of the solution considered. Conversely, only a systematic comparison of several different versions of the same problem with a range of solutions as varied as possible can allow an evaluation of the respective advantages and the degree of originality of these solutions.

This last type of knowledge is, in particular, the one that would prove the most useful to ethologists for interpreting a given animal's behavior as mobilizing the same solution as that which enables an animat to solve the same type of adaptive problem. Among the very large assortment of solutions that might on first thought be possible, obviously those that have already demonstrated their usefulness in similar fields of application should be tried out first. However, as suggested by Braitenberg (1984) when alluding to the "law of uphill analysis and downhill synthesis", it is bound to be more difficult for the ethologist than for the roboticist or the modeller to derive benefit from knowledge gained from animats. This will be true as long as the former is unable to be certain of the uniqueness of the solutions that are to be envisaged and as the latter's problem is simply to generate an *ad hoc* solution without attaching any particular importance to the manner in which it is brought about.

It is nevertheless clear that the times are rapidly changing and it is quite probable that the robotic solutions considered will not long remain equivalent. In particular, one may foresee that costs will be associated with robot behaviors as they are with animal ones (Meyer & Guillot, 1986b) and that the success of certain architectures will depend not only on such quantitative costs (McFarland, 1990) but also on more subtle and qualitative factors like the nature of the various emotions arising within such architectures (Albus, 1981; Dyer, 1987; Frijda & Swagerman, 1987; Gehm, 1989; Sloman & Croucher, 1981).

ACKNOWLEDGMENTS

The authors are greatly indebted to Stewart Wilson and Jody Mohammadioun for their valuable help in the conception and realization of this paper and wish to express to them their extreme gratitude.

They also want to thank all the people who helped provide this review by sending papers or technical reports. Corrections and additional references will be much appreciated.

REFERENCES

- Ackley, D.H. & Littman, M.S. (1990). Learning from natural selection in an artificial environment. Proceedings of The Second Artificial Life Conference. Santa Fe.
- Ackoff, R.L. & Emery, F.E. (1972). On purposeful systems. Aldine.
- Agre, P.E. (1988). The dynamic structure of everyday life. Technical Report (1085). MIT Artificial Intelligence Lab.
- Agre, P.E. & Chapman, D. (1987). Pengi: an implementation of a theory of activity. Proceedings of The Sixth National Conference on Artificial Intelligence. Seattle.
- Albus, J.S. (1979). Mechanisms of planning and problem solving in the brain. *Mathematical Biosciences*. 45: 247-293.
- Albus, J.S. (1981). Brains, behavior and robotics. Byte Books.
- Arbib, M.A. (1989a). The metaphorical brain 2. Neural networks and beyond. Wiley.
- Arbib, M.A. (1989b). Interacting subsystems for depth perception and detour behavior. In Arbib & Amari (Ed.), *Dynamic interactions in neural networks: models and data*. Springer Verlag.
- Arbib, M.A. & House, D. (1987). Depth and detours: an essay on visually guided behavior. In Arbib & Hanson (Ed.), *Vision, brain and cooperative communication*. The MIT Press.
- Arkin, R.C. (1987). Motor schema based navigation for a mobile robot: an approach to programming by behavior. Proceedings of The IEEE Conference on Robotics and Automation. Raleigh.
- Arkin, R.C. (1989a). Integrating behavioral, perceptual, and world knowledge in reactive navigation. To appear in *Robotics and Autonomous systems*. Elsevier/North-Holland.
- Arkin, R.C. (1989b). Towards the unification of navigational planning and reactive control. Proceedings of The AAAI Spring Symposium on Robot navigation. Stanford.
- Arkin, R.C. & Taylor, D.T. (1990). Reactive behavioral support for qualitative visual navigation. Proceedings of The IEEE Conference on Intelligent Motion Control. Istanbul.
- Ashby, W.R. (1952). Design for a brain. Chapman & Hall.
- Axelrod, R. (1987). The evolution of strategies in the iterated prisoner's dilemma. In (Davis, Ed.), *genetic algorithms and simulated annealing*. Pitman.
- Baerends, G.P., Drent, R.H., Glas, P. & Groenewold, H. (1970). An ethological analysis of incubation behaviour in the herring gull. *Behav. Suppl.* 17: 135-235.
- Barto, A.G. & Sutton, R.S. (1981). Landmark learning: an illustration of associative search. *Biological Cybernetics*. 42: 1-8.
- Barto, A.G. & Sutton, R.S. (1982). Simulation of anticipatory responses in classical conditioning. *Behavioural Brain Research*. 4: 221-235.

- Barto, A.G., Sutton, R.S. & Anderson, C.W. (1983). Neuronlike elements that can solve difficult learning control problems. *IEEE Trans. on Systems, Man and Cybernetics*. SMC-13(5): 834-846.
- Beer, R.D. (1989). Intelligence as adaptive behavior: an experiment in computational neuroethology. Technical Report (TR 89-118). Center for Automation and Intelligent Systems Research. Case Western Reserve University.
- Belew, R.K. (1989). Evolution, learning and culture: computational metaphors for adaptive algorithms. Technical Report (CSE TR #CS89-156). Computer Science and Engineering. Univ. of California.
- Belovsky, G.E. (1978). Diet optimization in a generalist herbivore: the moose. *Theor. Pop. Biol.* 14: 105-134.
- Bertin, R.J. (1990). Experiments with simulated eyes, brains and hypothetical animals. Of Paddlers and Glowballs. Unpublished manuscript. Vakgroep Vergelijkende Fysiologie. Rijksuniversiteit te Utrecht.
- Booker, L.B. (1982). Intelligent behavior as an adaptation to the task environment. PhD Thesis. Dept. of Computer and Communication Sciences. Univ. of Michigan.
- Booker, L.B. (1988). Classifier systems that learn internal world models. *Machine learning*. 3: 161-192.
- Booker, L.B., Goldberg, D.E. & Holland, J.H. (1989). Classifier systems and genetic algorithms. *Artificial Intelligence*. 40: 235-282.
- Booth, D.A. (1978). Prediction of feeding behavior from energy flows in the rat. In Booth (Ed.), *Hunger models: computable theory of feeding control*. Academic Press.
- Bower, G.H. & Hilgard, E.R. (1981). *Theories of learning*. Prentice-Hall.
- Braitenberg, V. (1984). *Vehicles. Experiments in synthetic psychology*. The MIT Press.
- Brooks, R.A. (1986). A robust layered control system for a mobile robot. *IEEE Journal of robotics and Automation*. RA-2(1): 14-23.
- Brooks, R.A. (1987). Intelligence without representations. *Proceedings of The workshop on Foundations of Intelligence*. MIT. Endicott House.
- Brooks, R.A. (1989). A robot that walks; emergent behaviors from a carefully evolved network. *Neural Computation*. 1: 253-262.
- Brooks, R.A. & Flynn, A.M. (1989). Fast, cheap and out of control. *Journal of the British Interplanetary Society*. 42(10): 478-485.
- Brooks, R.A., Maes, P., Mataric, M. & More, G. (1990). Lunar base construction robots. To appear in IROS-90, Japan.
- Busnel, R.G. & Fish, J.F. (1980). *Animal sonar systems*. Plenum.
- Camhi, J.M. (1984). *Neuroethology: nerve cells and the natural behavior of animals*. Sinauer Assoc.
- Caudell, T.P. & Dolan, C.P. (1989). Parametric connectivity: training of constrained networks using genetic algorithms. In Schaffer (Ed.), *Proceedings of The Third International Conference on Genetic Algorithms*. Morgan Kaufmann.
- Chapman, D. (1989). Penguins can make cake. *AI magazine*. 10(4): 45-50.
- Clark, C.W. (1990). Modeling behavioral adaptations. To appear in *Behav. Brain Sci.*
- Colgan, P. (1989). *Animal motivation*. Chapman & Hall.
- Dawkins, R. (1976). Hierarchical organization: a candidate principle for ethology. In Bateson & Hinde (Ed.), *Growing points in ethology*. Cambridge Univ. Press.
- Dawkins, R. (1986). *The blind watchmaker*. Longman.
- Deneubourg, J.L. & Goss, S. (1989). Collective patterns and decision-making. *Ethology, Ecology and Evolution*. 1: 295-311.
- Duchateau, G., Meyer, J.A. & Tarrux, P. (1990). Life histories and evolution of primitive organisms. *Proceedings of The Second Artificial Life Conference*. Santa Fe.
- Dyer, M.G. (1987). Emotions and their computations: three computer models. *Cognition and emotion*. 1(3): 323-347.
- Edelman, G.M. (1987). *Neural Darwinism: the theory of neuronal group selection*. Basic Books.
- Ellen, P. & Thinus-Blanc, C. (1987). *Cognitive Processes and spatial orientation in animal and man*. Martinus Nijhoff.
- Ewert, J.P. (1980). *Neuroethology*. Springer Verlag.
- Ewert, J.P. (1987). Neuroethology of releasing mechanisms: prey-catching in toads. *Behavioral and Brain Sciences*. 10: 337-368.
- Ewert, J.P., Capranica, R.R. & Ingle, D.J. (1983). *Advances in vertebrate neuroethology*. Plenum.
- Flynn, A.M. & Brooks, R.A. (1988). MIT mobile robot: what's next? *Proceedings of IEEE Robotics and Automation*. Philadelphia.
- Flynn, A.M., Brooks, R.A., Wells, W.M. & Barrett, D.S. (1989). Intelligence for miniature robots. *Sensors and actuators*. 20: 187-196.
- Fodor, J.A. & Pylyshyn, Z. (1988). Connectionism and cognitive architecture: a critical analysis. *Cognition*. 28: 3-71.
- Fogel, L.J., Owens, A.J. & Walsh, M.J. (1966). *Artificial Intelligence through simulated evolution*. Wiley.
- Franceschini, N., Riehle, A. & Le Nestour, A. (1989). Directionally selective motion detection by insect neurons. In Stavenga & Hardie (Ed.), *Facets of vision*. Springer Verlag.
- Frijda, N.H. & Swagerman, J. (1987). Can computers feel? Theory and design of an emotional system. *Cognition and emotion*. 1(3): 235-257.
- Gehm, T.L. (1989). Neural networks and research on emotions: a mutual perspective. Unpublished manuscript. Department of Psychology. University of Giessen.
- Gelperin, A., Hopfield, J.J. & Tank, D.W. (1985). The logic of Limax learning. In Selverston (Ed.), *Model neural networks and behavior*. Plenum.
- Giralt, G., Chatila, R. & Vaisset, M. (1984). An integrated navigation and motion control system for autonomous multisensory mobile robots. In Brady & Paul (Ed.), *Proceedings of Robotics Research, The First International Symposium*. The MIT Press.
- Goldberg, D.E. (1989). *Genetic algorithms in search, optimization, and machine learning*. Addison Wesley.
- Gould, J.L. (1986). The locale map of honey bees: do insects have a cognitive map? *Science*. 232: 861-863.

- Grefenstette, J.J. (1985). Proceedings of the First International Conference on Genetic Algorithms and their Applications. Lawrence Erlbaum Assoc.
- Grefenstette, J.J. (1987). Genetic algorithms and their applications. Proceedings of The Second International Conference on Genetic Algorithms. Lawrence Erlbaum Assoc.
- Griffin, D.R. (1982). Animal mind - Human mind. Springer-Verlag.
- Griffin, D.R. (1984). Animal thinking. Harvard University Press.
- Grossberg, S. (1987). The adaptive brain I & II. Elsevier/North-Holland.
- Grossberg, S. (1988). Neural networks and natural intelligence. The MIT Press/Bradford Books.
- Grossberg, S. & Kuperstein, M. (1989). Neural dynamics of adaptive sensory-motor control. Pergamon Press.
- Grossberg, S. & Schmajuk, N.A. (1989). Neural dynamics of adaptive timing and temporal discrimination during associative learning. Neural networks. 2: 79-102.
- Guillot, A. (1986). Revue générale des méthodes d'étude des séquences comportementales. Etudes et analyses comportementales. 2(3): 86-106.
- Guillot, A. (1988). Contribution à l'étude des séquences comportementales de la souris: approches descriptive, causale et fonctionnelle. Doctorat Thesis. Biomathématiques. Université Paris 7.
- Harp, S.A., Samad, T. & Guha, A. (1989). Towards the genetic synthesis of neural networks. In Schaffer (Ed.), Proceedings of The Third International Conference on Genetic Algorithms. Morgan Kaufmann.
- Hicklin, J.F. (1986). Applications of the genetic algorithms to automatic program generation. Master of Science Thesis. Dept. of Computer Science. Univ. of Idaho.
- Hillis, W.D. (1990). Co-evolving parasites improve simulated evolution as an optimization procedure. Proceedings of The Second Artificial Life Conference. Santa Fe.
- Hinton, G.E. & Nowlan, S.J. (1987). How learning can guide evolution. Complex systems. 1: 495-502.
- Hoage, R.J. & Goldman, L. (1986). Animal Intelligence. Insights into the animal mind. Smithsonian Institution Press.
- Holland, J.H. (1971). Processing and processors for schemata. In Jacks (Ed.), Associative information processing. Elsevier.
- Holland, J.H. (1975). Adaptation in natural and artificial systems. Univ. Michigan Press.
- Holland, J.H. (1986). Escaping brittleness: the possibilities of general purpose machine learning algorithms applied to parallel rule-based systems. In Michalski, Carbonell & Mitchell (Ed.), Machine learning: an artificial intelligence approach. Volume 2. Kaufmann.
- Holland, J.H. (1990). ECHO: explorations of evolution in a miniature world. Proceedings of The Second Artificial Life Conference. Santa Fe.
- Holland, J.H., Holyoak, K.J., Nisbett, R.E. & Thagard, P.R. (1986). Induction. Processes of inference, learning, and discovery. The MIT Press.
- Holland, J.H. & Reitman, J.S. (1978). Cognitive systems based on adaptive algorithms. In Waterman & Hayes-Roth (Ed.), Pattern-directed inference systems. Academic Press.
- Houston, A.I. & McNamara, J.M. (1987). Singing to attract a mate - a stochastic dynamic game. J. Theor. Biol. 129: 57-68.
- Jacobs, W. (1972). How a bug's mind works. In Robinson & Knight (Ed.), Cybernetics, artificial intelligence, and ecology. Spartan Books.
- Jamieson, I.G. (1986). The functional approach to behaviour: is it useful? Amer. Natur. 127(2): 195-208.
- Klopf, A.H. (1980). The hedonistic neuron: a theory of memory, learning and intelligence. Hemisphere.
- Koch, E., Yeh, C., Hillel, G., Meystel, A. & Isik, C. (1985). Simulation of path planning for a system with vision and map updating. Proceedings of IEEE International Conference Robotics and Automation. St. Louis.
- Koza, J.R. (1990). Genetic programming: a paradigm for genetically breeding populations of computer programs to solve problems. Technical Report (STAN - CS - 90 - 1314). Dept. of Computer Science. Stanford Univ.
- Krebs, J.R. & McCleery, R.H. (1984). Optimization in behavioural ecology. In Krebs & Davies (Ed.), Behavioural ecology: an evolutionary approach. Blackwell.
- Lendrem, D. (1986). Modelling in behavioural ecology: an introductory text. Timber Press.
- Lin, L.J., Mitchell, T.M., Philips, A.R. & Simmons, R. (1989). A case study in robot exploration. Technical Report (CMU-RI-TR-89-1). Carnegie Mellon University.
- Lloyd, D. (1989). Simple minds. The MIT Press/Bradford Books.
- Maes, P. (1989). How to do the right thing. Connection Science. 1(3): 291-323.
- Mangel, M. & Clark, C.W. (1988). Dynamic modeling in behavioral ecology. Princeton Univ. Press.
- Maynard Smith, J. (1974). The theory of games and the evolution of animal conflicts. J. Theor. Biol. 47: 209-221.
- Maynard Smith, J. (1982). Evolution and the theory of games. Cambridge Univ. Press.
- Maynard Smith, J. (1987). When learning guides evolution. Nature. 329: 761-762.
- McFarland, D.J. (1971). Feedback mechanisms in animal behaviour. Academic Press.
- McFarland, D.J. (1974). Motivational control systems analysis. Academic Press.
- McFarland, D.J. (1977). Decision making in animals. Nature. 269: 15-21.
- McFarland, D.J. (1982). Functional ontogeny. Pitman.
- McFarland, D.J. (1985). Animal behaviour. Psychobiology, ethology and evolution. Longman Scientific & Technical.
- McFarland, D.J. (1990). What it means for robot behaviour to be adaptive (this volume).
- McFarland, D.J. & Sibly, R.M. (1975). The behavioral final common path. Phil. Trans. Roy. Soc. London. 270: 265-293.

- McNamara, J.M. & Houston, A.I. (1986). The common currency for behavioural decisions. *Amer. Natur.* 127: 358-378.
- McNaughton, B.L. (1989). Neuronal mechanisms for spatial computation and information storage. In Nadel, Cooper, Culicover & Harnish (Ed.), *Neural connections, mental computations*. The MIT Press/Bradford Books.
- Menzel, E.W. (1978). Cognitive mapping in chimpanzees. In Hulse, Fowler & Honig (Ed.), *Cognitive processes in animal behavior*. Lawrence Erlbaum Assoc.
- Meyer, J.A. (1980). Techniques d'étude des stratégies adaptatives: approches mathématiques et informatiques. In Barbault, Blandin & Meyer (Ed.), *Recherches d'écologie théorique. Les stratégies adaptatives*. Maloine.
- Meyer, J.A. (1986). Intelligence artificielle et comportement naturel. *Etudes et Analyses comportementales*. 2(3): 154-169.
- Meyer, J.A. & Guillot, A. (1986a). La simulation numérique du comportement. *Etudes et analyses comportementales*. 2(3): 120-140.
- Meyer, J.A. & Guillot, A. (1986b). The energetic cost of various behaviors in the laboratory mouse. *Comp. Biochem. Physiol.* 83(3): 533-538.
- Meyer, J.A. & Guillot, A. (1990). From animals to animats: everything you wanted to know about the simulation of adaptive behavior. Technical Report (BioInfo-90-1). Ecole Normale Supérieure, Paris.
- Miller, G.F., Todd, P.M. & Hegde, S.U. (1989). Designing neural networks using genetic algorithms. In Schaffer (Ed.), *Proceedings of The Third International Conference on Genetic Algorithms*. Morgan Kaufmann.
- Miller, J.H. (1989). The co-evolution of automata in the repeated prisoner's dilemma. Technical Report (89-003). Santa Fe Institute.
- Minsky, M.C. (1963). Steps towards artificial intelligence. In Feigenbaum & Feldman (Ed.), *Computers and thought*. McGraw-Hill.
- Minsky, M.C. (1986). *The society of mind*. Simon and Schuster.
- Mowrer, O.H. (1960). *Learning theory and behavior*. Wiley.
- Moyson, F. & Manderick, B. (1988). The collective behavior of ants: an example of self-organization in massive parallelism. Technical Report (AI Memo 88-7). Artificial Intelligence Lab., Vrije Universiteit Brussels.
- Nachtigall, P.E. & Moore, P.W. (1988). *Animal sonar: processes and performance*. Plenum.
- Nilsson, N.J. (1984). Shakey the robot. Technical Report (323). SRI AI Center, Stanford University.
- Nolfi, S., Elman, J.L. & Parisi, D. (1990). Learning and evolution in neural networks. Technical Report (CRL TR 9019). Center for Research in language. Univ. of California.
- Nollet, P. (1988). Contribution à l'étude des stratégies adaptatives des insectes sociaux. Doctorat Thesis. Biomathématiques. Université Paris 7.
- O'Keefe, J. (1989). Computations the Hippocampus might perform. In Nadel, Cooper, Culicover & Harnish (Ed.), *Neural connections mental computations*. The MIT Press/Bradford Books.
- O'Keefe, J. & Nadel, L. (1978). *The hippocampus as a cognitive map*. Oxford Univ. Press.
- Packard, N. (1989). Evolving bugs in a simulated ecosystem. In Langton (Ed.), *Artificial Life*. Addison-Wesley.
- Parker, G.A. (1984). Evolutionarily stable strategies. In Krebs & Davies (Ed.), *Behavioural ecology: an evolutionary approach*. Blackwell.
- Parker, G.A. (1985). Population consequences of evolutionarily stable strategies. In Sibly & Smith (Ed.), *Behavioural ecology. Ecological consequences of adaptive behaviour*. Blackwell.
- Pearce, J.M. (1987). *An introduction to animal cognition*. Lawrence Erlbaum Assoc.
- Pichon, J.M., Blanes, C. & Franceschini, N. (1990). Visual guidance of a mobile robot equipped with a network of self-motion sensors. To appear in Wolff and Chun (Ed.), *Mobile robots*.
- Pomerleau, D.A. (1989). ALVINN: an autonomous land vehicle in a neural network. Technical Report (CMU-CS-89-107). Computer Science Dept. Carnegie Mellon University.
- Reeke, G.N. & Edelman, G.M. (1988). Real brains and artificial intelligence. In Graubard (Ed.), *The artificial intelligence debate. False starts, real foundations*. The MIT Press.
- Rescorla, R.A. & Wagner, A.R. (1972). A theory of Pavlovian conditioning: variations in the effectiveness of reinforcement and nonreinforcement. In Black & Prokasy (Ed.), *Classical conditioning II*. Appleton-Century-Crofts.
- Richard, G. (1984). Histoire des idées en Ethologie (1ère partie). *Etudes et analyses comportementales*. 2(3): 131-190.
- Richard, G. (1985). Histoire des idées en Ethologie (2ème partie). *Etudes et analyses comportementales*. 2(4): 195-254.
- Rivamonte, A. & Dral, A.D. (1990). Computer simulation of dolphin vision. Unpublished manuscript. Radiation Standards and Development Laboratory. Redstone Arsenal.
- Roitblat, H.L. (1987). *Introduction to comparative cognition*. Freeman and Co.
- Rudnick, M. (1990). A bibliography of the intersection of genetic search and artificial neural networks. Technical Report (CS/E 90-001). Department of Computer Science and Engineering. Oregon Graduate Institute.
- Rumelhart, D.E. & McClelland, J.L. (1986). *Parallel distributed processing. Explorations in the microstructure of cognition*. The MIT Press/Bradford Books.
- Salu, Y. (1984). Theoretical models and computer simulations of neural learning systems. *J. Theor. Biol.* 111: 31-46.
- Sannier, A.V. & Goodman, E.D. (1987). Genetic learning procedures in distributed environments. In Grefenstette (Ed.), *Genetic algorithms and their applications: Proceedings of the Second International Conference on Genetic Algorithms*. Lawrence Erlbaum Assoc.
- Schaffer, J.D. (1989). *Proceedings of the Third International Conference on Genetic Algorithms*. Morgan Kaufmann.

- Schaffer, J.D. & Grefenstette, J.J. (1990). A critical review of genetic algorithms. Unpublished manuscript. Philips laboratories. North American Philips Corporation.
- Schank, R.C. & Abelson, R.P. (1977). Scripts, plans, goals and understanding. An inquiry into human knowledge structures. Lawrence Erlbaum Assoc.
- Schmidhuber, J.H. (1990). Making the world differentiable: on using supervised learning fully recurrent networks for dynamic reinforcement learning and planning in non-stationary environments. Technical Report (FKI-126-90). Institut für Informatik. Technische Universität München.
- Sibly, R.M. & McFarland, D.J. (1976). On the fitness of behavior sequences. *Amer. Natur.* 110: 610-617.
- Simon, H.A. (1982). The sciences of the artificial. The MIT Press.
- Sloman, A. & Croucher, M. (1981). Why robots will have emotions. Proceedings of The Seventh International Joint Conference on Artificial Intelligence. Vancouver.
- Smolensky, P. (1988). On the proper treatment of connectionism. *Behavioral and Brain Sciences*. 11: 1-23.
- Steels, L. (1987). Self-organisation through selection. Technical Report (AI Memo 87-5). Artificial Intelligence Lab., Vrije Universiteit Brussels.
- Steels, L. (1989). Cooperation between distributed agents through self-organisation. Technical Report (AI Memo 89-5). Artificial Intelligence Lab., Vrije Universiteit Brussels.
- Sutton, R.S. (1990). Integrated architectures for learning, planning, and reacting based on approximating dynamic programming. Proceedings of The Seventh International Conference on Machine Learning.
- Sutton, R.S. & Barto, A.G. (1981). An adaptive network that constructs and uses an internal model of its world. *Cognition and Brain theory*. 4(3): 217-246.
- Sutton, R.S. & Barto, A.G. (1987). A temporal-difference model of classical conditioning. Proceedings of Ninth Conference of the Cognitive Science Society.
- Sutton, R.S. & Barto, A.G. (1989). Time-derivation model of Pavlovian reinforcement. In Gabriel & Moore (Ed.), *Learning and Computational Neuroscience*. The MIT Press.
- Thrun, S., Linden, A. & Moller, K. (1990). Adaptive look-ahead planning. To be published in (Dorffner, Ed.), *Proceedings OEGAI*. Springer Verlag.
- Tinbergen, N. (1951). The study of instinct. Clarendon Press.
- Toates, F.M. (1980). Animal behaviour. A systems approach. Wiley.
- Toates, F.M. (1986). Motivational systems. Cambridge Univ. Press.
- Toates, F.M. & Oatley, K. (1970). Computer simulation of thirst and water balance. *Medical and Biol. Engineer.* 8: 71-87.
- Todd, P.M. (1988). Evolutionary methods for connectionists architectures. Unpublished manuscript. Psychology Dept. Stanford University.
- Tolman, E.C. (1948). Cognitive maps in rats and men. *Psychological Review*. 55: 189-208.
- Travers, M. (1989). Animal construction kits. In Langton (Ed.), *Artificial Life*. Addison Wesley.
- Viola, P. (1988). Mobile robot evolution. Bachelor Thesis. Dept. of Electrical Engineering and Computer Science. MIT.
- Walter, W.G. (1950). An imitation of life. *Scientific American*. 182(5): 42-45.
- Walter, W.G. (1951). A machine that learns. *Scientific American*. 185(2): 60-63.
- Waltz, D.L. (1988). The prospects for building truly intelligent machines. In Graubard (Ed.), *The artificial intelligence debate. False starts, real foundations*. The MIT Press.
- Wang, G., Nonon, H., Diep, D., Dubant, O. & Esquirol, P. (1990). A possible approach for the fusion of multisensors data. Proceedings of The Third International IPMU Conference. Paris.
- Weir, M. (1984). Goal-directed behaviour. Gordon and Breach.
- Whitehead, S.D. & Ballard, D.H. (1990). Learning to perceive and act. Technical Report (TR 331). Dept. of Computer Science. University of Rochester.
- Whitley, D. & Hanson, T. (1989). The Genitor algorithm: using genetic algorithms to optimize neural networks. Technical Report (CS-89-107). Dept. of Computer Science. Colorado State Univ.
- Wiener, N. (1961). *Cybernetics*. 2nd edition. The MIT Press.
- Wilson, S.W. (1985). Knowledge growth in an artificial animal. In Grefenstette (Ed.), *Proceedings of The First International Conference on Genetic Algorithms and their applications*. Lawrence Erlbaum Assoc.
- Wilson, S.W. (1987a). Classifier systems and the animat problem. *Machine Learning*. 2: 199-228.
- Wilson, S.W. (1987b). Hierarchical credit allocation in a classifier system. In Davis (Ed.), *Genetic algorithms and simulated annealing*. Pitman.
- Wilson, S.W. (1990). The animat path to AI (this volume).
- Wilson, S.W. & Goldberg, D.E. (1989). A critical review of classifier systems. In Schaffer (Ed.), *Proceedings of The Third International Conference on Genetic Algorithms*. Morgan Kaufmann.

The Animat Path to AI

Stewart W. Wilson
The Rowland Institute for Science
100 Cambridge Parkway
Cambridge, MA 02142 USA
(wilson@think.com)

Abstract

A research methodology is proposed for understanding intelligence through simulation of artificial animals ("animats") in progressively more challenging environments while retaining characteristics of holism, pragmatism, perception, categorization, and adaptation that are often underrepresented in standard AI approaches to intelligence. It is suggested that basic elements of the methodology should include a theory/taxonomy of environments by which they can be ordered in difficulty—one is offered—and a theory of animat efficiency. It is also suggested that the methodology offers a new approach to the problem of perception.

1. Introduction

There are two broad approaches to the scientific understanding of intelligence, or how mind arises from brain. One is the natural science approach, analyzing and experimenting with phenomena of life, mind, and intelligence as they exist in nature. In this there are two main branches: physiology and especially neurophysiology, in which living systems are subject to detailed internal investigation; and experimental psychology, including studies of animals, in which living systems are studied through their external behavior. Related to the latter, but more observational, are fields such as linguistics and anthropology.

In contrast, the second broad approach to intelligence may be termed synthetic and computational, in which the objects studied are constructed imitations of living systems or their behavior. In "Computing machinery and intelligence", Turing (1950) suggested two possible directions for the computational approach:

We may hope that machines will eventually compete with men in all purely intellectual fields. But which are the best ones to start with? Even this is a difficult decision. Many people think that a very abstract activity, like the playing of chess, would be best. It can also be maintained that it is best to provide the machine

with the best sense organs that money can buy, and then teach it to understand and speak English. This process could follow the normal teaching of a child. Things would be pointed out and named, etc.

Turing's first proposed direction led to "standard AI" or computational cognitive science. Standard AI is basically competence-oriented, modelling specific human abilities, often quite advanced ones. However, while many AI programs exhibit impressive performance, their relevance for the understanding of natural intelligence is, in several respects, limited.

In addressing isolated competences, AI systems typically ignore the fact that real creatures are always situated in sensory environments and experience varying degrees of need satisfaction. Furthermore, the systems attach less importance to such basic natural abilities as perception, categorization, and adaptation than they do to algorithmic processes like search and exact reasoning. This leads eventually to problems connecting the arbitrary symbols used in internal reasoning with external physical stimuli ("symbol grounding" (Harnad, 1990)), and "brittleness" (Holland, 1986), the tendency for AI systems to fail utterly in domains that differ even slightly from the domain for which they were programmed.

AI systems also have an arbitrariness: it is often not clear why one program that exhibits a certain intellectual competence is to be preferred over some other one exhibiting the same competence, especially since the field has not agreed on—or too much sought—a clear definition of intelligence. In a sense, the programmer's facility for imitating a high-level fragment of human competence is a kind of trap, since from a natural science perspective there is usually no strong relation to nature.

Turing's second proposal, for a "child machine", received, over forty years, little attention or resources, perhaps because it seemed fantastic. Yet the child machine was to be situated from the start in a real sensory environment and was to learn through experience. It would have emphasized precisely the abilities that standard AI minimized. Turing's proposal is in fact

close to what we have called the "animat" (or artificial animal) approach (Wilson, 1985a), and the child machine is an advanced form of animat. Rather than isolated competences, the animat approach is holistic, focusing on complete systems (simulated or, when possible, realized) that, like animals, exist in realistic environments and must cope with the varied problems that they present.

Obviously, we can't yet simulate human intelligence holistically. But the basic *hypothesis* of the animat approach is that by simulating and understanding complete animal-like systems at a simple level, we can build up gradually to the human. At each point we will be careful to include full connection with a sensory environment, together with maximum use of perception, categorization, and adaptation. Thus when we reach the human level these crucial abilities will not be missing. We hope to reach human intelligence "from below", instead of piecemeal through high-level competences as in Standard AI.

The animat approach also brings with it a needed element of pragmatism (Holland, Holyoak, Nisbett & Thagard, 1986). Survival needs and their derivatives are evidently the principal drivers of animal behavior, and so, at bottom, they must be for human beings. The effect is that needs have a powerful influence on the formation of percepts and concepts—in machine learning terms, they set the *inductive bias* (Mitchell, 1980)—yet this has been little acknowledged in AI work. The animat approach explicitly brings in needs by making them the drivers of system behavior.

Introduction of needs opens the way to operational definitions of intelligence since the efficiency of need satisfaction is in principle quantifiable. For example, some years ago van Heerden (1968) summarized his observations on human intelligence as follows:

Intelligent behavior is to be repeatedly successful in satisfying one's psychological needs in diverse, observably different, situations on the basis of past experience.

With suitable changes, this definition can be applied from human to very simple animal levels. It brings in perception, categorization, and adaptation, and it bases degree of intelligence on rate of need satisfaction.

Our aim in this paper is to outline themes in the animat approach to AI—that is, to suggest "how to go about it", at least in first approximation. A number of efforts already exist [for a review, see Meyer & Guillot (1990); also see Smith & Wilson (1989)], and the interest is accelerating. We shall explain our view of what is needed, and suggest potential directions of formalization.

2. The Basic Animat Strategy

The basic strategy of the animat approach is to work toward higher levels of intelligence "from below"—using minimal ad hoc machinery. The essential process is incremental and holistic: given an environment and an animat with needs and a sensory/motor system that satisfies these needs to some criterion, increase the difficulty of the environment or the complexity of the needs—and find the minimum increase in animat complexity necessary to satisfy the needs to the same criterion. Alternatively, the environment could stay the same but the needs satisfaction criterion might be increased; again find the minimum animat complexity increase. In either case it is vital (1) to maintain the realism and whole-ness of the environment, however simple it is, so as to avoid special-purpose solutions; (2) to maximize physicality in the sensory signals, so as to avoid predefined symbolic inputs; and (3) to employ adaptive mechanisms maximally, to minimize the rate of introduction of new machinery and maximize understanding of adaptation.

Note that the strategy has a "problem side" (harder environments, increased efficiency) and a "solution side" (new architecture: sensory/motor, internal, adaptive). Changes in the problem side can be due to the experimenter, but also to (co-)evolutionary effects if the environment is evolving. Similarly, changes in the solution side can be deliberate or evolved, based on a genotype and selection.

Research on animat-like systems has tended to emphasize the solution side. A certain experimental environment is selected as being in some sense interesting, but most of the work goes into testing and refining a particular architecture in that environment. The result is often a successful system, but accompanied by insufficient insight from a formal point of view into the properties or difficulty of the environmental problem that has been solved, and with what efficiency. A major aim of this paper is to suggest the need for a more systematic understanding of environments.

We now discuss the problem and solution sides in more detail.

3. The Problem Side

3.1 Environments

Environments differ enormously in their complexity, uncertainty, and degree of reinforcement. Needed is a formal theory and taxonomy that will order environments and reveal their differences in difficulty. For example, some environments (e.g., some food concentration gradients) can be thought of as pure stimulus-response: the local environmental signal directly indicates the optimal action and provides reinforce-

ment immediately. In others, the reinforcement is deferred, though the optimal action is still knowable from the stimulus. An example would be an odor gradient that reliably pointed toward the location of food.

In a somewhat more complicated environment, information-bearing stimuli are not as simply related to reinforcement as odor is related to food. Instead they may consist of more or less arbitrary cues like stimuli from a certain kind of bush that prey like to hide in or, to mention a human context, a certain kind of golden arch! In still more complicated environments, the optimal action is no longer knowable from the immediate sensory stimulus. Consider leaving your office and turning in the correct direction in accordance with a phone call received five minutes earlier. Or the environment of a stalking animal in which the current objective is temporarily out of sight. At such times the immediate sensory stimulus may contain no information at all relevant to attaining the objective. Further complexity is of course introduced by the presence of competing creatures with similar or different needs. All cases become more difficult if environment characteristics are statistical, or stimuli or reinforcements contain noise. These are just examples, but they suggest a bit of the range and subtlety of real environments.

Given this variety, a more formal characterization is desirable. A start can be made by noting that from an animat's (or animal's) point of view, the environment is a kind of machine that (in general) responds with a new sensory stimulus (which may include reinforcement) whenever the animat executes a motor action. One formal way to describe such an environment is as a *finite-state machine* (FSM) for which the motor actions are inputs and the sensory stimuli are outputs (Riolo, 1987; Rivest & Schapire, 1987). The behavior of a finite-state machine is defined by two equations (Minsky, 1967):

$$\begin{aligned} Q(t+1) &= F(Q(t), A(t)) \\ E(t+1) &= G(Q(t), A(t)), \end{aligned}$$

where A is the machine's input (in this case the animat's motor action), E is the machine's output (in this case the sensory stimulus), and Q represents the machine's (the environment's) current "state". Time t is assumed to be discrete. The variables A and E are in general vectors.

The first equation says that the environment's next state is a function F of its current state and the motor action. The second equation says that the next sensory stimulus to the animat is a function G of the current state of the environment and the motor action. The FSM formalism captures the idea that actions in a given environment result in new sensory stimuli; the state variable Q makes it possible for the machine to respond differently to the same action in different circumstances, a common property of real environments. That the FSM is "finite-state" means essentially that the number

of different possible outputs for a given input is finite (though possibly very large), or equivalently, that the number of possible values of the state variable Q is finite. A strict finiteness property for real environments is perhaps debatable, but since large FSMs provide a good approximation in many problems of interest, the debate can be left for another occasion.

Besides reacting to animat actions with new sensory stimuli, real environments also sometimes present new stimuli in the absence of action (e.g., the clouds move while you gaze at them, other animats move in your field of view, etc.). This important property is not captured by the FSM formalism, and needs to be included in a fuller environment theory.

Before continuing, it is necessary to be quite careful about the meanings of E and A . Knowledge about the environment comes only through the use of the sensory and motor apparatuses, each acting as a kind of communication channel. Because these tend to be fixed in phylogeny ("hardwired") it is often useful to define the sensory and motor channels as part of the environment. Then, a particular environment of interest might consist, for example, of a (physical) maze as detectable by two eyes of a certain retinal description and manoeuvrable by four legs of a certain musculoskeletal description. From this, one would proceed to establish the appropriate functions F and G , treating the retinal outputs as E and the motor command signals as A .

Alternatively, one could treat the sensory and motor channels as part of the animat and not as part of the environment. Such a division might be desirable in problems in which the sensory and motor equipment was subject to an evolutionary process. However, for our current purposes we shall use the former approach in which the two channels are regarded as fixed and part of the animat's environment.

The FSM formalism has advantages and disadvantages. An advantage is that the environmental description can be as precise as desired, and it is necessary to be precise in order to program a simulation. A disadvantage is that the FSM description has a certain opaqueness from the point of view of understanding levels of environmental difficulty. A further disadvantage is that animats deal in stimuli and actions while the FSM also contains the state variable Q , which the animat can't detect. Though in some sense the animat should learn "the reality behind appearances", there is merit in examining an environmental formalism from which Q is absent.

Suppose we try to express the next sensory stimulus directly in terms of the current stimulus and current action. Examination of some FSMs will show that the result is in general not determinate, as indicated by the following relation:

$$\{E(t+1)\} = f(E(t), A(t)).$$

Here the next stimulus is indeed a function of the current stimulus and action, but, as indicated by curly brackets, $E(t+1)$ will be one of a *finite set* of possible stimuli, not unique. Although the set is determined by the function f , the particular member of the set that occurs is not. Thus the above relation expresses a *non-determinacy* of the environment with respect to the variables E and A .

We might term this kind of environmental description a *sensory-state machine* (SSM). In fact, for every FSM there is an SSM that can be derived by straightforward (though perhaps tedious) examination of the FSM state diagram. The SSM, trades the determinacy of the FSM for a formalism that expresses the environment—or the animat's problem—solely in terms of variables that the animat knows about. Furthermore, the SSM's non-determinacy is familiar: the reaction of an environment to an action is very often not fully predictable from knowledge of that action and one's immediate sensory situation.

Let us note that the SSM as defined above is an *incomplete* description of the environment. It can be derived from the environment's FSM, but the FSM cannot be derived from it (in general). Nevertheless, the SSM appears to be a more useful construct for understanding levels of environmental difficulty, as we now attempt to show.

Consider an environment which an animat detects through extremely limited sensory apparatus. For example, the animat might have only a single small touch detector, pointed straight ahead. The SSM for this environment-cum-sensory-apparatus would be extremely non-determinate, since a large number of object shapes would be consistent with stimulation of the single small touch detector. Should, for example, the animat turn 30 degrees to the right, subsequent stimulation of the detector would be nearly unpredictable.

On the other hand, consider an animat in the same environment but having elaborate stereoscopic vision. In this case the SSM would contain little non-determinacy, since for example the visual stimulation subsequent to the same 30 degree turn or most other actions would be a unique function of the current image and therefore predictable.

Predictability of the results of actions in the context of sensory stimulation is the foundation of an animat's survival and, indeed, prosperity. Attainment of reinforcement depends on the ability to choose actions that lead to reinforcement, whatever the sensory circumstances. The examples above suggest that the degree of non-determinacy of an environment's SSM is an important measure of the environment's relative difficulty.

A simple and tentative taxonomy of environments can be constructed based on SSM non-determinacy.

Class 0. Environments with completely determinate SSMs and in which for every sensory stimulus there exists at least one action which if taken will result in positive reinforcement. This might be called a pure stimulus-response environment, meaning that the optimal action in each situation is a function only of the current stimulus. The "landmark" environment of Barto & Sutton (1981) is an example of a Class 0 environment.

Class 1. Environments with completely determinate SSMs in which for only some sensory stimuli does there exist at least one action which will result in positive reinforcement. This could be called a stimulus-response environment with sparse or deferred reinforcement. The 288-state environment of Grefenstette (1988) and the maze environment of Sutton (1990) are examples of Class 1 environments.

Class 2. Environments with partially non-determinate SSMs. In contrast to Classes 0 and 1, reliable prediction can no longer be based on the current sensory stimulus and action. The environment "WOODS7" of Wilson (1985a) is an example of a Class 2 environment, as is the "Little Prince" environment of Rivest & Schapire (1987).

In many cases it will be possible to reduce or eliminate the non-determinacy of a Class 2 environment by taking into account some degree of recent history. For example, suppose that for a particular Class 2 environment we construct the *second-order* SSM:

$$\{E(t+1)\} = f_2(E(t), A(t), E(t-1), A(t-1)).$$

It may well be the case that this SSM is less non-determinate than the first-order one, the additional context of the prior time-step's stimuli and actions serving to reduce the uncertainty. We can further imagine that for *some* order of SSM, the non-determinacy is eliminated. Let that order be k . Then we could describe the environment in question as being of Class 2. k , with higher values of k standing for greater difficulty.

This concludes our discussion of environments, in which we developed the idea that environments could be ordered in difficulty according to the non-determinacy of their SSMs.

One complication that we have not mentioned, but will take up in Section 4.2, is the fact of "stimulus profusion" in real environments. The environment may not have a tricky SSM, but at the sensory interface it always has a *very* large one. Thus realistic environments pose a problem of selection of relevant data. At higher levels this can be a problem of "pattern recognition" and quite complex.

3.2 Efficiency

The second part of the problem side has to do with animat needs and the efficiency with which they are satisfied. An animat can have (1) somatic needs (food, shelter), (2) reproductive needs, (3) additional needs like play, exploration, and prediction. A particular problem may address just one or a few of these, depending on how reinforcement is defined. In addition, hierarchies of secondary needs can in principle result from the primary ones, though this should be explicated experimentally. In the end, satisfaction of all needs can be viewed as in the service of reproduction/survival.

Efficiency of needs satisfaction is the grounds for choosing one solution over another. The best way to do this is probably using a competitive, evolutionary approach in which solutions have costs, niches exist or can form, etc. Though data exist from natural science, determining costs will be difficult. Simulations may tend to use computational in contrast to true somatic costs. This could ultimately turn the animat approach away from nature and toward artificial worlds, where the implications for natural intelligence may not be clear. In any case, one would like to have a reasonable theory of animat efficiency in terms of need satisfaction that will take into account costs and provide criteria for preferring solutions.

4. The Solution Side

4.1 Architecture

Here there appears to be a great deal of choice. However, the animat approach (going slowly "upward") should permit a strong criterion of "necessary and sufficient". The progression should at some level of abstraction parallel what actually exists in nature, but that is conjecture; the parallels are unlikely to be obvious, given the apparent role of accident in evolution (Gould, 1989). We can expect, however, that if the SSM description of environments is valid and useful then the best architectures at each stage should be those that most efficiently cope with increasing environmental non-determinacy.

For example, stimulus-response (Classes 0 and 1) environments should imply any of a set of associative memories, which could be implemented with networks, etc. However, as soon as the sensed environment does not uniquely characterize its state (Class 2), the animat can only reach optimal performance using some form of short-term memory, which suggests recurrent networks, classifier systems, etc. Further Class 2 complication will occur when reinforcements are highly delayed and the system must form and retain an intention, and its subordinate intentions, etc., until reinforcement is obtained. Efficiency may then require the

introduction of higher-order internal states, modularity, etc. The animat strategy offers a way to bring these in naturally.

One interesting hypothesis is that the most efficient systems will be those that convert every frequently encountered important situation to one of "virtual stimulus-response" in which internal state (intention, memory) and sensory stimulus together form a compound stimulus that immediately implies the correct next intention or external action. This would be in contrast to a system that often tends to "figure out" or undertake a chain of step by step reasoning to decide the next action. The latter more contemplative system would presumably possess increased flexibility in the face of an uncertain environment. However, the present hypothesis is that greater overall efficiency will be found in systems that set up generalized S-R methods in the above sense. The motivation for the hypothesis is that in animals and people, even complex behavior, if frequent and important enough, tends to become reflexive. Standard AI has addressed the question of whether knowledge should be "interpreted" or "compiled" (Laird, Rosenbloom & Newell, 1986). The animat approach offers a new and perhaps more natural context in which to address it.

4.2 Perception

The preceding discussion of architecture bypassed the issue of sensory profusion, tacitly assuming inputs are few and well-defined, as in Standard AI. Perception—which might be defined as knowing what in the environment is relevantly the case—has proved very difficult to imitate computationally. It has a chicken and egg quality: How do you know what aspects of a complex profuse input to select or combine into patterns until you know how to view the input so as to find them, which in turn means knowing where or what they are in the first place.

One approach with some success is a combination of top-down and bottom-up processing in which, iteratively, fragmentary data from below suggest candidate remembered percepts above which in turn guide the lower search for confirming or disconfirming additional data [see, e.g., Grossberg (1987)]. Most uses of these and the related relaxation techniques [e.g., Geman & Geman (1984)] have occurred in the analysis of scenes or images from specialized domains, and so are subject to Standard AI's brittleness. In contrast, the animat approach, retaining stimulus-profuse environments but aimed at simpler percepts, should permit the development of more general and adaptive top-down/bottom-up strategies.

Woods-like environments containing simple objects ("tree", "food", etc.) offer an interesting test-bed for such strategies if the animat receives a stimulus that

simulates natural light reflected from the objects. Stimuli could be as profuse as desired depending on the variety of the objects, the resolution and spectral sensitivity of the light detectors, and the range of vision. In such environments, the experimental procedure should be alert to the possibility that the system might form effective percepts quite different from, e.g., "tree 90 degrees to the right", (as stimuli are often directly given in Standard AI). Among other things, the percepts would reflect the bias of the animat's needs.

Besides top-down/bottom-up methods, there is another class of techniques for dealing with stimulus profusion. Since the animat has effectors that can change the sensory input (as when the animat moves), he can learn ways of doing so that select information contingently. This ability becomes useful in cooperation with detectors that condense stimuli to different degrees in different parts of the sensory field. Then the animat can move so as always to place the highest-resolution detectors at the point of greatest interest, leaving the rest of the field relatively diffuse.

The focal-peripheral vision of some animals is an example of stimulus condensation and coordinated action that should be investigated using animats. In primates, for instance, the environment is seen in detail in a central region but with falling definition toward the visual periphery [see, e.g., Wilson (1983)]. This permits vast reduction of stimulus detail over most of the visual field, a primitive form of generalization, but the reduction can always be reversed by moving the eyes. Such a move would be triggered by current needs and internal state in combination with a stimulus cue sufficient to get through the peripheral generalization (Wilson, 1985b).

Perception is one of the hardest human abilities to understand. Progress in machine perception has been slow. The animat approach offers a fresh perspective because well-defined experimental mechanisms can be investigated in contexts that retain essential characteristics of real organisms and environments.

5. Summary

This paper has outlined "the animat path to AI", a strategy for progressively understanding intelligence or the relation of mind to brain that differs significantly from Standard AI, and from the natural science approaches to the same problem. The approach is not new, in that examples of prior work exist and are somewhat known. This paper however attempts to bring out the value of the approach, calls for a more systematic effort, and offers some working themes.

Fundamentally, the animat approach advocates maintaining the holism of the situation of real animals in real environments, while progressively but efficiently

increasing animat complexity only as necessary. The approach's hypothesis is that this program is feasible, and will ultimately lead to understanding of intelligence, adaptation, and perception at high levels. In support of the program, the paper proposes:

- (1) Creation of a theory/taxonomy of environments, based on a "sensory-state machine" formalism;
- (2) Establishment of criteria of animat efficiency in terms of need satisfaction and costs;
- (3) The hypothesis that efficient animats will have architectures that deal with frequent, important situations by "virtual stimulus-response";
- (4) The suggestion that the problem of "stimulus profusion" can be reduced through stimulus-condensing sensoria and contingent action.

References

- Barto, A. G. & Sutton, R. S. (1981). Landmark learning: an illustration of associative search. *Biological Cybernetics*, 42, 1-8.
- Geman, S. & Geman, D. (1984). Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence PAMI-6*, 721-741.
- Gould, S. J. (1989). *Wonderful Life*. New York: W. W. Norton.
- Grefenstette, J. J. (1988). Credit assignment in rule discovery systems based on genetic algorithms. *Machine Learning* 3, 225-245.
- Grossberg, S. (1987). Competitive learning: from interactive activation to adaptive resonance. *Cognitive Science*, 11, 23-63.
- Harnad, S. (1990). The symbol grounding problem. *Physica D*, 42(1-3), 335-346.
- Holland, J. H. (1986). Escaping brittleness: the possibilities of general-purpose learning algorithms applied to parallel rule-based systems. In R. S. Michalski, J. G. Carbonell & T. M. Mitchell (Eds.), *Machine learning, an artificial intelligence approach. Volume II*. Los Altos, California: Morgan Kaufmann.
- Holland, J. H., Holyoak, K. J., Nisbett, R. E., and Thagard, P. R. (1986). *Induction: Processes of Inference, Learning, and Discovery*. Cambridge, MA: MIT Press.
- Meyer, J.-A. & Guillot, A. (1990). Simulation of adaptive behavior in animats: review and prospect. *These Proceedings*.
- Minsky, M. (1967). *Computation: Finite and Infinite Machines*. Englewood Cliffs, NJ: Prentice-Hall.
- Mitchell, T. M. (1980). *The Need for Biases in Learning Generalizations* (Technical Report CBM-TR-117). New Brunswick, NJ: Rutgers University, Dept. of Computer Science.
- Laird, J. E., Rosenbloom, P. S., and Newell, A. (1986).

- Chunking in Soar. *Machine Learning*, **1**, 11-46.
- Riolo, R. L. (1987a). Bucket brigade performance: I. Long sequences of classifiers. *Genetic Algorithms and Their Applications: Proceedings of the Second International Conference on Genetic Algorithms* (pp. 184-195). Hillsdale, New Jersey: Lawrence Erlbaum Assoc.
- Rivest, R. L. & Schapire, R. E. (1987). A new approach to unsupervised learning in deterministic environments. *Proceedings of the Fourth International Workshop on Machine Learning* (pp. 364-375). Los Altos, CA: Morgan Kaufmann.
- Smith, S. J. & Wilson, S. W. (1989). Rosetta: toward a model of learning problems. *Proceedings of the Third International Conference on Genetic Algorithms* (pp. 347-350). San Mateo, CA: Morgan Kaufmann.
- Sutton, R. S. (1990). Integrated architectures for learning, planning, and reacting based on approximating dynamic programming. *Machine Learning: Proceedings of the Seventh International Conference* (pp. 216-224). Palo Alto, CA: Morgan Kaufmann.
- Turing, A. M. (1950). Computing machinery and intelligence. *Mind*, October, **59**, 433-460. Reprinted in *Computers and Thought*, E. A. Feigenbaum and J. Feldman, eds. (pp. 11-35), 1963.
- van Heerden, P. J. (1968). *The Foundation of Empirical Knowledge*. Wassenaar, The Netherlands: Wistik.
- Wilson, S. W. (1983). On the retino-cortical mapping. *Int. J. Man-Machine Studies*, **18**, 361-389.
- Wilson, S. W. (1985a). Knowledge growth in an artificial animal. *Proceedings of the First International Conference on Genetic Algorithms and Their Applications* (pp. 16-23). Hillsdale, New Jersey: Lawrence Erlbaum Associates.
- Wilson, S. W. (1985b). Adaptive "cortical" pattern recognition. *Proceedings of the First International Conference on Genetic Algorithms and Their Applications* (pp. 188-196). Hillsdale, New Jersey: Lawrence Erlbaum Associates.

WHAT IT MEANS FOR ROBOT BEHAVIOUR TO BE ADAPTIVE

BY DAVID MCFARLAND
Dept. Zoology, University of Oxford, U.K.

ABSTRACT

There is an analogy between animal and product design that can be formulated as an exact mathematical analogy. The success of a biological design is measured by the success of the genes that produce it, and this depends upon the ability of those genes to increase their representation in the population in the face of competition from rival genes. Similarly, when a variety of products is under consideration, they vary in the period required for product development, in the chance of failure in the market place, and in the expected returns from sales if the product is successful. The development period refers to the period before any return is achieved on investment. For animals this is the period between birth and reproduction, and for products it is the period prior to time that financial return accrues to the investor. The success of a design is evaluated by the net rate of increase of the genes coding for it (i.e. the return on investment) in the animal case, or, in the case of a product launched into the marketplace, of the money invested in it. If we are to take the biological approach to robot design seriously, then we should first consider the ecological (or market) niche that a proposed robot is to occupy. Is the robot to be a toy, a brick-laying robot, or a bomb-disposal robot? Just as there are no general-purpose animals, so there should be no general-purpose robots. For robot behaviour to be adaptive, in terms of the analogy, it must optimise with respect to the selective pressures of the market place. Other forms of adaptation, such as acclimatisation and learning, are subject to the same criteria.

The term adaptation, as used in biology has a number of meanings: Biologists usually distinguish between (1) evolutionary adaptation, which concerns the ways in which species adjust genetically to changed environmental conditions in the very long term; (2) physiological adaptation, which has to do with the physiological processes involved in the adjustment by the individual to climatic changes, changes in food quality, etc.; (3) sensory adaptation, by which the sense organs adjust to changes in the strength of the particular stimulation which they are designed to detect; and (4) adaptation by learning, which is a process by which animals are able to adjust to a wide variety of different types of environmental change.

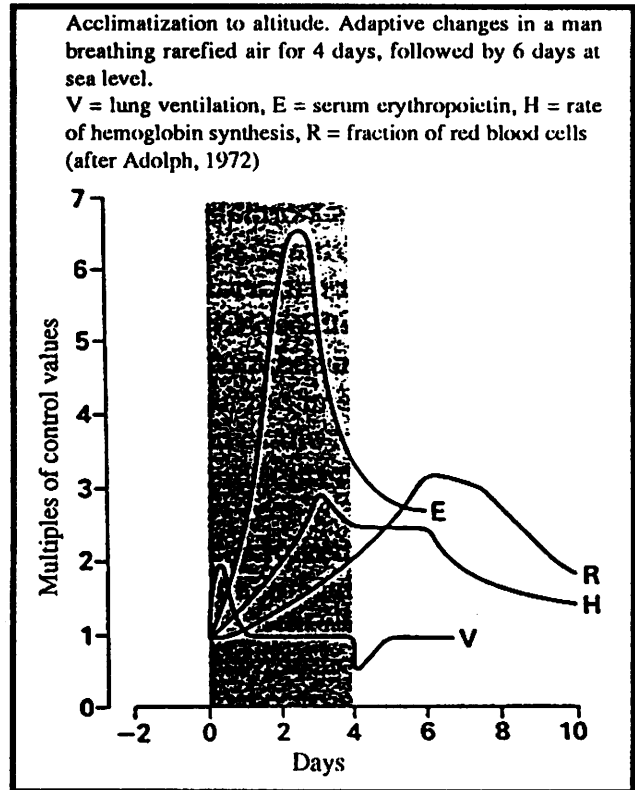


Fig. 1 Adaption by acclimatisation. The physiological changes that occur in acclimatisation to altitude run through a spectrum, ranging from fast but costly processes to slow-acting processes that are cheap in energetic terms.

Adaptation implies cost reduction, as can be seen from the example in Figure 1. In animal behaviour, real costs relate to Darwinian fitness. So, in considering the usage of the term adaptation in robotics, we should ask if there a concept equivalent to the fitness of a robot?

I will argue that there is such an equivalent concept, and that the analogy between animal and product design can be formulated as an exact mathematical analogy. Briefly, the success of a biological design is measured by the success of the genes that produce it, and this depends upon the ability of those genes to increase their representation in the population in the face of competition from rival genes. How does this

relate to product design? Suppose that a variety of products is under consideration. They vary in the period required for product development, in the chance of failure in the market place, and in the expected returns from sales if the product is successful. The development period refers to the period before any return is achieved on investment. For animals this is the period between birth and reproduction, and for products

it is the period prior to time that financial return accrues to the investor. The success of a design is evaluated by the net rate of increase of the genes coding for it (i.e. the return on investment) in the animal case, or, in the case of a product launched into the marketplace, of the money invested in it. This argument is outlined mathematically in Figure 2., and is to be spelled out fully by McFarland and Sibly (in prep).

ANIMALS

All gene-carriers reproduce at age t , and then produce n offspring. The chance of a juvenile surviving to breed is S . The instantaneous per capita rate of increase of genes coding for this life cycle is

$$r = \frac{1}{t} \log_e \left(S \frac{n}{2} \right)$$

NB The representation of a single gene after a period t is e^{rt} .

The expected number of surviving copies of a gene is $\frac{1}{2nS}$.

Therefore

$$\frac{1}{2nS} = e^{rt}$$

ROBOTS

All sales occur at the end of the development period t , and the development costs are k . Total returns are n , but these are realised only if the product successfully reaches the marketplace, which occurs with chance S . Hence expected returns, allowing for the chance of product failure, are Sn , and expected net returns are $Sn - kt$.

The rate of return is given by

$$r = \frac{1}{t} \log_e \frac{Sn}{k}$$

NB The expected gross income = Sn (the expected sales in money terms).

The return ration (gross returns/initial investment) = e^{rt} , assuming instantaneous r to compare investments over different periods of time.

$$\text{Therefore, } \frac{Sn}{k} = e^{rt}.$$

Fig. 2. Outline of the mathematical analogies between the life cycles of animals and robots (From McFarland and Sibly, in prep).

Figure 3 summarises the analogies between the design cycles of animals and robots. Development period t refers to the period before any return is achieved on investment. For animals this is the period between birth and reproduction, and for products it is the period before any financial returns accrue to the investor. S measures the chance of a product (animate or inanimate) surviving to enter the gene pool or marketplace. n refers to the return on investment. For animals this is the number of offspring produced when the individual reproduces (i.e. when the gene is copied into half the offspring on average, in accordance with Mendel's laws). In the case of man-made products, n refers to the income from sales. The success of the design is evaluated by the net rate of increase of genes coding for it, in the animal case, or the money

invested in it, in the case of man-made products.

The cases of longer adult life of sales period, can be analysed by an extension of the above method. In the simplest such case the adult achieves a constant breeding performance, producing n offspring every time it breeds, which occurs at intervals t apart. Similarly a product might achieve sales at a constant rate over a given period. If the animal is subject to mortality at a constant rate, the expected breeding performance is likely to follow a negative exponential curve. Similarly expected product sales might decline with time, taking into account the chances of failure to sell as a result of market competition or loss of market appeal.

ANALOGIES BETWEEN ANIMAL AND ROBOT LIFE CYCLES			
return on investments	n	number of offspring	gross sales income assuming no failures
reproductive probability	S	chance of juvenile surviving to breed	chance of product reaching market
development period	t	age at breeding	development cost = k
design success (rate of return)		net rate of increase of genes coding for design ("fitness")	net rate of increase of money invested in design (instantaneous interest rate)

Fig. 3. Table of analogies between animal and robot life cycles

1. THE ECOLOGICAL NICHE

If we are to take the biological approach to AI design seriously, then we should first consider the ecological niche that a proposed robot is to occupy. In animal ecology, the niche is the role that the animal plays in the community in terms of its relationship both to other organisms and to the physical environment. Thus a herbivore eats plant material and is usually preyed upon by carnivores. The species occupying a given niche varies from one part of the world to another. For example, a small-herbivore niche is occupied by rabbits and hares in northern temperate regions, by the agouti and viscacha in South America, by the hyrax and mouse deer in Africa, and by wallabies in Australia.

Niche occupancy usually implies ecological competition. When animals of different species use the same resources or have certain habitat preferences or tolerance ranges in common, niche overlap occurs. This leads to competition between species, especially when resources are in short supply. The competitive exclusion principle states that two species with identical niches cannot live together in the same place at the

same time when resources are limited. The corollary is that, if two species coexist, there must be ecological differences between them.

In thinking about designing robots, we should first think about what market niche we envisage. Is the robot to be a toy, a brick-laying robot, a bomb-disposal robot, or what. Just as there are no general-purpose animals, so there should be no general-purpose robots. If our robot is to be employed as a brick-layer, then it must be able to compete in the market place with human brick-layers. Of course, the two species of brick-layers will not be alike in every (brick-laying) respect. Employers will value one for certain qualities, and the other for other qualities. In other words there will be only partial niche overlap. Nevertheless, employers will apply roughly the same cost-efficiency criteria to each species, and these criteria will supply the selective pressures characteristic of the brick-layer niche. These selective pressures provide the main ingredients of the cost function that is characteristic of the environment in which the robot is to operate.

2. COST FUNCTIONS

In biology, the cost function "specifies the instantaneous level of risk incurred by (and reproductive benefit available to) an animal in a particular internal state, engaged in a particular activity in a particular environment" (McFarland, 1977).

An animal in a particular state runs a (specifiable) risk of incurring real costs. It is obvious that it is more risky to be in some states than others. For example, it is obviously dangerous to allow hunger to approach lethal levels if the food supply is not guaranteed. Thus the risk of death increases steeply the nearer a variable is to its lethal boundary. It has been shown (McFarland and Houston, 1981) that the risk function will generally be quadratic for state-variables that have a lethal boundary. In animal behaviour studies, the term 'cost' is generally used to signify the risk of death plus other factors that may lead to a decrement in Darwinian fitness, such as reduction of reproductive potential.

Such cost considerations apply to robots as well as animals. The bomb-disposing robot that runs out of fuel may as well be dead as far as its usefulness is concerned. Such a robot would have low fitness in the sense that it would not fare well in the market place. In considering the question of the fuel supply of our bomb-disposing robot, we assume that the robot has some sort of on-board energy store, and that robot

activity is an energy-consuming business. The robot is equipped with a sensor for monitoring the fuel level, but this is not perfectly accurate.

In Figure 4 we assume that x_f is the state of the fuel reserves at a particular time and that the probability P of the state x_f moving to a new position along the state axis is given by a normal distribution. As the robot goes about its business, the state x_f moves towards the boundary B at which the fuel reserves are zero. The area A represents the probability of crossing the boundary B . The area A increases as the square of the distance ox . In other words the risk of crossing the boundary B increases as the square of the fuel deficit, as shown in Figure 5.

The exact parameters of this function will depend upon the ecology of the refuelling situation. The frequency and reliability of refuelling opportunities will obviously be important factors influencing the risk of running out of fuel. The cost of the refuelling behaviour must also be taken into account. In the present case, we assume that the robot refuels by replacing its old rundown battery with a new recharged one. So it is the availability and location of the new batteries that matters.

In animal behaviour, real costs can be studied experimentally, and this is what some animal behaviourists spend their time doing. To obtain similar information about

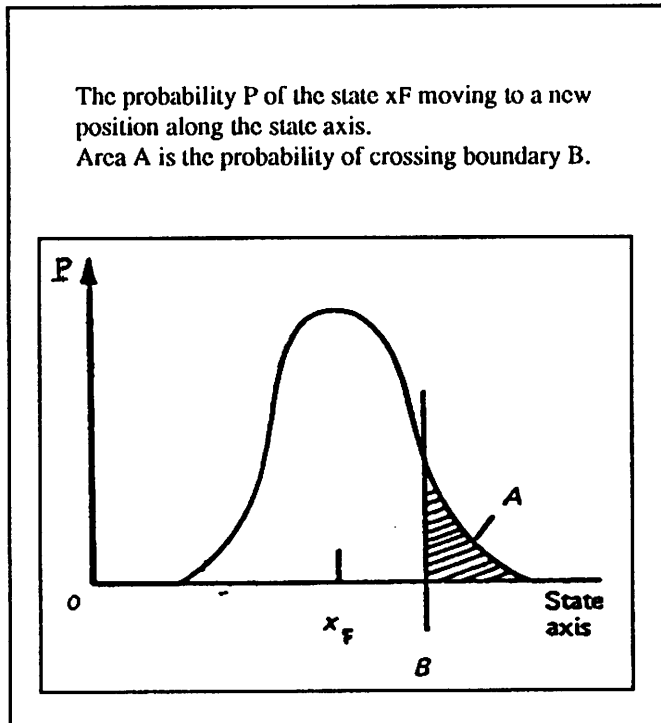


Fig. 4. The risk of fuel reserves falling below the lethal limit.

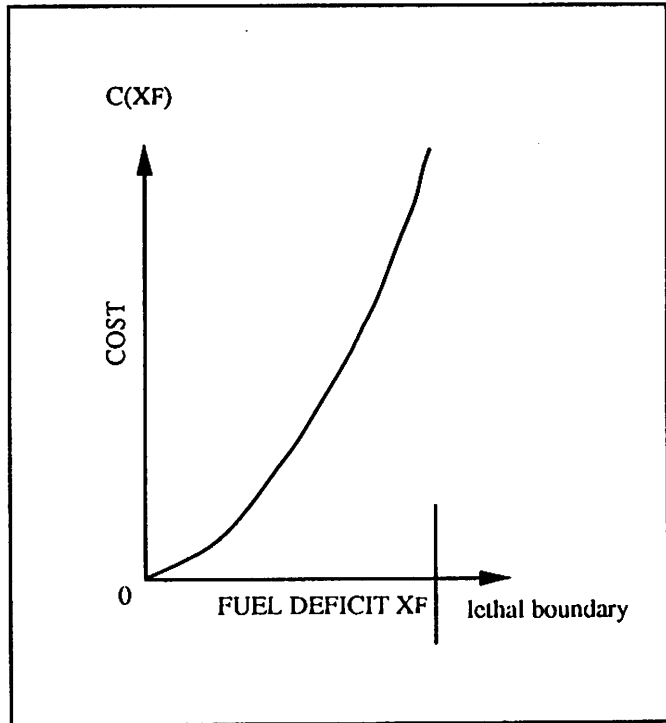


Fig 5. The cost of the fuel state as a function of the deficit.

the equivalent costs for a robot, it would be necessary to do field studies. Reliability studies of motor cars provide an example of this sort of approach.

Of course empirical study is unlikely to provide all the answers, and a good deal of computer modelling is required to fill in the gaps (For an example see Sibly and McCleery, 1985). Similarly, to evaluate the risks, costs and benefits of a robot launched into the market place, it would be necessary to do some ecological modelling that resulted in a cost function.

Let us now look at an example of a simple cost function that has received considerable attention from animal behaviourists.

Consider the two-dimensional quadratic cost function:

$$C(x) = Kx_1^2 + Lx_2^2 + Mu_1^2 + Nu_2^2 \quad (1)$$

where $C(x)$ is the instantaneous cost, x_1 and x_2 are state variables, u_1 and u_2 represent the rates of performing two activities, and K, L, M and N are scaling parameters, which we will assume to have value one, for our present purposes. The behaviour of an automaton would conform to this cost function if it obeyed the following simple rule:

if $x_1u_1 > x_2u_2$ perform activity u_1 , else perform activity u_2 .

We will call this RULE B.

The effect of applying this rule can be seen in Figure 6. In this figure RULE B is contrasted with an even more simple rule (RULE A), which is to perform that activity for which x is larger. Figure 6 shows that the two rules give different behaviour (shown here in terms of the reduction in x), and different cumulative costs, as calculated on the basis of the cost function. RULE B is less costly and comes close to the optimal solution for this problem (Sibly and McFarland, 1976).

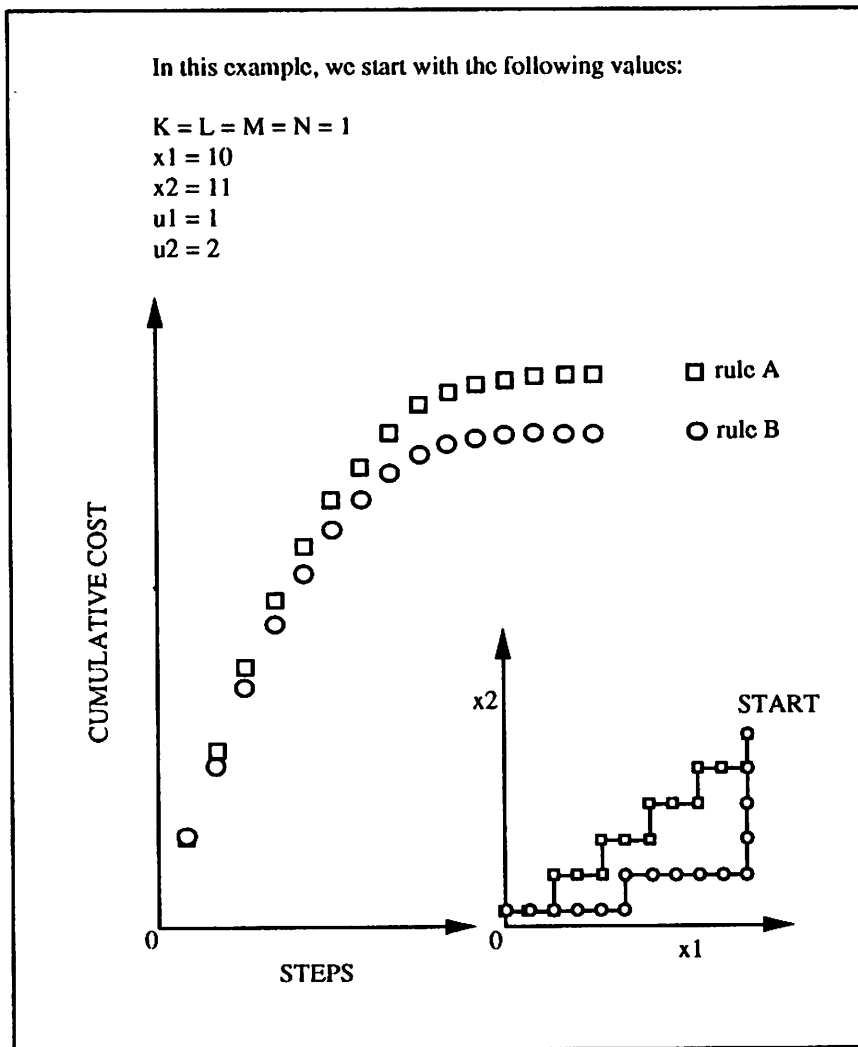


Fig. 6

The cumulative cost of following rules (large graph), and the changes in state that result (small graph)

Notice that it is possible for the rules governing behaviour to conform to some extremal or optimality principle, without there being any obvious sign of this in the formulation of the control system. Of course, the cost function can be explicitly represented (in which case McFarland and Houston (1981) would call it a goal function), as would be necessary in a planning system.

As a matter of fact, a simple planning approach to the same problem gives a very similar cost profile. (McFarland, unpublished). If we plan one step ahead (PLAN1), then we simply ask what next activity will result in the lower cost as calculated from the cost function. In planning two steps ahead (PLAN2) we ask which subsequent activity yields the lower cost.

If we compare PLAN1 and PLAN2 we find, not surprisingly that PLAN2 results in lower cost than PLAN1. If, however, we compare with our automaton (ABOVE), we find that $PLAN2 < AUTOMATON < PLAN1$. The performance differences are very small, but there are vast differences in the capital costs involved. The planner requires some reference to a representation of the cost function (called the goal function). The planner has to compute the notional cost of each of its various options, and then compare them. This all requires investment in hardware. It is difficult to know exactly how to make these calculations. I have tried various ways, and I have come to the conclusion (based on comparison of analogue simulations) that $PLAN2 = PLAN1/9 = AUTOMATON/90$. In other words it would take a brain 90 times the size of the AUTOMATON brain to improve on the automaton performance

by planning.

The cost function provides a performance index against which the behaviour produced by a particular brain architecture can be evaluated objectively - that is in terms of the robot ecology, or in terms of the real world. In comparing different brains for a given robot designed to operate in a given ecological niche - what we should be looking at is value for money. Figure 7 shows a plot of value for money versus price for the four different ways of minimising the quadratic cost function discussed above. The price is calculated on the basis of the development costs (assuming no profit). So this is the absolute minimum price that could be charged without incurring a loss. The actual price charged would, of course, depend upon the expected sales.

In terms of value for money, RULE B is by far the best buy. Of course, this is a very simple problem, with only 2 dimensions. For more complex problems we would expect a planning approach to result in some savings in brain size, because a planner can re-use certain brain areas over and over again, whereas an automaton's brain is made up of circuits dedicated to particular procedures. Moreover, a brain that has some representation of the cost function (i.e. a goal function) should be capable of learning. So we might expect some improvement in performance.

Overall, we should expect that, with increasing complexity, the value for money provided by an automaton should decline, because the increasing cost of the hardware will mean that the selling price will have to increase. The value for money provided by a system capable of planning and

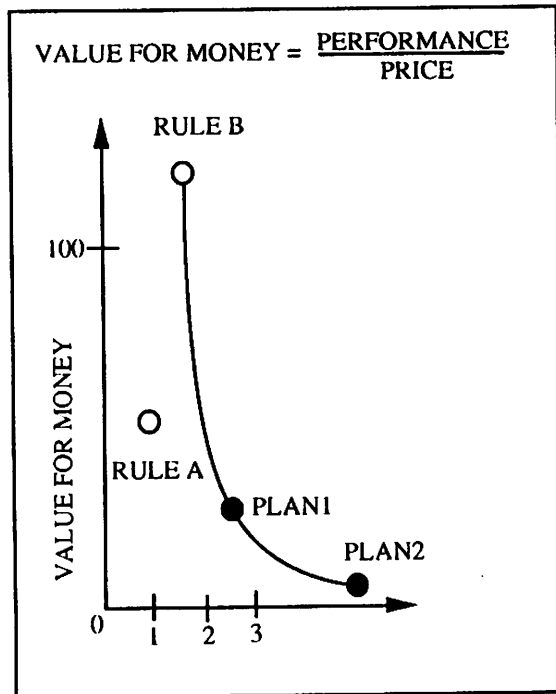


Fig. 7. Value for money of different rule-following and planning solutions to minimising the 2 dimensional quadratic cost function.

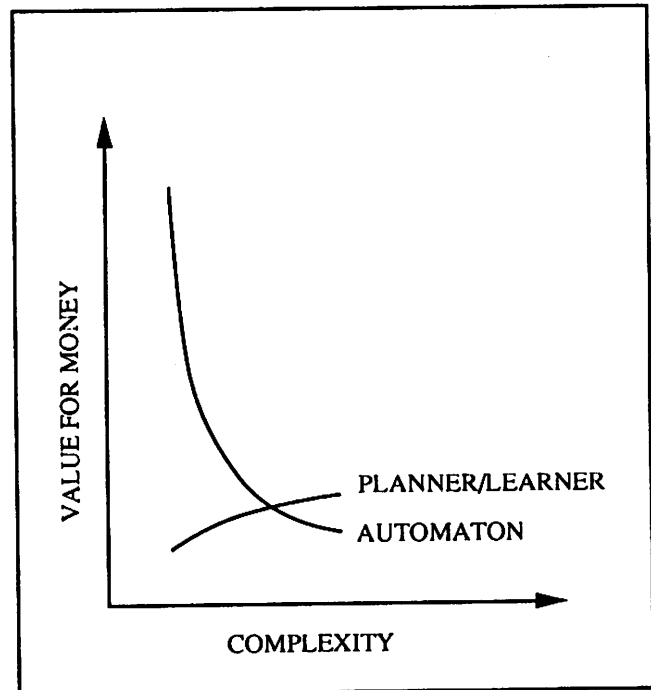


Fig. 8. Value for money of automation (rule following) and planning brain architectures as a function of the complexity (dimensionality of the niche) of the agent's life-style.

learning should increase, because of improvements in performance, as illustrated in Figure 8. We should remember, however, that in a multitask agent, the automaton solution may be better for some tasks, and the planning solution for others. Therefore, complex multitask agents should have a modular architecture, and I see the kind of exercise that I have sketched here, as leading to methods of costing different brain architectures.

3. LEARNING

So far we have concentrated upon evolutionary adaptation and its industrial equivalent. Let us now turn to adaptation of the individual. The equivalent of physiological adaptation is familiar to engineers as adaptive control.

An adaptive control system has the capacity to modify its strategy of seeking some optimum, or set of performance criteria, either by reference to some idealised form as represented by a model, or by learning based on the remembered results of past behaviour. Such adaptive control systems occur in animals. (See for example McFarland, 1971, pp 109-116).

Adaptation of individual behaviour also occurs in the form of learning. True learning requires some kind of planning architecture, which embodies a representation of the cost function. An automaton has no representation of the cost function, and can, therefore, engage only in preprogrammed forms of learning. The planning architecture embodies a goal function representation (which we call the goal function), and is therefore in a position to learn, in the true sense of the word. True learning requires some access to a representation of the cost function. Trial and error implies an experiment on the part of the agent, the results of which have to be evaluated. The results can be evaluated only by reference to an immutable set of values. For learning to be adaptive (i.e. of advantage) this set of values must reflect the cost function that is characteristic of the ecological niche. The more the set of values, or goal function, resembles the cost function then the better adapted will be the agent (see McFarland & Houston (1981) for a discussion). Note that it is not possible for the agent to modify the goal function to make it more like the cost function. Learning requires feedback about success and failure, and this feedback must be based upon an immutable set of values. This set of values form part of the goal function.

Learning has to operate in relation to a representation of the cost function, which we call the goal function. So the improvement in performance that results from learning is an improvement in relation to the goal function. Now the goal function may or may not be a good representation of the cost function that is characteristic of the robots ecological niche. In animals it can never be a perfect representation, because of

evolutionary lag, genetic diversity, and ecological competition. In robots the same logic holds. This means that learning is not always adaptive. Learning in robots is adaptive only in so far as the goal function resembles the cost function.

4. REFERENCES

- McFarland, D. (1971) *Feedback mechanisms in animal behaviour*. Academic Press, London.
- McFarland, D (1977) Decision-making in animals, *Nature*, 269, 15-21.
- McFarland, D. and Houston, A. (1981) *Quantitative Ethology: the state-space approach*. Pitman Books, London.
- Sibly, R.M. and McCleery, R.H. (1985) Optimal decision rules for gulls. *Animal Behaviour*, 33, 449-465.
- Sibly, R.M. and McFarland, D.J. (1976) On the fitness of behaviour sequences. *American Naturalist* 110: 601-617.

Computational Neuroethology: A Provisional Manifesto

Dave Cliff

University of Sussex School of Cognitive and Computing Sciences

Brighton BN1 9QN, England, U.K.

E-mail: davec@uk.ac.sussex.cogs

Abstract

This paper questions approaches to computational modeling of neural mechanisms underlying behaviour. It examines the “simplifying” (connectionist) models used in computational neuroscience and concludes that, unless embedded within a sensorimotor system, they are meaningless. The implication is that future models should be situated within closed-environment simulation systems: output of the simulated nervous system is then expressed as observable behaviour. This approach is referred to as “computational neuroethology”. Computational neuroethology offers a firmer grounding for the semantics of the model, eliminating subjectivity from the result-interpretation process. A number of more fundamental implications of the approach are also discussed, chief of which is that insect cognition should be studied in preference to mammalian cognition.

1 Introduction

This paper questions approaches to computational modeling of the neural mechanisms underlying behaviour. It examines the relationship between computational neuroscience [26] and that style of modeling popularly referred to as “connectionism”, “parallel distributed processing”, or “neural networks”¹ which has recently been subject to renewed attention in the fields of cognitive science and artificial intelligence (see e.g. [25, 23]).

Connectionist models are characterised by their simplified nature and concomitant inattention to biological data, and it is argued here that such “simplifying” computational neuroscience has serious inadequacies. A different approach is suggested which pays far more attention to the sensorimotor system and hence to behavioural interactions with the external environment. This approach involves computational modeling of the neural mechanisms underlying behaviour, in a manner akin to that used in connectionism. Such an analysis of behaviour as a product of neural activity is properly the

domain of the field of neuroethology, and the new approach is therefore referred to as “computational neuroethology”. Meaning is supplied to the models by embedding them in simulated environments which supply visual feedback without human intervention, that is they close the external feedback loop from motor output to sensory input.

The advantage of computational neuroethology is that the semantics of the network are well grounded, and thus results are generated by observation rather than by interpretation. That is, the fruits of computational neuroethology simulations are “hard” objective measurements rather than “soft” subjective ones. At a metatheoretical level, it is argued that the computational network simulation of cognitive processing should pay much more attention to the evolutionary history of those faculties it wishes to replicate. In particular, a conclusion of this paper is that the study of linguistic processes using network models is wildly premature. The study of insects is advocated as the most fruitful path for future research.

As the reader will probably already have detected, this paper is intentionally polemic. It is aimed at an interdisciplinary audience, and the author is no polymath. For that reason, this paper is offered as a provisional manifesto in the hope that it provokes some interesting discussion. The argument is based on previous work by a number of authors. Because of its disputatious nature, there are more direct quotes in this paper than is common. There is no denying that this is a *selective* review of the literature. This paper is abridged from [9].

The paper opens with a discussion of computational neuroscience, distinguishing it from neural engineering, and identifying two classes of model: realistic and simplifying. Following this, the connectionist paradigm is briefly summarised. Next, criticisms of connectionism are discussed, with particular attention to the argument that connectionist models have no semantic grounding without behavioural linkage to a sensorimotor system. Then, a remedy to this objection is proposed: the adoption of the computational neuroethology approach. Computational neuroethology is defined, and a specific technique for providing a behavioural linkage is discussed. This approach has some important implications for future research, the most significant of which

¹In this paper, these three terms will be treated as synonymous, and referred to collectively as “connectionist” models.

is that it encourages an evolutionary approach to understanding intelligence.

2 Neurosomething

The recent excitement in the cognitive science and artificial intelligence communities about “neural” issues has been motivated by a number of interests, and consequently the terminology has not been developed in a uniform manner. The first requirement is to distinguish between issues of science and issues of engineering. To this end, two important terms, computational neuroscience and neural engineering, are informally defined below.

- *Computational Neuroscience* is the study of real brains or nervous systems as computational systems, “...in the sense of representing, processing, and storing information.” [20, p.3]. Sejnowski and Poggio define it as:

“...an approach to understanding the information content of neural signals by modeling the nervous system at many different structural scales, including the biophysical, the circuit, and the systems level.” [20, series forward]

Most often, this entails modeling the system under study on a digital computer. The term *computational neurobiology* [1, p.186] is here taken to have meaning identical with *computational neuroscience* [*ibid.* p.203].

- The construction of intelligent systems and massively parallel (fine-grain) digital computers inspired by principles found in naturally occurring neural systems has been referred to as *neural computing* [1, p.186] and *neural engineering* [1, p.203]. In this paper only the latter term will be used. The most important feature that distinguishes neural engineering from computational neuroscience is that in neural engineering the onus is on optimality. That is, neural engineering usually has the goal of creating an artefact that is (near-) optimal with respect to some measure of efficiency.

Computational neuroscience and neural engineering are not mutually exclusive endeavours: “...the two subjects have different goals. Nonetheless, they overlap and are mutually stimulating.” [1, p.186]. But the science/engineering distinction is an important one:

“Engineering is often based on science, but its aim is different. A successful piece of engineering is a machine which does something useful. Understanding the brain, on the other hand, is a scientific problem.” [10, p.132].

This paper is concerned only with science; not with engineering. In particular, it is concerned with the relationship of computational neuroscience to cognitive science. It is not concerned with any attempt to create ‘intelligent’ artefacts that employ optimal processing strategies: that is a matter for artificial intelligence (AI); the processing strategies of naturally occurring cognitive activity may well be suboptimal, and (for the purposes of this paper) cognitive science will be considered as studying only naturally plausible models of cognitive processing. That is, I’ll assume that the creation of what Dennett [11] calls “cognitive wheels”² is properly only the domain of AI; although, of course, some cognitive wheels may be lurking within the cognitive science canon, yet to be refuted.

3 Computational neuroscience

The advantages claimed [26] for modeling and simulation in computational neuroscience over conventional experimental techniques include:

1. Increased accessibility to the consequences of complicated nonlinear brain systems with many interacting components.
2. The possibility of discovering new phenomena by comparing experimental results to the predictions of simulation, and using these predictions as the basis for the design of new experiments.
3. Facilitating experiments (such as selective lesioning or ablation of particular channels, synapses, neurons or pathways) which would be difficult or even impossible to perform on living tissue.

Practitioners of computational neuroscience employ two classes of brain model: *realistic* models and *simplifying* models; these are discussed in more detail below. These two classes of model represent the extreme points of a continuum: in practice, models are likely to have features of both, and many different types of model will be required to span all levels of analysis.

3.1 Realistic Models

A realistic model “...consists of a very large scale simulation that tries to incorporate as much of the cellular detail as is available” [26, p.1300]. The realism of the model, while offering the advantages listed above, introduces two important weaknesses [26, p.1300]:

1. As more parameters and variables are added to the model to increase its realism, the complexity of the model grows, and so there is an increasing danger of

² “...an elegant but unnatural solution to a problem of natural design.” [6, p.65]

the simulated nervous system being as poorly understood as the real thing.

2. The results may be invalidated by the inadvertent exclusion of important features, because all the cellular details are not yet known.

Probably the best known realistic models are the Hodgkin-Huxley neuron model [16] and the Hartline and Ratliff model of lateral inhibition in the eye of *Limulus* [24]. The realistic modeling approach is most appropriate when knowledge of the circuit to be modeled is almost complete down to the biophysical level, and the function of the circuit is already known [22, p.361].

3.2 Simplifying models

The use of “simplifying” models is an approach applied at the network level of nervous systems: the procedure involves starting with a function such as a perceptual ability and designing “simplified neural circuits that can perform the function within the constraints of the state of knowledge.” [22, p.361].

Proponents of simplifying models claim that they offer greater conceptual clarity, and that they fulfill a perceived need for models capturing important principles. Their supporters propose [26, p.1300] that they are analogous to aspects of the physics literature such as textbook examples that admit exact solutions:

“These models abstract from the complexity of individual neurons and the patterns of connectivity in exchange for analytical tractability.” [26, p.1301].

Sejnowski *et al.* state [26, pp.1300–1301] that simplifying models include those currently being investigated under the general headings of connectionist models, parallel distributed processing models, and “neural networks”; they note that some of the models reported on under these headings are used not for brain research, but for neural engineering or for the study of purely psychological processes. It is those connectionist models used for brain research that are discussed here.

4 Connectionism

Again, this paper concentrates only on connectionist models which are also simplifying computational neuroscience models: engineering or purely psychological models are ignored. Examples of all three styles can be found in [25, 23].

It is not within the scope of this paper to attempt a comprehensive definition of connectionism, nor is it possible to fully examine here the position of connectionism in the context of past practice in cognitive modeling. However, both of these tasks have been performed

by Smolensky [27], whose definitions are quoted below. They are presented here as representative of the field, rather than as the view of a small group of Smolensky cohorts.

“Connectionist models are large networks of simple parallel computing elements, each of which carries a numerical *activation value* which it computes from the values of neighboring elements in the network, using some simple numerical formula. The network elements, or *units*, influence each other’s values through connections that carry a numerical strength, or *weight*. ...connections carrying positive weights are called *excitatory* and those carrying negative weights are *inhibitory*.

“In a typical connectionist model, input to the system is provided by imposing activation values on the *input units* of the network; these numerical values represent some encoding, or *representation* of the input. The activation on the input units propagates along the connections until some set of activation values emerges on the *output units*; these activation values encode the output the system has computed from the input. In between the input and output units there may be other units, often called *hidden units*, that participate in representing neither the input nor the output.

“The computation performed by the network in transforming the input pattern of activity to the output pattern depends on the set of connection strengths; these weights are usually regarded as encoding the system’s knowledge.” [27, p.1; original emphasis]

Smolensky describes connectionism’s relation to the two main levels of analysis in previous work: the neural (as in realistic computational neuroscience) and the symbolic (as in nearly all work in cognitive modeling and artificial intelligence (AI) up until the early 1980’s). Smolensky posits connectionism as constituting a new level of analysis, intermediate between the symbolic and neural levels. This he refers to as the *subsymbolic* level. The argument for the existence of this new level is involved, and will not be repeated here; see [27].

5 From frying pan to fire?

Connectionism has not been received with universal acclaim. Criticisms have been made both of specific models and of the methodology as a whole. Presented below are summaries of critiques by some other authors, along with some criticisms of my own. Again, this paper is only interested in connectionism *qua* simplifying computational neuroscience. I am aware that some criticisms have been replied to by proponents of connectionism, but a full review of the debate would be an unwelcome digression here. The four criticisms made below are those

which I feel connectionism has most consistently failed to acknowledge or answer.

5.1 *The Hunting of the Snark*

A forceful critique of the connectionist approach was made by Crick [10]. He argues that connectionist “models” do not correspond sufficiently closely to real neural systems to be regarded as models in the usual sense: “In another context they might reasonably be referred to as existence proofs. As such they have a certain use.” [10, p.131]. He attacks connectionist research on methodological and ontological grounds.

His methodological objection is that connectionists employ mathematics as an intellectual prop: he suggests “...that within most modellers a frustrated mathematician is trying to unfold his wings. It is not enough to make something that works. How much better if it can be shown to embody some powerful general principle for handling information, expressible in a deep mathematical form, if only to give an air of intellectual responsibility to an otherwise rather low-brow enterprise.” [10, p.132]. Lehnert made a similar criticism, she argues that “the interdisciplinary appeal of connectionism is not so much a computational appeal, as it is an appeal based on theorem envy. ...connectionism has come to the rescue of a new generation of psychologists who are really closet mathematicians and physicists.” [27, p.40].

Crick criticises past work in psychology and linguistics for its preference for simple “intelligible” models, an approach rooted in the belief that the brain is intractably complex. He notes that proponents of such an approach are generally not concerned by the absence of criteria of biological or psychological feasibility, and characterises their philosophy thus: “If it describes, in a succinct way, some of the psychological data, what can be wrong with that? Notice, however, that by using such arguments, one could easily make a good case for alchemy or the existence of phlogiston” [10, p.131].

Crick’s ontological objection is rooted in his methodological criticism. He argues that the search for mathematical expression is highly questionable because there is no guarantee of deep general principles being embodied in the functioning of the brain. He offers the genetic code as a good example of a complex natural system which is not easily characterised by a small set of general principles: the brain may achieve its aims using a “series of slick tricks” [10, p.132]. He contends that further research should be aimed at resolving this issue:

“Learning about neurons, their behaviour and their connections, will not by itself solve our problems, but will at least suggest the sort of answer to look for and can be used, often rather decisively, to disprove false theories.” [10, p.132].

5.2 *Biological studies of cognition*

This preoccupation with elegant mathematics has led, I believe, to connectionists ignoring a significant school of thought in neuroscience: that of Humberto Maturana and his colleagues. Maturana’s work is concerned with understanding how phenomena such as cognition and language are rooted in biological processes. His work is challenging, using a large specialised vocabulary, and a full review of it is way beyond the scope of this paper: it will suffice here merely to nod in his direction. The notes that follow are based on an overview of his work by Winograd and Flores [28, pp.38–53].

The most germane of Maturana’s arguments is that the nervous system should not be treated as an input-output device. But most connectionist models do exactly that. That is, they treat the neural function to be modeled as being implemented on a ‘pipeline’ processor. The most likely reason for this is the mathematical tractability of “feed-forward” network models, where each unit sends output only to units in subsequent layers in the network (i.e. later stages of processing). For instance, the “back-propagation” network learning algorithm [25, ch.8], the subject of much attention in the literature, operates only on feed-forward networks and is defined solely in terms of the input-output profile of the function at hand.

This maths-driven nature of connectionist architecture is acknowledged by Smolensky: “In the drive for more computational power, architectural decisions seem to be driven more and more by mathematical considerations and less and less by neural ones.” [27, p.9].

So, the view (implicit in much connectionism) of nervous systems as input/output pipeline devices is mistaken and should be avoided.

Maturana further argued that there is no clear modular separation between an entity and its ‘environment’ [28, p.43]: an entity is defined only in terms of its surrounding environment. This is an issue ignored by virtually all current connectionist models. Most treat the portion of the nervous system under study as capable of being modeled *in vacuo*. At best, connectionist models are extremely poor approximations to *in vitro* studies.

5.3 *Microscope Envy*

Unfortunately, the connectionist disregard for past work in biology does not stop at the philosophical level of Maturana. Lehnert’s diagnosis of connectionists as closet physicists suffering from theorem envy was noted above. I suggest that the complementary case also applies: some connectionists are physicists or mathematicians who are closet biologists suffering from microscope envy. Specifically, it appears that one concrete achievement of the connectionist paradigm has been to recast old problems of the symbolic paradigm at a new, biological, level.

This is a problem rooted in the cavalier attitude most proponents of connectionism adopt towards biological reality. Only one example of this is presented here: see [9] for another.

5.3.1 *Micro-worlds, again*

This example concerns the micro-world trap in symbolic cognitive modeling. This is about previous work in the symbolic AI paradigm which proposed to have created systems which ‘understood’ concepts or which were claimed to exhibit learning or ‘discovery’ of new concepts. There is a lot of such work, and a comprehensive literature review is out of the question here: this is an eclectic overview of the argument.

Essentially, the criticism of such models (which dogged AI research through the early 1970’s; see [12] for a discussion and pointers to the micro-world literature) was that all the ‘understanding’ was actually being performed by the programmers, because in creating the system it was necessary to create a micro-world for the system to work on. Such toy worlds implicitly incorporated a vast amount of pre-processing.

Similarly, systems which were purported to learn or discover new concepts make their discoveries by “... working on data presented in notational formats that represent the fruits of centuries of human labor. Manipulating these representations could be the tip of the iceberg; creating them and understanding them may constitute the unseen bulk.” [6, p.14].

One of the appeals of the connectionist approach listed by its proponents is the availability of autonomous learning procedures for tuning the weights in the network, thereby ‘programming themselves’ to perform the task at hand. At first glance, the supposed autonomy (no human intervention is required to specify the changes or to actually alter the weights) seems to indicate that connectionism does not suffer from micro-world problems.

Any learning network requires a source of input vectors — specifications of the activities of the input units of the network. And there’s the rub. Unless the network is involved in the first stages of vision or hearing, the input vectors have to be prepared “off-line”, invariably by humans rather than by other connectionist networks. That is, connectionist models rely on pre-processed information and are thus susceptible to the same problems as beset the micro-world studies of the symbolic paradigm. Smolensky acknowledges this problem. He notes [27, pp.7–8] how important the choice of representation for input (micro)features is, and that a model’s performance depends crucially on the input and output representations chosen by the modeler.

Probably the most celebrated example of this is the Rumelhart and McClelland verb past-tense learning system [23, pp.216–271]: its ‘wickelfeature’ representation scheme (among other things) has been attacked, most

notably in the critique by Prince and Pinker (e.g. [27, pp.46–47]) where the model is described as a fairy-tale account of the actual cognitive process.

5.4 *Grounding the semantics*

This final criticism of connectionism continues the theme of the *in vacuo* problem, further demonstrating the extent to which connectionism exists within a biological vacuum. This criticism is Lakoff’s critique of the semantics of connectionist systems; Harnad [15] makes a similar point. Lakoff [21] criticises Smolensky’s [27] account of the connectionist paradigm as making a huge omission in ignoring the body. He states:

“The neural networks in the body do not exist in isolation; they are connected to the sensorimotor system. For example, the neurons in a topographic map of the retina are not just firing in isolation for the hell of it. They are firing in response to retinal input, which in turn is dependent on what is in front of one’s eyes. An activation pattern in the topographic map of the retina is therefore not merely a meaningless mathematical object in some dynamical system; it is *meaningful*. A different activation pattern over those neurons would mean something different. One cannot just arbitrarily assign meaning to activation patterns over neural networks that are connected to the sensorimotor system. The nature of the hookup to the body will make such an activation pattern meaningful and play a role in fixing its meaning.

“Compare this, for example, with a string of symbols in a ... computer program. The symbols are not meaningful in themselves. They have to be “given meaning” by being associated with things in the world. If the symbols are to stand for categories, those symbols must be given meanings by being associated with categories that are out there in the world.” [21, p.39]

The essence of Lakoff’s argument is thus that if neurons are appropriately located relative to the sensorimotor system then activation patterns over a network of neurons are meaningful *in themselves*. That is, the activation patterns do not have to be “given meaning” in the same way that symbol-strings do. But surely any connectionist model that is *not* connected to a sensorimotor system *has* to be “given meaning” in much the same way as symbolic systems do: whether you are giving meaning to one discrete symbol, or to a configuration of activations over a number of input or output units, you still have to associate these (sub)symbols with things in the world; you are still the source of meaning.

Lakoff continues:

“In a full-blown connectionist theory of mind activation patterns over neurons are meaningful in

themselves by virtue of what they are connected to. The intractable problem of assigning meanings to symbols does not arise here.

"It is also important to remember that the isolated models connectionists build to study the properties of networks are not full-blown connectionist theories of mind. They vastly oversimplify, or totally ignore, sensorimotor input and output, assuming that, for the purpose of the study at hand, one can just as well use feature names, to which the model-builders must assign meanings. This is a crucial difference between isolated models and a full-blown theory.

"...it is vital to bear in mind that a full-blown connectionist theory of mind is a lot more than just an information-processing system." [21, p.40]

Lakoff's argument that the linking of neural models to a sensorimotor system provides an automatic grounding for the semantics of the model is persuasive. Pursuing this approach prevents theorists from inventing the semantics of patterns of activity (as they are free to while the models remain in their current isolated status). Smolensky's reply to this ("[it] seems an important philosophical point, but one that cannot really do any modeling work until the gap is bridged (...) between the subconceptual [i.e. connectionist] and neural levels..." [27, p.66]) is distinctly *mezza voce*. Again, the attack here isn't *ad hominem*, but Smolensky's views are, I believe, representative of the field. Acceptance of *in vacuo* modeling is widespread within connectionism.

5.5 Summary

Several criticisms of connectionism have been discussed. It has been argued that the connectionist paradigm is biologically *in vacuo* and in this sense is no advance on the symbolic paradigm: connectionism has acted merely as a palliative for several of the maladies of symbolism. The solution to this problem, as identified by Lakoff, is linking the model neural network to the external world via a sensorimotor system. Such an approach is more in line with the philosophy of the Maturana school. Such an approach is embodied in computational neuroethology.

6 Computational Neuroethology

6.1 Towards a definition

Computational Neuroethology (as far as I am aware, I originated this term, in [7]) is proposed here as a method of eliminating the *ad hoc* semantics of contemporary connectionist models. The definition of the term "computational neuroethology" builds on the definition of "neuroethology", given below.

Neuroethology (e.g. [4, 14]) is a young discipline within

the biological sciences: it is where ethology (the study of behaviour) meets neuroscience. Its youth puts it in a situation not dissimilar to cognitive science; its precise aims and methods are still a subject of debate (see e.g. [18, 14]). Put most simply, neuroethology is the study of the neural mechanisms of animal behaviour [18, p.384]. As Clarac remarks, "By definition, neuroethology is a "vertical" science, whose main interest is to link the results obtained from many different levels of complexity in the nervous system." [18, p.384]. That is, neuroethology spans most levels of analysis in neuroscience.

The relevance of neuroethology to the *in vacuo* problem in connectionism is neatly captured by Grossberg:

"Neuroethology teaches us that neural circuits are organised to generate adaptive goal-oriented behaviors. Without a behavioral linkage, no amount of superb neurophysiological experimentation can lead to an understanding of brain design, because this type of work, in isolation, does not probe the functional level on which an organism's behavioral success is defined." [13, p.389].

It is in this respect that connectionist computational neuroscience has most to learn from neuroethology, and it is this belief in the importance of behavioural linkage that most distinguishes computational neuroethology. Computational neuroethology replaces the *in vacuo* approach of connectionism with a (simulated) *in vivo* approach; and in doing so, the semantics of the model are automatically grounded.

So, computational neuroethology can be provisionally defined as the study of neuroethology using the techniques of computational neuroscience. This definition intentionally admits many classes of model, but the significant aspect is the increased attention to the environment that the neural entity is a component of. The vertical nature of neuroethology precludes restriction purely to "simplifying" (connectionist) models: modeling techniques from realistic computational neuroscience should also be applicable to computational neuroethology.

Furthermore, this broad definition is noncommittal on the interpretation of the word "computational": a strict interpretation would allow only neuroethological models that focus on behaviour as a result of computation; but the vertical nature of neuroethology surely indicates that models with no *direct* reference to computation should still be of interest, and the Maturanian anti-pipeline argument outlined above additionally indicates that sole focus on information processing may omit important factors. So "computational" can simply imply a reliance on computerised experimental techniques, as in e.g. computational physics.

And that, for the time being, is the working definition of computational neuroethology. It is a broad definition, but this is a provisional manifesto, and ensuing debate

may refine the definition. What is required is guidance on *how* the connectionist model should be connected to a sensorimotor system, thereby granting the required behavioural linkage. This is a subissue of the field, and is discussed at length in the next section.

As I mentioned above, the term “computational neuroethology” was coined to describe my work in studying visual control of insect flight [7, 8], but plenty of other research in the literature could be classed as work in computational neuroethology.

6.2 The behavioural link

This section discusses my personal attitude towards the issue of providing connectionist models with a behavioural linkage. As such, it is only one of many possible approaches within computational neuroethology.

The need for a behavioural linkage is satisfied by recognising in the model the importance of the environment within which the neural entity is a component. The methods of connectionist computational neuroscience can be adapted to computational neuroethology in a fairly straightforward way: by embedding the network model within a *closed-environment* simulator. A closed environment simulator is one which provides a data-path that models the external feedback loop provided by the environment.

That is, closed-environment models eliminate any “humans in the loop”: current connectionist models have a data-path that resembles the architecture of an open-loop control system; humans are responsible for feeding data (and meaning) into the input end and collecting data (and assigning meaning) at the output end. Closed-environment models are closer in nature to closed-loop control systems: the feedback loop in a closed-loop controller corresponds to the sensorimotor feedback provided by the simulated environment in a closed-environment model. The feedback properties of the environment have long been recognised by (neuro)ethologists.

Closed-environment simulator systems already exist, in the domain of aeroplane flight simulators. Here the neural network is real (and human) but that is not material: motor commands are issued (to the cockpit controls such as joystick and throttle) and the simulator performs the computations necessary on a model environment to generate the sensory input (e.g. dynamic 3-D graphics, etc.) which provides sensorimotor feedback.

Closed-environment simulators for neuroethology can be fashioned in much the same manner. The simulation of the medium thus involves *two* models: the model network and the model environment.

The use of simulated environments is open to the accusation of falling into the micro-world trap, discussed above. Closed-environment simulators are not subject to micro-world problems because they are not “toy” sys-

tems reliant on human interpretation. They are real-world models for a certain definition of real. Consider:

“There is great methodological danger in tackling AI problems in toy worlds. The definition of “toy” here includes all worlds where it is not up to the AI system itself to do all the understanding of the world itself without relying on a human interpreter. Likewise a world is a “toy” world if the AI system is not responsible for carrying out its actions in the world without a human agent to interpret its responses. Such requirements on an AI system thus force it to be part of a robot system acting in a *real world* for some definition of real. This is a much stronger definition of real world than is normally used.” Brooks [3, p.5, original emphasis].

Even under such a definition of real world, it is possible to work on real-world problems using closed-environment simulators rather than robotics (thus ignoring some of the more problematic engineering issues of sensorimotor transduction). An example is the SYCO simulator-model of visual processing in a hoverfly [8].

The advantage in using simulated “real” worlds rather than *the* real world (i.e. a robotics approach) is that the real real world has many superfluous degrees of freedom: many experiments in e.g. visual neuroethology take great care to minimise the dimensionality of the experimental animal’s world by creating visually impoverished environments. Such precautions are also necessary, at least during development and testing, in perceptual robotics; but in simulated “real” worlds *identical* conditions can be recreated as many times as is required.

The closed-environment simulator approach thus provides a behavioural linkage for a simulated neural network. It grounds the semantics of the network in the semantics of the simulated environment. And this is the fundamental point: if the simulated environment is “real” in the above sense, then the model ceases to demand *interpretation*; results are *observed*. That is, discussion of the model network ceases to be *subjective* and becomes *objective*. Soft data is replaced by hard data. It is possible to talk of computational neuroethology as a falsifiable scientific endeavour.

6.3 Finale: Implications

The implications of computational neuroethology for cognitive science are, I believe, more fundamental than simply tightening up a slack discipline. I briefly outline some thoughts below.

The need for a behavioural linkage, and its satisfaction using a closed-environment simulation system, raises a supplementary issue: if there is so much emphasis placed on the preservation of the data-path from sensory input

to motor output, how is it possible to model high-level cognitive functions where there is believed to be significant processing performed between input and output? Surely the sequence of functional units between input and output in complex tasks such as speech understanding or 3D reasoning from vision is so long as to make the modeling task prohibitively complex?

If this problem cannot be solved, at least in principle, then closed-environment simulation is only useful for a very small class of problems: problems which traditionally have been peripheral to cognitive modeling and artificial intelligence.

Fortunately, help is at hand. This problem is resolved by careful attention to two assumptions implicit in its formulation: first, that a sequential functional decomposition of cognitive processes is the best approach; second, that what have traditionally been 'core' topics in cognitive modeling really are core topics, i.e. that problems amenable to closed-environment simulation *really* are peripheral issues. Both of these assumptions have been questioned and found wanting by Brooks.

6.3.1 Subsumption Architecture

Brooks [2] has developed an approach to the design of control architectures for mobile robots operating in the real world, which can readily be adopted for use in the simulated real worlds of closed-environment simulators. The key to Brooks's approach is the concept of a *layered* control system, which leads to control architectures and strategies radically different from convention.

The conventional approach to constructing a mobile robot control system is to decompose the problem into a set of *functional units*, creating an essentially sequential linear data-path which starts at the sensory transducers and ends at the motor actuators.

Brooks proposes that the primary decomposition of the control problem should be into *task achieving behaviours*. That is, he advocates dividing the desired intelligent behaviour of a system into a collection of simpler behaviours; behaviourally complex systems are thus decomposed into a number of computational systems achieving simpler behaviours. The intention is that the simpler systems should be independent, but some degree of overlap is likely in practice.

Brooks organises the behavioural decomposition on the basis of two concepts: task achievement and competence levels. Task achievement dictates that: "Each behavior should achieve some task. I.e., there should be some observable phenomenon in the total behaviour of the system which can be used by an outside observer to say whether the particular sub-behaviour is operating successfully." [3, p.6]. Competence levels specify overall system performance in a rather informal manner: "A set of task achieving behaviours together provide the robot with some level of competence. They should be designed

so that as new task achieving behaviours are added to the system, the level of competence increases." [ibid.].

The different task-achieving behaviours form *layers* of a control system. The layers form what Brooks [2, p.7] calls a *subsumption architecture*: initially, the complete robot control system is constructed to achieve level 0 competence — this is referred to as the zeroth layer control system; once it is completely debugged it is never subsequently altered. Next, the first layer control system is constructed: it can receive input from the layer 0 system, and additionally it can suppress, or *subsume*, the output of the zeroth layer. The zeroth and first layers together implement level 1 competence. Similarly, higher layers are constructed to realise higher levels of competence.

A particular layer always runs unaware of the activities of all higher layers, any of which might interfere with its activity. Each level of competence includes as a subset all earlier levels of competence [2, p.7], so successive layers can be viewed as representing increasingly more sophisticated and constrained classes of allowable behaviour. In Brooks's original robot, each layer is built from a number of asynchronous modules, and each module is a finite state machine [2, p.9].

So the central ideas in the Brooksonian philosophy are: that the control of a mobile robot can be viewed in terms of behaviours rather than functional modules; that the subsumption architecture allows for incremental construction and debugging of complex mobile robot control systems; and that there is no need for centralised control: "The control system can be viewed as a system of agents each busy with their own solipsist world." [2, p.19].

The intention then in closed-environment simulator computational neuroethology is to create a subsumption architecture where the modules forming each layer are not formal (symbolic) finite state automata, but small artificial neural networks. A network model constructed according to a subsumption architecture is an explicit recognition of the inhomogenous nature of natural nervous systems.

A guiding principle in Brooks's work that is particularly relevant to such an approach is:

"Complex (and useful) behavior need not necessarily be a product of an extremely complex control system. Rather, complex behaviour may simply be the reflection of a complex environment. ... It may be an observer who ascribes complexity to an organism — not necessarily its designer." [2, p.3]

6.3.2 Arthropod Imperialism

Moreover, Brooks argues that there are severe limitations on the generally accepted methodologies of artificial intelligence, and that the currently accepted conceptual

decompositions and static representations are wrong. He argues for a shift to process-based modeling, contending that: "...mobility, acute vision, and the ability to carry out survival related tasks in a dynamic environment provide a necessary basis for the development of true intelligence." [3, p.2]. Brooks's argument is based in part on the evolution of intelligent beings: that intellectual capabilities such as writing and 'expert' knowledge are, on the evolutionary timescale, very recent developments indicates to him that the creation of artificial entities truly possessing such abilities will be relatively straightforward "...once the essence of being and reacting are available. ... This part of intelligence is where evolution has concentrated its time — it is much harder." [3, p.1]. Clark makes a related point:

"...intelligence has evolved as a means of satisfying our basic survival requirements. It has *not* been selected for its capacity to achieve the high-level mental feats which so much work in AI is dedicated to modeling. If we *can* perform such feats... it is only *in virtue* of our being endowed with a set of low-level capacities which just happen to facilitate the higher-level activity.

...

"The right microworlds to study are not fragments of the sophisticated human achievements, but the less sophisticated achievements of the various animal intelligences, ranged across the phylogenetic tree." [5, pp.4–5; original emphasis]

In particular, Brooks advocates the construction of robotic insects:

"Insects are not usually thought of as intelligent. However, they are very robust devices. They operate in a dynamic world, carrying out a number of complex tasks... No human-built systems are remotely as reliable... Thus I see insect level behavior as a noble goal for artificial intelligence practitioners. I believe it is closer to the ultimate right track than are the higher level goals now being pursued." [3, p.7].

Similarly evolutionist arguments have been proposed in neuroethology. One of the most passionate advocates of restricting study to arthropods (the animal class to which insects belong) was Hoyle:

"Many neurophysiologists... express no, or very little interest in invertebrates. Their goal is to understand the higher mammalian nervous system, period. ... The invertebrates have nervous systems which will certainly be understood first in cellular and connectivity terms... by definition, the general principles must be those features which vertebrates possess in common with

invertebrates... The study of specifically human nervous systems is an applied, not a general, science." [17, p.17].

From the point of view of neural modeling, a fundamental advantage lies in insect neuroarchitecture. There is ample data in the biology literature demonstrating that insect (and other arthropod) nervous systems are constructed with an economical use of neurons.

The smaller number of nerve cells involved does not necessarily indicate that the neural principles of operation are simpler in insects than in higher animals. However, it does mean that a computational network simulation of insect nervous function is closer in architectural terms to biological reality than any corresponding model of mammalian or primate function, by several orders of magnitude.

So, perhaps the best approach to studying intelligence is, phylogenetically speaking, bottom-up rather than top-down. All previous work in cognitive modeling has focused on advanced animals, near the top of the phylogenetic hierarchy, i.e. humans (and, in vision research, other mammals). Yet the only case for such anthropocentricity is an *a priori* one. Studying intelligence by modeling insects might seem counterintuitive, but then again it might yield some useful results. Perhaps we have been missing something.

Studying animals lower than ourselves is not without precedent in other fields. The most notable example is in genetics, where countless person-years of research have been expended on the fruitfly *Drosophila melanogaster*; with the human genome project commencing only very recently. As John McCarthy (a founder of AI) puts it:

It may be that AI is a problem that will fall to brilliance so that all we lack is one Einstein. [but] I think this is one of the difficult sciences like genetics, and it's conceivable that it could take just as long to get to the bottom of it." (McCarthy, quoted by Johnson [19, p.13]).

Perhaps cognitive science research should go the same way as genetics: we should cease to concentrate on high-level functions and get the basics right first; following a development path strongly influenced by the evolutionary history of natural intelligence.

Even if the study of insects provides us with no data specific to higher animals, it will help clarify the status of the general principles that connectionists search for — perhaps the answer is not to study simple models of large brains, but to study large models of simple brains.

This shift to an antianthropocentric focus for AI has manifest implications for the connectionist study of language. This is a core cognitive function [27, p.62]. But if the arguments listed above are accepted, then (*pace* the researchers involved) all current connectionist models of language are wildly premature. Language will be best

understood as a very high layer in a subsumption architecture: how it interacts with lower layers could be of vital importance, and we should study these lower layers first.

If we are still yet to determine the subsymbolic or neural basis underlying the dance-language of bees, how then are we supposed to study such aspects of human language at anything but the most gross level of neuroanatomy (i.e. studies of lesioned patients)? We simply do not know enough.

7 Conclusion

The significance of computational neuroethology is its spanning of many levels of analysis in the brain and behavioural sciences. The prospects for computational neuroethology look good; it will never answer all of the questions asked of computational neuroscience, but it will provide "simplifying" network models with well-grounded semantics, and hopefully lead to a consequent rejection of interpretation in favour of observation. Subjectivity will give way to objectivity.

The use of closed-environment simulators, coupled with the rejection of the phylogenetically top-down study of intelligence, hails a new approach. This new approach is evolutionist and antianthropocentric. It focuses attention on the true issues underlying intelligence. Language must wait. Discuss.

Acknowledgements

Thanks to Harry Barrow, Inman Harvey, David Hogg, David Pickles, Aaron Sloman, and David Young for their comments on earlier versions of this paper, and to Hilary Tunley for proof-reading. This work was supported by UK SERC studentship grant #87306795.

References

- [1] M. A. Arbib. Schemas and neural networks for sixth generation computing. *Journal of Parallel and Distributed Computing*, 6:185–216, 1989.
- [2] R. A. Brooks. A robust layered control system for a mobile robot. A.I. Memo 864, M.I.T. A.I. Lab, September 1985.
- [3] R. A. Brooks. Achieving artificial intelligence through building robots. A.I. Memo 899, M.I.T. A.I. Lab, May 1986.
- [4] J. M. Camhi. *Neuroethology: Nerve Cells and the Natural Behaviour of Animals*. Sinauer Associates Inc., Sunderland, Mass., 1984.
- [5] A. Clark. Artificial intelligence and the biological factor. CSRP 049, University of Sussex School of Cognitive and Computing Sciences, 1985.
- [6] A. Clark. *Microcognition: philosophy, cognitive science, and parallel distributed processing*. M.I.T. Press — Bradford Books, Cambridge MA, 1989.
- [7] D. T. Cliff. A closed-environment computational network model of visual processing performed by an airborne insect, 1988. Unpublished D.Phil. Research Proposal. University of Sussex School of Cognitive and Computing Sciences.
- [8] D. T. Cliff. The computational hoverfly; a study in computational neuroethology. In *Proceedings SAB90*. M.I.T. Press — Bradford Books, 1990.
- [9] D. T. Cliff. Computational neuroethology: A provisional manifesto. CSRP 162, University of Sussex School of Cognitive and Computing Sciences, May 1990.
- [10] F. Crick. The recent excitement about neural networks. *Nature*, 337:129–132, January 1989.
- [11] D. Dennett. Cognitive wheels: The frame problem of AI. In C. Hookway, editor, *Minds, Machines, and Evolution*. Cambridge University Press, Cambridge, 1984.
- [12] H. L. Dreyfus. From micro-worlds to knowledge representation: AI at an impasse. In J. Haugeland, editor, *Mind Design: Philosophy, Psychology, Artificial Intelligence*, pages 161–204. M.I.T. Press — Bradford Books, Cambridge MA, 1981.
- [13] S. Grossberg. Neuroethology and theoretical neurobiology. *The Behavioral and Brain Sciences*, 7(3):388–390, 1984.
- [14] D. M. Guthrie, editor. *Aims and Methods in Neuroethology*. Manchester University Press, Manchester, 1987.
- [15] S. Harnad. The symbol grounding problem. In *CNLS Conference on Emergent Computation*, Los Alamos, May 1989. Submitted to *Physica D*.
- [16] A. L. Hodgkin and A. F. Huxley. A quantitative description of membrane current and its application to conduction and excitation in nerve. *J. Physiol.*, 117:500–544, 1952.
- [17] G. Hoyle. Neural mechanisms underlying behaviour of invertebrates. In M. S. Gazzaniga and C. Blake-more, editors, *Handbook of Psychobiology*, pages 3–48. Academic Press, New York, 1975.
- [18] G. Hoyle. The scope of neuroethology. *The Behavioral and Brain Sciences*, 7:367–412, 1984.
- [19] G. Johnson. *Machinery of the Mind*. Microsoft Press — Tempus Books, Redmond, Washington, 1986.

- [20] C. Koch and I. Segev, editors. *Methods in Neuronal Modeling: From Synapses to Networks*. M.I.T. Press — Bradford Books, Cambridge MA, 1989.
- [21] G. Lakoff. Smolensky, semantics, and the sensori-motor system. *The Behavioral and Brain Sciences*, 11(1):39–40, 1988.
- [22] S. R. Lehky and T. J. Sejnowski. Simplifying network models of binocular rivalry and shape-from-shading. In C. Koch and I. Segev, editors, *Methods in Neuronal Modeling: from Synapses to Networks*, pages 361–396. M.I.T. Press — Bradford Books, Cambridge MA, 1989.
- [23] J. L. McClelland and D. E. Rumelhart, editors. *Parallel Distributed Processing. Volume 2: Psychological and biological models*. M.I.T. Press — Bradford Books, Cambridge MA, 1988.
- [24] F. Ratliff. *Studies on Excitation and Inhibition in the Retina*. Rockefeller University Press, New York, 1974.
- [25] D. E. Rumelhart and J. L. McClelland, editors. *Parallel Distributed Processing, Volume 1: Foundations*. M.I.T. Press — Bradford Books, Cambridge MA, 1986.
- [26] T. J. Sejnowski, C. Koch, and P. S. Churchland. Computational neuroscience. *Science*, 241:1299–1306, September 1988.
- [27] P. Smolensky. On the proper treatment of connectionism. *The Behavioral and Brain Sciences*, 11:1–74, 1988.
- [28] T. Winograd and F. Flores. *Understanding Computers and Cognition: A New Foundation for Design*. Addison-Wesley Publishing Co. Inc., 1986.

ON THE FEASIBILITY OF COMPUTATIONAL ARTIFICIAL LIFE
A REPLY TO CRITICS

Harold C. Morris
P.O. Box 657
Los Alamos, NM
87544 USA

Abstract

The thrust of this paper is to meet the objections of what may be termed a philosophical school, whose principals are Robert Rosen, Howard Pattee, and Peter Cariani. The objection that a computational universe is a flat "pseudo-world," because it is "all syntax and no semantics," is inquired into and refuted, as is the claim that nothing really new can evolve within such an artificial universe. It is concluded that no persuasive reasons have been advanced as to why computational artificial life is not feasible.

A convergence of several fields has resulted in the new discipline of Artificial Life ("AL") research. And just as AL science has several sources, it is moving in several distinct directions. Life-like entities are being developed as biochem-

ical "wetware," robotic hardware, and as computer software. This paper shall, however, deal only with controversies surrounding the third variety, namely the computational AL form.

Generally stated, the AL program is to develop life-like organisms in the medium of choice. For myself and some other AL researchers, the computer is our medium of choice. Our objective is to implant or evolve individuals or colonies in automaton universes, to observe instances of propagation, adaptation, or communication, such as one usually associates with life forms.

Since the pioneering work of von Neumann (1966), cellular automata have been much used as computational media for AL research. In recent years, more sophisticated systems, for example

the ComMet Computational Metabolism of Marek Lugowski (1989), have expanded on the basic cellular automaton theme.

A computational medium along these lines is my own Robotix World (Morris, 1990). In Robotix World, virtual robots rove over a grid, getting from fixed dispensers chunks of information, "modules," containing instructions written in the robots' programming language. The acquiring robot may then assimilate this module into its own internal program, to enact the instructions subsequently. Alternatively, it may stack such modules atop each other to build a brand new robot. These robots are thus self-programming, and propagating. Since robots can steal from or cannibalize modules off other robots, they can compete quite directly with one another. Robots can build other robots for later consumption or service, as we husband plants and animals. Since robots can receive instructional modules from other robots, and set down modules for future generations to discover, they are capable of communicating

messages, directing the actions of others, and of leaving maps or other records for the edification of posterity. This system would seem to have most of the potentialities desirable in an AL universe.

Nevertheless, there are "naysayers" who would have us believe that nothing really deserving of the appellation "living" could arise in or inhabit any computational medium. One of these critics, Robert Rosen, has very recently stated that, in spite of all our contemporary efforts and advances, "we are in fact not an inch closer" to the fabrication of an organism than were our predecessors (Rosen, 1990). With colleagues Howard Pattee and Peter Cariani, Rosen is especially vehement in his denial that computational approaches to artificial life hold any promise of success. Let us proceed to their objections.

The first of these, expressed by Howard Pattee (1989), essentially says that a simulation necessarily refers to something outside itself and by definition cannot become something over

and above a merely referential entity. A paper dragon can never become a real dragon. A simulation of a dairy cow can never yield a drinkable beverage, no matter how vivid or accurate a model it is.

Pattee has been moved to make this objection on noting the loose terminology of some AL researchers, who do speak of simulating life. However, let's be clear on one thing: There must be grounds independent of what a researcher says he is doing, in deciding what he has created. It makes no more sense to say that something cannot, by definition, be alive because it has been (perhaps carelessly) termed a simulation, than it makes sense to say that something must be alive, because it has been termed a realization of life.

Actually, there is no reason why an automaton cannot be simultaneously modelling some possibility for life, while also realizing a new type of organism. By close analogy, an artificially engineered crystal is a crystallographic model, i.e. does display a possible structure of atoms, even while it is also an artifact, something new under

the sun. Of course, a geologist may not be interested in this new crystal as an artifact, but he can hardly afford to ignore it as a possible arrangement that atoms may take. And by the same token, a (natural) biologist may not consider an artifact to be within the purview of his field, but should not ignore those possibilities of propagation, adaptation, etc., brought to light by the computational AL researcher.

The idea that the same program may be treated on the one hand as a referential model, and on the other hand as a realization, should not be any more disturbing to one's sensibilities than is our tendency, for example, to treat a Voyager communicated photograph of Neptune as, on the one hand, an illustration of an aspect of nature, to be judged by its fidelity to the original, and on the other, as an aesthetic object, to be judged on its aesthetic merits.

1. The Measurement Problem!

A seemingly more substantive criticism of the Rosen/Pattee/Cariani school involves what Pattee terms the "incomprehensible" measurement problem. Members of this school hold that a living thing must be able to make measurements, but that measurement is a process that cannot be simulated by physical processes alone, and cannot be realized computationally at all. Cariani (1989) has made much of this.

It is easy to agree that we should like for our AL forms to have the capacity to make at least rudimentary observations, or to have something on the order of the venus fly-trap plant's sensitivity.

However, the "measurement problem" Cariani et al. have in mind has been borrowed from quantum physics, and as far as this author is concerned no computational AL researcher is seriously bent upon creating little quantum physicists.

Carpetlayers, and other scientists for that matter, do manage to make measurements without having to cope with mind-boggling philosophical obstacles. Cariani et al. must

carry a heavy burden of proof in establishing the direct relevance of the measurement problem, as it is known in quantum physics, to the AL enterprise. Certainly nothing in Pattee's or Cariani's writings stands as sound objection to the creation of a computational entity with the sensitivity of a venus fly trap.

Related to this "measurement problem" is a directly relevant logical problem associated with a part observing the whole of a system it inhabits. To frame the problem in the form of a diagonal argument: We may infer that for an artificial life form or organ existing within a larger environment to observe all the parts (of that universe) that cannot observe themselves, is to do the impossible, as one realizes on considering the matter of self-observation (this being, of course, a variant on the Russell Paradox).

One can also formulate an infinite regress problem. To observe the entire universe is to entertain some model internally that corresponds to what is observed. Again, self-observation becomes problematic, in that it involves making a

model of a model, ad infinitum, of the self. Rucker (1983) considers a closely analogous paradox, which he attributes to turn of the century philosopher Josiah Royce. Rucker suggests a possible "out" for the human mind infinitely introspecting, by hypothesizing that the brain may be itself infinitely divisible. Clearly this will not work, though, for a digital computer, which has a "lowest" level of discrete components.

At any rate, though these philosophically interesting conundrums may establish some limitations on the capacity of an AL form to arrive at an absolutely all-encompassing knowledge of its environment, all the life forms with which we are acquainted in nature are similarly limited. Such limitations can hardly be taken as decisive on the question of whether something is alive or not! And the modest, practical objective of obtaining "sensitive" organs or organisms, perhaps even entities that can recognize some patterns occurring in the world they occupy (or represent those patterns, however crudely), is not shown infeasible by these

types of arguments.

Related to the above, Cariani (1989) has set as a requirement for an artificial life form, that it construct for itself a new "device for measurement."

Now, only human beings, among known natural life forms, create new measuring devices for themselves, in what I take to be customary usage of the terms "create for themselves" and "measuring devices." What Cariani must mean is that he would like to see a genealogical line evolve a new kind of sense organ. As far as that goes, I think many AL researchers would like to work toward realizing just such a prospect. The only question is whether this is a good criterion for admitting something to be a life form or not.

To be sure, this may come down to one's intuitions. Suppose a form evolves within a computational medium that can somehow sense the general excitation level of its immediate surroundings, as we sense heat and hear air vibrations as sounds. Ignore the fact that Cariani et al. believe that evolutionary emergence cannot occur in a computational environment, since that objection will

be dealt with shortly. Then suppose that I, the AL researcher, having observed with delight the appearance of this new sensory capacity, figure out how it works, and imbue other creatures with it.

Why should we think that the latter creatures, those which had the sensory capacity built in by the experimenter, are less "entitled" to being called living (or life-like), than those creatures that acquired it adventitiously. My intuitions insist that the main thing is whether the entity has the capacity or not.

Of course, in the somewhat parallel debate over artificial intelligence, some skeptics simply rule out mathematical calculating ability as evidence of machine intelligence, since, "after all, the machine was built to be a computer." Perhaps this same attitude will prevail in AL. One undesirable pragmatic consequence of this will be a temptation to "fudge," however. Again, referring to the "thermal" sensitivity example, the AL researcher who has evolved something with this sense may be reluctant to admit that the built the capacity into subsequent AL

forms, if he knows that a certain ideology discounts such pre-designing of sensory capacities.

2. A Semantic Void

Pattee and Cariani view a computational medium as intrinsically, irredeemably sterile, because it is, they claim, all syntax and no semantics. Cariani alludes to the semiotics of Charles Morris, in defining syntax as the rules governing a language or formal system, and semantics as the relationships holding between a language and the world. According to Cariani, a computational universe is, so to speak, a flat pseudo-world supporting no real semantics. Symbols are shuttled about lawfully but meaninglessly; lacking real world efficacy or reference, they can never really amount to a living thing in a real environment. Cariani evidently pictures the fruit of computational AL research as being something like a genetic blueprint for which no epigenetic rules exist, a blueprint for something that possesses sense organs which have nothing available to sense, in short, a mere string among strings.

Actually, there is just no reason to think of a computational

universe as necessarily isolated from the world beyond the computer. Many if not most of us doing computational AL work at least entertain the possibility of allowing input into our computational universe from peripheral sources. However, I will address the case Cariani feels his objection most applies to, namely the self-contained computational medium that does not receive input from the external world.

Cariani's misconception lies in his understanding of semantics, and specifically his forgetting that a language may be an important part of the world it talks about. Thus a language may talk about itself. Quine invented his use/mention distinction to allow us to conceptualize the situation clearly. A syntactical configuration may consist in part of mention of syntactical entities, facts, or rules. And once such embedding is permitted, it can lead to indefinitely rich nestings of mention. The essential point is that a rich language can have semantics even without reference to anything but itself.

Granting that at one level

of description, an artificial life form, say a virtual robot or bug or fish or what have you, is a syntactical string or configuration, that is not to say that it is semantically empty, for substrings within it may be instances of mention of other strings external to this entity but within the computational universe. In other words, reference is perfectly possible. The outside observer (e.g. the AL researcher) can then make interpretations of these instances of mention in terms of sensations, or memories, or even thoughts, entertained by the AL form.

3. Evolvability

Yet another contention put by Cariani and Pattee is the requirement that life forms be "evolvable." This contention is meant to go hand in hand with their contention that genuine evolution is impossible within a computational medium. Obviously, these two contentions add up to the proposition that artificial life cannot arise within a computational universe.

It actually seems a bit absurd to speak of "evolvability" as an affirmative property possessed by life forms. Both

animate and inanimate things change under environmental pressure. It is precisely life's relative resistance to change, achieved through adaptation and self-replication, that distinguishes it from the nonliving. Affirmative capacities of life such as these are in fact under intensive study by AL science today.

Rather than speak of "evolvability" as a criterion for identifying life forms, then, let us speak instead of the evolvability of an artificial universe. I am confident that most of us in computational AL believe that we have realized evolutionary processes within computer media. Let us see why Cariani and Pattee object to our characterizations of these processes (and their fruits).

Cariani (1989) begins with the inarguable premiss that a computational medium can be described at one level as a formal system, with a finite alphabet of primitive symbols and a fixed set of rules of inference/substitution. But he then leaps to the conclusion that this means that nothing really new can arise in such a system, because no

new primitive symbols can arise-- only new combinations of the given primitives.

The fallacious jump here is in thinking that an artificial life form would be a new primitive symbol, or necessarily be composed at least in part of such new primitive symbols. There is no reason to think this. An organism or organ arising in a computational medium's evolution will be a new combination of the given primitive symbols. That's what "something new" is: a combination not present at the starting up of the evolutionary process. In most systems, the potential number of new combinations is going to be infinite.

The bad reasoning here is quickly exposed by some analogies. Would we tell an aspiring composer that he cannot possibly come up with any new, original musical compositions, because he's using a finite number of scales, each of which has a finite number of notes(i.e. primitive symbols)? A new musical composition is just a new combination, isn't it? And would we imagine that a child who has learned the English alphabet now "knows" all the works of literature that have been or ever will be written in that

language, because, after all, those works of literature are only combinations of the same old primitives? As for the natural universe itself, are life forms unable to arise in evolutionary systems that have only finite types of elementary particles (e.g. kinds of quarks)? Is it absurd to refer to an artificial compound as "new" if it is, after all, only another combination of the same old elements?

I have endeavored in this paper to refute certain objections posed against the computational artificial life program, by Drs Rosen, Pattee, and Cariani. I wish to conclude with words of rapprochement.

First and most generally, I hope that all philosophers and scientists are aware of the value of criticism. We move forward blindly if we move forward with unchecked and unchallenged assumptions and an attitude of infallibility. The critics to whom I have made rebuttal here are to be thanked for their sincere efforts to make us in computational AL keep our feet on the ground and headed

down the right path. Certainly, in trying to meet their objections, I have been forced to think more clearly and generally about what we are about in this field, and I am thus indebted to these critics.

Finally, I would remind us all that refutations of objections raised against the feasibility of computational artificial life do not have affirmative force in establishing the feasibility of such "software life forms." For that, we must continue to strive to obtain the kind of impressive results that will most effectively win over the skeptics and agnostics. It may be hoped that this conference will be looked back upon as a milestone in the demonstration of such results.

Bibliography

Cariani, Peter. On the Design of Devices with Emergent Semantic Functions.

Unpublished dissertation, State University of New York at Binghamton, 1989.

Lugowski, Marek. "Computational Metabolism," in Artificial Life, C. Langton, Ed. Addison-Wesley, New York, 1989.

Morris, Harold. Formal Sciences Approaches to Artificial Life. Book manuscript on submission, 1990.

Pattee, Howard. "The Measurement Problem in Artificial World Models," paper delivered at first international workshop on artificial life, Los Alamos National Laboratory, 1987.

Pattee, Howard. "Simulations, Realizations, and Theories of Life," in Artificial Life, C. Langton, Ed. Addison-Wesley, New York, 1989.

Rosen, Robert. "What Does It Take to Make an

Organism?" Paper delivered at second international workshop on artificial life, Santa Fe Institute, 1990.

Rucker, Rudy. Infinity and the Mind. Bantam, New York, 1983.

The Animat and the Physician

Alexandre Parodi

Indodyne SARL
10, rue de la Paix
75002 Paris
Fax: (33) (1) 40 20 98 98
Tel: (33) (1) 69 81 88 41
E-mail: (uucp) parodi@lri.lri.fr

Pierre Bonelli (LRI)

Equipe I&A
LRI, bâtiment 490
Domaine Universitaire de Paris-Sud
91405 Orsay cedex
Tel: (33) (1) 69 41 64 07
E-mail: (uucp) bonelli@lri.lri.fr

Abstract

Animat research has already produced interesting concepts and algorithms. In this paper, we analyze how this research can be applied to human intelligence understanding and to reproducing of some expert behaviors. To support these ideas, we experiment with an improvement of Boole, a Genetic Based Learning algorithm from animat research, in a medical domain of expertise. We experimentally demonstrate that our system obtains good results on a well known realistic medical diagnosis task, and we analyze its potential ability to solve more complicated problems.

Introduction

There has been much debate about how one can consider that a system is intelligent, most of the time according to how it processes information (rules, neurons etc.) in connection with the human brain. However, in [Wilson, 85] Wilson developed the idea that we could probably learn more from ethology, and he introduced the concept of animats, which are autonomous systems which learn how to survive and expand in a given environment.

We propose to discuss how this research can be profitable to the understanding of human intelligence, and how animat algorithms can be used to reproduce some intelligent human behavior.

1. Intelligence hierarchy

Let us consider the following intelligent systems hierarchy based on how explicit the input knowledge from the environment must be: systems that learn by

being told, systems that learn by complete examples, and animats, i.e. systems that learn by reward.

1st level: Systems that learn by being told

Most computers get their knowledge by being given programs, i.e. a list of instructions to be executed in a specified order; the processing is completely explicit in the input knowledge.

Production systems without learning ability get their knowledge from rules which are used to reason about the input data and conclude about the output data to provide. They are somehow more intelligent because the order in which rules are executed depends on the data: control is data driven. This means that a few rules implicitly specify many different reasoning traces.

Specifically built neural networks get their knowledge from a set of predetermined weights which indicate how a formulated hypothesis (micro-feature represented by one neuron) influences another. However, without learning, they are not really much more intelligent than production systems, but some experiments tend to show that they are less brittle and noise sensitive and can exhibit even richer behavior than can be expected because they use parallel analog processing.

2nd level: Systems that learn by complete examples

Learning systems can manage with even less formalized knowledge: they only need examples which contain both the input and the corresponding desired output; they take care of extracting the appropriate knowledge that is needed to generalize the sampled behavior to new inputs. Such systems can be either rule based systems, connectionist networks, or classifier systems. A number of learning algorithms

exist for each of these, and there is much debate about whether connectionist systems [Bounds, 89], rule based systems [Kodratoff, 89], homogeneous hybrid systems [Parodi & Khouas 89], or inhomogeneous hybrid systems [Hendler, 88] should be used.

3rd level: Animats. Systems that learn by reward

However, according to our hierarchy, animats are even more intelligent since their only source of knowledge is the reward (payoff or penalty) that they receive from the environment according to their behavior in each situation.

Therefore, they are not given explicit knowledge about the adapted behavior corresponding to each encountered situation: they have to infer it from past experience, and to generalize it for new situations.

In addition, they cannot be given all their knowledge at once: they need to learn it *incrementally*, and they cannot memorize it as it is: they have to extract and memorize useful knowledge which is diluted into massive, diffuse, inaccurate and inconsistent information.

It appears that this kind of intelligence is needed by autonomous systems in nature, such as animals, animal societies, or animal species which all learn to survive in a hostile environment by being given more or less fulfillment of their basic needs, expansion of their territories, or their total population.

It seems at the present time, that Genetics Based Machine Learning is best suited to fulfill these needs.

2. Animat and human intelligence

How can AI benefit from research in the animat domain ? This is the question we will try to answer in this section.

For some human activities, reasoning can be well conceptualized because people have access to complete consistent knowledge about the domain; additionally, the associated reasoning is conscious. In these cases explicit models of both knowledge (rules) and reasoning (meta-rules) can be build and permit expert systems (which "learn by being told") to be effective in behaving like humans for these tasks.

This tends to be true for instance in the case of computer configuration, machine trouble checking and diagnosis, etc., and in a number of man created domains.

However, in some other domains, man cannot really conceptualize the needed knowledge because it is not available or beyond his understanding.

A typical case is medical diagnosis where physicians interact with the human body which is extremely complex, not yet completely understood, and where input symptom data has a complicated relationship with the diagnostic output. Of course, academic (conceptualized) knowledge helps, but doctors claim that past experience (set of examples) is most important: it develops the intuition capability.

In this sense, acquiring the intuition that is needed for medical diagnosis is merely learning by examples.

However, a system which does medical work and behaves as a physician has to be more than a simple "learning by examples" system. Indeed, the information is very diffuse, and doctors induct knowledge from it incrementally during a very long period of time, with no specific memorization of each and every case, as is needed for animats.

In addition, medical researchers also design therapies which involve new treatments and drugs against diseases that are not yet very well understood: they are not given examples that contain what treatment to give for an inferred diagnostic, or a given set of symptoms.

They rather guess and try, using their incomplete present knowledge to treat a patient, and then check whether the patient evolves toward a better or worse condition; from this point they update their knowledge iteratively. Therefore, improving therapies might be considered as learning by reward; this last quantity is directly related to the patient's condition improvement.

It clearly appears that this last activity is close to those performed by animats.

Hence, when attempting to build a machine which behaves as a physician, it seems quite natural to use genetic algorithms to learn the medical "intuition" for both diagnosis and therapy. However, one might say that classifier systems mechanisms do not have much to do with the human brain, neither at the conscious reasoning level nor at the neural one. However, "intuition" merely consists in being able to "feel" the correct solution unconsciously, sometimes with no possibility to explain or even justify it absolutely. Since modeling is not possible, our approach is "behaviorist"; i.e. we do not care if knowledge is coded inside our machine in the same (unknown) way as in a human being who has the desired behavior. Hence, the idea of identifying the animat problem and the doctor's problem and use a genetic algorithm to perform learning in this context is sound.

3. Experimental program

In the context of this paper, we restricted our experiments to a medical diagnosis task, which seems

a natural step toward therapy learning. In addition, the data base with which we experiment has been tested with a number of learning systems, which allows to validate our ideas quantitatively.

We therefore adapted Boole [Wilson 87] a classifier system that emerged from animat research to a medical diagnosis problem.

We modified it in a way so that it gathers its knowledges in the form of 0^+ order production rules: this allows some expert checking, but also human use for knowledge that cannot be easily acquired by humans such as therapy.

4. Quick overview of the NEWBOOLE system and its improvements.

In this section, we briefly describe the NEWBOOLE Classifier System used to perform the medical diagnosis task described in the experimental section of the paper.

NEWBOOLE is a modified version of Wilson's system BOOLE, which was initially designed to fulfill the animat problem constraints.

* BOOLE is a simplified version of the standard Classifier System (CS) which learns difficult boolean functions.

Like any CS, Boole maintains a population of classifiers (which can be thought of as bit-level zero order rules) according to Darwinian evolution principles. However, classifiers are not chained; they directly provide an output and the decision is made within a single step during recognition; consequently there is no message list nor Bucket Brigade Algorithm. Thus each classifier consists of a condition (taxon) and an action which are fixed length strings over the $\{0,1,\#\}$ alphabet.

Like other CS, BOOLE contains three components, namely a performance component, a reinforcement component, and a discovery component.

* NEWBOOLE integrates changes to both the performance and reinforcement components, which were made in order to get a faster convergence of the error rate in the context of learning from examples.

The changes resulted in the following differences with BOOLE:

1/ General output:

NEWBOOLE is no longer a specialized CS for Boolean functions: the number of classes can exceed 2.

2/ Learning rate:

NEWBOOLE obtains much faster learning rates than its predecessor, partly through the use of a new

reinforcement component that we developed jointly with Wilson and Sen.

It is important to mention that BOOLE's generalization capabilities were maintained in the new system. For more details about the NEWBOOLE system, please refer to [Bonelli, Parodi, Sen & Wilson, 90].

3/ Initialization:

For each experiment, the classifier population was initialized by directly applying a modified version of Wilson's create operator.

4/ Rule crossover:

In addition, for the following experiments, NEWBOOLE has been improved by enhancing the crossover component of the genetic algorithm.

Indeed, the knowledge of the system is appropriately represented by a set of explicit 0^+ order rules rather than a set of strings of bits as is the case with classifier systems. Each rule consists of a condition part, which is an *unordered set* of (attribute, value) couples, and a conclusion part, which provides the output value.

Therefore, the crossover that is used takes into account general properties of this representation. More generally, our approach consists in directly apply modified genetic algorithms onto the appropriate representation of the knowledge, rather than attempting to convert the knowledge into a classifier representation which can be processed through classical genetic algorithms.

Our first results tend to show that this feature further enhances the generalization quality, i.e. the ability to find appropriate answers for examples that the system has never seen before.

The error rate on such examples is reduced by about one third, hence providing results which are better than ID3, and identical to neural networks using back-propagation, Michalski's logic reduction system or statistical Bayesian systems.

5. Description of the algorithms

In this section, we describe the algorithms that were used in the experiments.

5.1. The BOOLE Classifier System

We now present BOOLE with some detail concerning the parts that have been changed, in order to explain the improvements in the following paragraphs. Boole includes three components:

1/ *Performance component*: in the performance cycle, an input string is presented to the system, the match set M of all classifiers whose taxa match the input string is formed, and a single classifier from M is selected (using a probability that is proportional to its strength) whose action is output as the system's decision.

2/ *Reinforcement component*: this component modifies the strengths of classifiers according to performance level:

a/ Form the action set [A] consisting of classifiers from [M] whose action is the same as the chosen action; the remaining members of [M] form the set Not[A];

b/ Deduct a fraction e from the strengths of all classifiers in [A];

c/ * If the system's decision was correct, distribute a payoff quantity R to the strengths of [A]; but

* If the decision was wrong, distribute a payoff quantity R' (where $0 \leq R' < R$) to the strengths of [A] and deduct a fraction p from the strengths of [A] (at least one of R' and p is equal to 0);

d/ Deduct a fraction t from the strengths of Not [A].

The distribution of payoff is done so that rules which have many # 's (thus more general) are favored.

3/ *Discovery component*, which modifies the classifier population according to Holland's (1975) genetic algorithm and employs reproduction, genetic operations (crossover and mutation), and deletion.

BOOLE's version of the genetic algorithm is quite particular in the sense that only one offspring is added per invocation of the genetic algorithm. In this context, the parameter p will represent the average number of invocations of the genetic algorithm per cycle (i.e. the number of offspring added per cycle). For more details, please see [Wilson, 1987].

Here $p = 4$.

5.2 Reinforcement component enhancement

BOOLE's reinforcement component, under the "payoff-penalty" reinforcement regime ($p \neq 0$) adjusts classifier strengths in the following way:

- if the system's decision is correct, distribute a quantity R to the strengths of the Actionset [A].

- if the system's decision is false, penalize the strengths of [A] by deducting a fraction p from their values.

- finally, whether the system's decision is correct or not, deduct fractions e and t respectively from the strengths of [A] and Not[A].

Thus, following each performance cycle, *only the strengths of [A] are adjusted according to the system's performance*.

However, once we know that [A] contains accurate classifiers, we also know that Not[A] only contains inaccurate classifiers; in this case it would make sense to penalize the rules in Not[A]. This acknowledgement led us to a *symmetrical payoff-penalty* algorithm, in which we respectively reward and penalize the accurate and inaccurate classifiers present in the Matchset.

The new reinforcement component is the following:

1/ Form the subset of [M] consisting of those classifiers whose action is accurate; this is the correct set [C]. The remaining members of [M] form the set NOT[C].

2/ Deduct a fraction e from the strengths of [C].

3/ Since [C] contains the accurate classifiers, distribute a payoff quantity R to the strengths of [C].

4/ Since Not[C] contains the inaccurate classifiers, deduct a fraction p from the strengths of Not[C].

Thus, the effect of the reinforcement component can be written as:

$$S_{[C]}(t+1) = (1-e) \times S_{[C]}(t) + R \quad (1)$$

$$S_{\text{Not}[C]}(t+1) = (1-p) \times S_{\text{Not}[C]}(t) \quad (2)$$

where $S_{[C]}$ and $S_{\text{Not}[C]}$ are respectively [C]'s and Not[C]'s total strengths.

This new algorithm constitutes a clear departure from Boole: indeed, if we have several possible output values then the knowledge of the correctness of the output of each classifier from the match set is used. This information can be provided by the knowledge of the correct output for each example, as is done in most learning systems. However, this does not make any difference with boolean functions such as the Multiplexer since only two values are possible: if one is known as wrong, then the other one is right.

As in Boole, the payoff R to [C] is distributed by a biased distribution function D, which

favors more general rules (i.e. with many "don't cares" #) as follows.
First, the generality of each classifier n° i of length L is computed as:

$$g_i = \frac{\text{number of \#s in } i}{L} \quad (3)$$

Let us define:

$$d_i = 1 + G \times g_i \quad (4)$$

where G is a "generality emphasis" parameter.

Then, the portion of reward R_i that is given to classifier i becomes:

$$R_i = D(i) \times R = \frac{d_i}{\sum_i d_i} R \quad (5)$$

In addition, some the following improvements were made for the experiments which are considered in this article.

5.3 Initialization algorithm

The reason for choosing a different initialization procedure is that the search space to be explored is far too vast to permit a random initialization process since the population classifiers are supposed to recognize examples from the Learning Base which represents a microscopic fraction of the search space.

Therefore, the initial classifiers are generated in the following way:

a) An example from the learning base is chosen at random.

b) The condition part of the new classifier is a copy of the symptomatic part of the example, except that "don't care" symbols are inserted with a probability of 0.5.

c) The conclusion part of the new classifier is chosen randomly among the 4 possible diagnoses.

d) Steps a), b), c) are repeated P times, where P is the size of the population.

5.4 Symbolic discovery

In this section we describe how we address symbolic processing in our system, in order to use symbolic data, as is needed for medical work.

. Usual representation

We assume that the examples are described with a collection of attributes. Each attribute can take a small set of discrete mutually exclusive values. For instance, in the weather domain, we consider four attributes {SKY, TEMP, WIND, OUTLOOK} which can respectively take the following values:

SKY: {blue, grey, rainy};
TEMP: {cold, mild, warm, hot};
WINDY {true, false};
OUTLOOK: {mistral, east wind}.

A weather description example might be:

$$(SKY = \text{blue}) \& (TEMP = \text{cold}) \& (WINDY = \text{true}). \quad (6)$$

In this case, the following rule would fire:

$$(SKY = \text{blue}) \& (WINDY = \text{true}) \rightarrow (OUTLOOK = \text{mistral}) \quad (7)$$

The straight forward GA coding of such a description would consist in a mapped concatenation of fixed length binary strings, each of them representing an attribute's value. In the same weather domain, the attributes values could be coded in binary as:

SKY: (blue = 00, cloudy = 01, grey = 10, rainy = 11);
TEMP: (cold = 00, mild = 01, warm = 10, hot = 11);
WINDY: (false = 0, true = 1);
OUTLOOK: (mistral = 0, east wind = 1).

The same weather description (6) would then be represented by the data :

$$00001 \quad (8)$$

and the classifier corresponding to rule (6) is :

$$00\# \# 1 / 0 \quad (9)$$

The problem here is to check whether we could apply GBML directly on the attribute-value representation without preliminary coding.

Firstly, the number of attributes to be treated for each rule would be smaller than the number of bits in the corresponding CS classifier, resulting in faster processing.

Moreover, classifiers have many explicit don't care positions, which must be processed, resulting in a waste. Using the attribute-value representation, attributes whose values do not have any incidence in the conclusion of a rule will not be present in its premise, and will not be explicitly processed.

Set representation

The way rules are coded does change the matching algorithm which is then close to what is used in O^+ expert systems, but the reinforcement algorithm does not have to be changed at all.

However, it is obvious that the discovery component will have to be modified in order to make it work on this symbolic representation.

Since a condition is actually a conjunction and because the logical "AND" is commutative, we can note that it actually is a variable size unordered set of (attribute, value) couples; for instance the logic condition:

(SKY=blue) & (TEMP=cold) & (WINDY=true) (10)

can be represented as the unordered set of three couples:

{(SKY,blue), (TEMP,cold), (WINDY,true)} (11)

We developed an algorithm which exploits this representation by performing crossover and mutation directly on these sets without any conceptual reference to strings. This algorithm will be presented in an upcoming article.

6. Experiments and discussion of results.

We tested NEWBOOLE's ability to perform medical diagnosis using data on the lymphography domain, provided by the Institute of Oncology in Ljubljana, Yugoslavia. We chose this particular set of data because numerous other inductive learning systems have been applied to it in the past, hence enabling quantitative comparisons.

This example base consists of 148 examples separated into 4 distinct diagnoses. Each example is completely described by 18 discretely valued attributes representing the symptoms and one of 4 classes representing the corresponding diagnosis.

6.1 Conditions of the experiments

Our main goal was to test the quality of the generalization, i.e. the induction capability.

For each experiment, 70% of the examples (i.e. 103 examples) were randomly chosen for learning, and the remaining 30 % (i.e. 45 examples) for testing. Hence the system is tested with examples it has never seen, thus enabling quantitative measurement of its performance for induction.

During each learning cycle, an example from the learning base is chosen at random and presented as input to the system. The system then makes its decision according to its past experience and applies the reinforcement and discovery components according to the experimental parameters.

Learning was performed over 10,000 cycles to ensure that proper generalization has been achieved. Indeed, even after the error rate on the learning data base has converged to almost zero, the system still improves its generalization capability.

6.2 Experimental Results

Results below are the average of 4 independent experiments. Figure 1 shows Symbolic Newboole's performance over the Test Base, i.e. its induction ability. The graph plots the smoothed system's *average score* which is the percentage of correct decisions over the past 100 cycles versus the number of cycles since the experiment began.

This plot shows that the system more or less stabilizes its performance after a few thousand cycles at approximately 82 %.

The average result for the test experiment was 37 right answers out of 45, which represents approximately 82 % of correct classifications. This score is to be compared with an estimate of 85 % of correct diagnoses by specialists, 76 % with NewBoole without enhanced crossover, and 76% for assistant-86 which is an ID3 based learning system.

We provide here after a table which compares these results in induction with other learning systems which are used in the same conditions.

Of course, these results have to be considered with the idea that they are the average of four independent estimations of probabilities of around 80 % with 47 samples each. Hence, we get a mean square root error of about 2.5 %. Therefore, the differences between Bayes, BP and Newboole with rule crossover are insignificant here.

6.3 Discussion of the results

These results clearly validate our intuition that a Classifier System can provide high classification rates in a domain such as medical diagnosis.

This is amplified by the fact that no other inductive, incremental learning system obtains such results besides Connectionist Systems. However, Classifier Systems have the advantage of being equivalent to O^+ rule based systems which provide comprehensive solutions, and thus contain an explanation capacity, whereas neural networks merely provide sets of numerical coefficients without any clear semantic meaning.

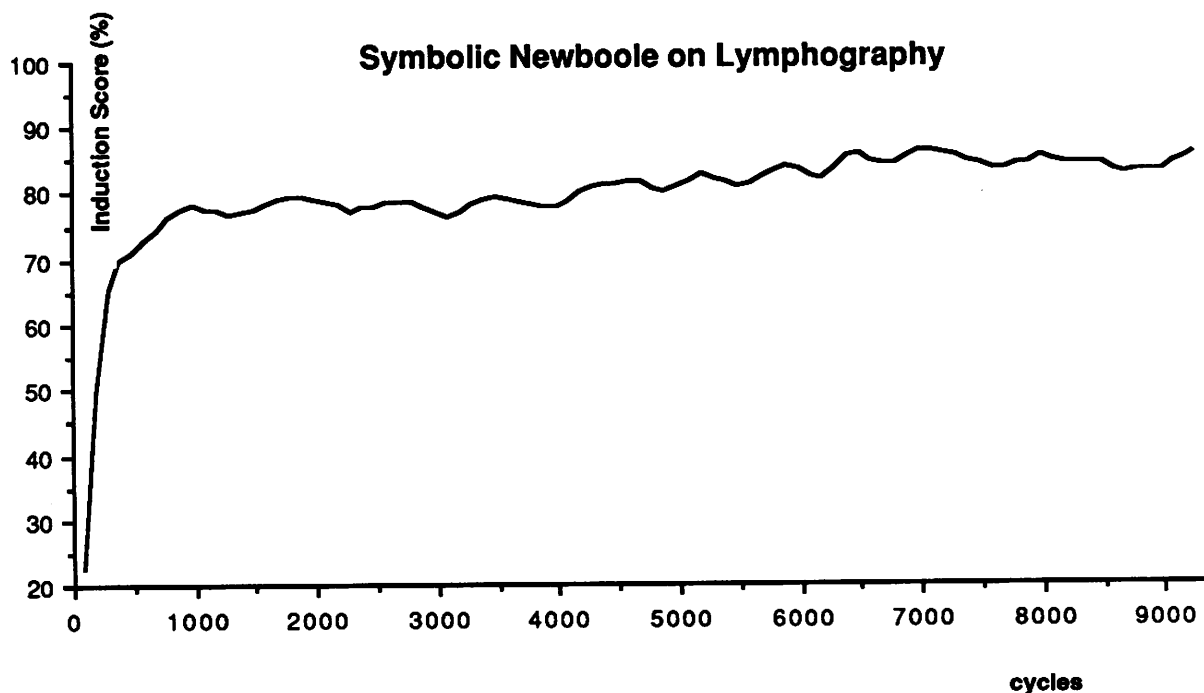


Figure 1: Symbolic Newboole convergence over test base

(smoothed over 3 points: one point every 100 cycles)

Name	Source	Inc.	Principle	Represent.	Induction
Bayes	[Clar., 87]	NO	Statistical	Probabilities	83 %
BP	⁽¹⁾	YES	Connectionist	Weights	82 %
Assistant-86	[Cestnik et al., 87]	NO	Entropy min.	Decision tree	76 %
AQ15	[Michalski et al., 86]	NO ²	Logic reduc.	Rules	81 %
Newboole	[Bonelli, Parodi Sen & Wilson]	YES	Genetic	Classifiers	76 %
Sym. Newboole		YES	Genetic Rules	Rules	82 %
Experts	[Michalski et al., 86]	YES	Unknown	Unknown	85 %

Table 1: Comparison of various learning systems using the lymphography data base:

This table presents the source of this data, learning incrementality, principle of the algorithm, knowledge representation and induction rate.

¹ We conducted this Back-Propagation test with a 18:60-40-20-20-4 (18 inputs, 4 hidden layers and 4 outputs) network architecture and only 3 trials rather than 4.

² This system acquires rules from examples which are considered one after the other; however, it only adds generalized logic formula and cannot revise its past knowledge accordingly to compress it. In this sense, it is not incremental as Back-Propagation, ID6 and GBL systems are.

More importantly, these systems have the potential to do therapy design as well, since they possess all the animat qualities.

Indeed, even Newboole learns by reward internally, once the example is known.

However, it is stricken that the best learning systems all do the same score on this medical diagnosis problem. This result also happens to be true in other domains such as phoneme recognition. This suggests that there is a limit to the quality of induction that a system can get by learning from a given data base; this might simply be the implicit knowledge that is contained inside it. This also supports the idea that if humans do slightly better on this problem, it is mainly because they had access to other knowledge before, in the form of other examples from a neighboring domain, or academic courses. This last remark illustrates how background knowledge could be useful in such a system. The ease to describe it in terms of rules partly motivates the need to use such a representation in lieu of classifiers.

Conclusion

In this paper, we have developed the idea that animat research helps to understand and reproduce complex human capabilities such as medical work.

We have experimentally shown that an improvement of Boole can learn to do medical diagnosis as well as other learning algorithms, and we discussed its unique potential to do therapy design. Moreover, our results suggest that genetic algorithms can be successfully applied directly on other knowledge representations than strings of bits.

Some interesting extensions of this work would be to check whether this remains true in other domains, apply our algorithm for medical therapy design, and to include background knowledge in the system.

Bibliography

- [Bonelli, Parodi, Sen & Wilson 90], "NewBoole: A Fast GBML System", Bonelli P., Parodi A., Sen S., Wilson S., Proceedings of the International Conference on Machine Learning, June 1990, Austin, Texas.
- [Bounds, 89], "Expert Systems and Connectionist Networks", Connectionism in Perspective, Pfeifer R. et al., North-Holland 1989
- [Cestnik et al., 87] Cestnik G., Kononenko I., & Bratko I., "Assistant-86: A Knowledge-Elicitation Tool for Sophisticated Users". In Bratko & Lavrac (editors) Progress in Machine Learning, 31-45, Sigma Press.
- [Clark & Niblett, 87] Clark, P. & Niblett, T. (1987), "Induction in noisy domains". In I. Bratko & N. Lavrac (Editors) Progress in Machine Learning, 11-30, Sigma Press.
- [Hendler, 88] "Activation Spreading Inconsistencies: On the need for hybrid symbolic/connectionist models", Proceedings of the AAAI 1988 Symposium on Parallel Models of Intelligence
- [Kodratoff, 89], "Enlarging Symbols to more than Numbers or Artificial Intelligence is the science of explanations", p. 157-172
- [Michalski et al., 86] Michalski, R., Mozetic, I., Hong J., & Lavrac N., (1986). "The multi-purpose Incremental Learning System AQ15 and its Testing Applications to three medical domains". In proceedings of the fifth National Conference on Artificial Intelligence, 1041-1045. Philadelphia, PA: Morgan Kaufmann.
- [Parodi & Khouas, 89], "Parodi, A., Khouas, S., "Adaptive Approach for Learning Abstractions in Semantic Networks", Neuro-Nîmes 89, International Workshop on Neural Networks and their Applications, Nîmes (France), 13-16 Nov, 1989.
- [Wilson, 85] Knowledge growth in an artificial animal, Wilson S. W., Proceedings of the first International Conference on Genetic Algorithms and their Applications, July 24-26, 1985, CMU, Pittsburg, PA, p. 16-23, Lawrence Erlbaum associates, publishers, Hillsdale, New Jersey
- [Wilson, 87] "Classifier Systems and the animat problem", Machine Learning 2, 199-228, 1987

Extended classifiers for simulation of adaptive behavior

Renaud Dumeur
 Département Informatique,
 Université de Paris-8,
 2, rue de la Liberté,
 93256 St Denis cedex 02
 red@litp.ibp.fr,
 C²V,
 82, Boulevard Haussmann,
 75008 Paris,
 renaud@ccv.fr

Abstract

In this paper, we propose the use of the animat approach to automatically generate animation scripts for computer synthesized movies. We want to design a system where animats are actors able to improvise from a few contextual informations. We plan to implement such animats with extended classifiers allowing a compact encoding of behavior rules while preserving their ability to be modified by inductive genetic operations.

1 Introduction

When computer animated characters become more and more realistic in their rendering, the problem of specifying more realistic individual and collective behaviors also appears. We think that animats can be used to solve this new problem and that behavior simulation based systems can change the way computer animations are designed. We propose a “programming by environment” approach using animats and evolution simulation which can drastically reduce the work of animation script writing.

Animats are computer simulated entities, exhibiting animal-like autonomous individual or collective behavior. We want to use them as low-cost credible crowd artists in order to ease the writing of animation scripts. In the world of film making, crowd artists are employed to give the audience background informations about the time and location where the action of the movie takes place. The movie director gives to these actors fuzzy indications on how to behave, then they are left nearly without control during the filming. On the other hand, as the central character behaviors may not easily inferred from the movie context, the corresponding actors are more precisely directed and have less freedom than crowd artists.

2 Animation automation

Consider traditional animation movies: drawn by hand, they require a frame-by-frame precision level script for each character of the animation. A first level of automation consists in drawing a few frame images by hand and apply an interpolation procedure to generate the missing frames required for a smooth animation. Improvements from the traditional paper based technique only reside in greater flexibility of the drawing tools and reduction of involved manpower.

By using computers, we can automate the generation of the script by writing procedures in a computer language. Object oriented [6] languages seem to match the requirements of animation programming. In such languages, an object is specified by a local state (a set of state variables) and a set of procedures (the object’s methods). An object’s behavior is implemented by its methods. As its methods process the object’s local state, it is easy to obtain a wide range of behaviors from a few procedural specifications, by varying the content of the entities local states.

If object oriented programming systems facilitate the behavioral specification of large groups of simulated actors, they do not give any assurance that the resulting acting will meet the desired goals. This work is still under the responsibility of the designer who must keep in mind a model of the potential interactions between actors.

A few fixed procedures can simulate complex behaviors. Even a cellular automaton, Conway’s *game of life* [2], can generate primitive animats – *gliders* and *glider guns*, for instance – with interesting behaviors. In the frame of computer animated graphics, we are moreover helped by spectators who will tend to interpret the events occurring in the movie. This fact is highlighted in V. Braitenberg’s book [3] where simple animats (called *vehicles*) are involved. These animats are carts, endowed with captors (photosensible cells) and effectors (propellers such as wheels), which are wandering in an environment containing light sources. By changing the disposition and the properties of the different vehicle brain compo-

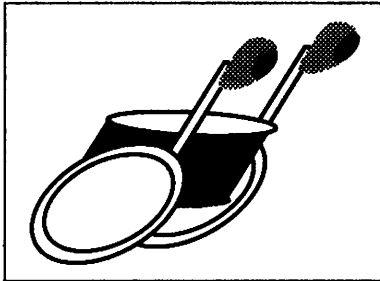


Figure 1: An animat speeding up.

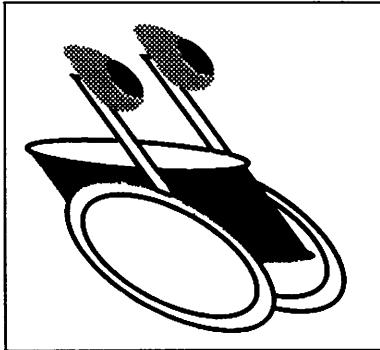


Figure 2: An animat hits the brakes.

nents, one can produce different behaviors, such as goal seeking strategies or avoiding reactions. Sometimes, these behaviors can be interpreted as the results of the animat "feelings", because the candid spectator, ignorant of the involved internal machinery, applies human rules for analyzing behavior. This effect can be amplified if we choose a cartoon-like animat representation. For instance, the figure 1 shows how we could represent an accelerating animat.

The same way, when a roving animat notices an attack or an obstacle, we can amplify this fact with a representation as shown by the figure 2.

If we make our animat interact with other object such as light sources, we can use classical rendering techniques to amplify the animat action. The figure 3 shows an animat locating a light source, its shade being projected on the ground.

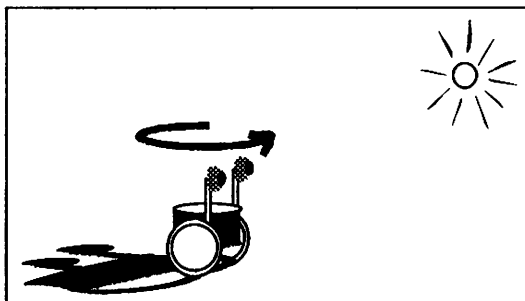


Figure 3: An animat locating a light source.

3 Adaptive behavior

For the spectator, the interest of an animated character also consists in the way it interacts with its environment. The more different environmental responses the character will have, the more the spectator will be speculating on the progress of the animation and will keep his attention focused on the movie. His motivation will last until he has a full model of simulated characters' way of reacting to environmental stimuli.

The most straightforward way of achieving variety in animat environmental response would be to encode large set of rules in the animats, each rule associated with a specific environmental input. Such a design will only extend to a fixed amount the duration of the spectator interest. If the environmental stimuli are fluctuating enough, they will exhaust the set of possible behavior of the animat. It is still possible to choose a stochastic rule selection scheme in order to introduce some noise in the animat response. If such a solution will confuse the spectator, maybe enough to keep his attention, it will still allow him to find the underlying determinism of the rules. Moreover, if this noise is too high, the spectator will perceive the character's actions as groundless and may be bored.

When large groups of animats are put into play, the problem of an initial rule set does is not as important as it is for a few animats. The collective behavior not only depends from the rule set contained in each animat, but also from animats interactions. Compared to a single animat rule set, this complexity is a better match to both the environmental variety and the analytic abilities of the spectator. Real world entities such as ants give goods example of adaptive behavior emerging from interaction between primitive elements [10] which can be transposed into a computer simulation. Work on bird and fish flocks [11, 1] have also produced attractive results. Nevertheless, in both examples, the the relationships – repulsion or attraction for instance – between the different animation actors must still be defined by the animation designer. As we wish to overcome this kind of constraints when generating animation scripts, we must use inductive mechanisms that allows the generation of new behavioral rules.

4 Adaptive vehicles

For our experiment, we want to use animats similar to V. Braitenberg vehicles. In these animats, environmental events are transformed by the captors, then processed by a network of fibers linking threshold devices [9] reaching effectors, and eventually transformed by the effectors in action on the environment. In the case of our very simple animat, this action is a motion caused by the rotation of powered wheels.

Such a design is based on analogic signal processing which does not match formalisms human beings are used to deal with. As one of our goal is the evolution of a predefined set of rules, we prefer a different model: production rules. Such approach has been introduced by J. Holland [7] with its Genetic Algorithm (GA) processed classifier systems (CFS) and used

for adaptive behavior simulation by [12].

The use of production rules for computer animation allows the animation designer to specify some basic behavior and let them be processed by an evolutionary mechanism.

5 Genetic Algorithms

Genetic Algorithms are search and optimization techniques based on the mechanics of natural evolution as described by the neodarwinian theory [5]. GA's main goal is to solve combinationally exploding search problems with processes drawn from biology. The relation between the two domains becomes clear if we look at biological evolution as a search process for the possible life forms in the space of the possible chemical constructs.

The RNA-DNA couple is the basic material containing informations which specify the way life forms are built and are functioning. This large and complex chemical construct is a product of natural evolution initiated from simpler molecules¹. Alterations (random *mutations*) of this molecule have consequences on the physical characteristics (phenotype) of the life form it builds. Another operation, the *crossover* which, occurs during sexed reproduction, allows genetic characteristics to be quickly spread in a population.

Simplified genotypes and genetic operations can be simulated on a computer. In simulated genetics, genomes are reduced to a vector of features which are modified and diffused in the population by random mutation and crossover. A selection pressure is also simulated which corresponds to a reproduction probability for the different genotypes. As fitting individuals will reproduce more often than the others, their genetic characteristics will be spread faster in the population. Consequently, the average fitness of the population will tend to an optimum, the stochastic character of the genetic operators allowing the system to escape from local maxima of the search space.

In the case of animats, we can use GAs to explore the space of possible behaviors. To do this, it is necessary to code behavioral descriptions under a form which supports the application of genetic operators.

6 Classifiers

A classifier is a linear string of symbols from $\{0, 1, \#\}$. This string is divided into two parts : a *condition* part and an *production part* of equal sizes. A classifier system (CFS) is a group of classifiers exploiting messages coming from a mailbox. A message is a linear string of symbols from $\{0, 1\}$ which size is half the classifier's one.

Briefly, a CFS run is a repetition of a cycle decomposed as follows:

1. construct an environmental message and put it in the

¹it is also suggested that DNA has been generated by clay crystals [4]

mailbox. Mark it specifically as an *input* message to give to classifiers the opportunity to distinguish it from other messages.

2. collect matching classifiers. A matching classifier has a *condition* part identical to the message, except that both 0 and 1 match #.
3. evaluate the bid made by each matching classifier. The bid usually depends on the classifier's strength (past utility) and expresses the ability of a classifier to solve the current situation.
4. choose matching classifiers for message production (the higher the classifier bid is, the more probably the classifier will be chosen). Post the output message which will be inserted in the CFS mailbox for the next step. This output is a copy of the *action* part of the classifier, except that each # symbol has been substituted by the identically placed value in the input message. The result of the rewriting is marked as an *output* message so the environment will be able to interpret it.
5. remove *output* messages and interpret them as actions on the environment.
6. reevaluate each classifier strength coefficient. This coefficient can be modified by:
 - an external payoff which is the evaluation of the CFS performance by the environment,
 - by incomes from other classifiers which have received previously posted messages,
 - and by penalties associated to the posting of these messages during the current cycle. Other penalties, such as life costs, may help to put tune the selection pressure on the classifiers.

The whole system can be viewed as a microeconomy, where external events are communicated with messages to elementary agents. Messages can be passed between agents via the CFS mailbox and can form activation chains. Those chains are usually maintained by strength revision schemes such as J. Holland's *Bucket Brigade algorithms*.

Without any induction mechanism, a CFS will select, depending on environmental constraints, the subset rules which is relevant for the environment and have been proved useful to the system (their actions have received a payoff from the environment).

If we apply a GA to a CFS, it becomes able develop new rules. Mutations in the classifier conditions create new pattern matching abilities and also create new responses to stimuli when they touch the action part. When Crossover recombines features from two existing classifiers, the two resulting *off-springs* have good chances to fit the environmental constraints because their conditions and actions are derived from existing features owned by selected individuals, thus reflecting some

aspects of the environment. This aspect has been thoroughly studied in [8].

6.1 A CFS based animat

Consider a minimal CFS based animat brain. Captors put binary representations of stimuli in specific places of the environmental message stored in the mailbox. Suppose we want to specify a basic set of rules for this animat, to make him attracted by light. We choose to set three one-bit quantities representing the animat perception in the environmental message. The two first values of this message are function of the light intensities received at each captor :

- If one eye receives more light than the other, the corresponding bit in the message is set to one.
- When no light is received by any of the eyes, the two bits are set to zero.
- When both eyes receive the same amount of light, both bits are set to one.

The third bit indicates if the animat has reached its goal.

On the effector side, the first bit and the second bit of the output message respectively command the left and the right wheel motors. When an effector bit is set to true, then the corresponding motor is on.

Using the classifier semantics, we can implement a light-seeking behavior with the following rules:

```

if not (LEFT_EYE) and RIGHT_EYE and not (TOUCH)
then
    LEFT_MOTOR = true
    RIGHT_MOTOR = false
endif

if LEFT_EYE and not (RIGHT_EYE) and not (TOUCH)
then
    LEFT_MOTOR = false
    RIGHT_MOTOR = true
endif

if LEFT_EYE and RIGHT_EYE and not (TOUCH)
then
    LEFT_MOTOR = true
    RIGHT_MOTOR = true
endif

if TOUCH
then
    LEFT_MOTOR = false
    RIGHT_MOTOR = false
endif

```

The above rules can be coded in the following four classifiers:

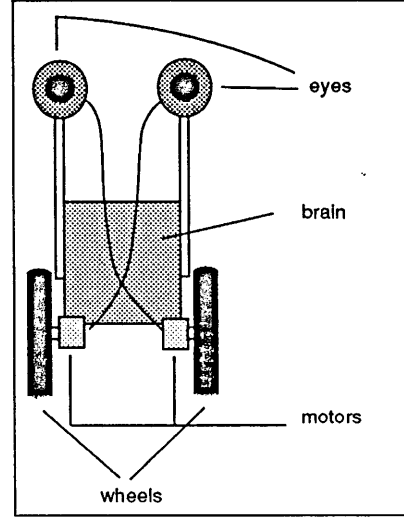


Figure 4: A light seeking animat built with analogic fiber technology (front view).

Conditions			Actions	
left eye	right eye	touch	left motor	right motor
0	1	0	1	0
1	0	0	0	1
1	1	0	1	1
#	#	1	0	0

Such a simple behavior requires four rules because of the implicit information used by the message rewriting process: the positioning of # symbols.

Using the other way, based on fibers/threshold devices technology, a similar behavior is easily implemented if we connect each animat's captor to the opposite motor (cf. figure 4).

Of course, we can arrange the animat interface so that each motor effector in the classifier action part matches the opposite captor in the classifier conditions. But in this case, things become more complicated if we decide to change the behavior of our animat.

In order to reduce the number of rules needed by so simple behaviors we can endow the classifiers with a connection map. This map specifies paths between input features detected by the rule condition filter and the output features which will be stored in the output message.

7 Rules with connection maps

Paths between input and output classifier filters are implemented by adding a *projection area* to each classifier. This projection area is a temporary work memory where some input message features are stored when the classifier is fired. Thus, the size of this memory is the same as a CFS message: half the size of a classifier. The projected features are then stored in the output message. The *connection map* specifies the paths followed by input features, filtered by the classifier

conditions, stored into the projection area, then transferred in the output message through the rule actions filter.

The message rewriting process can be decomposed in two phases:

1. fill each projection area slot with input features pointed from the projection map. For each projection slot i :

$$projection[i] = input[map_{conditions}[i]]$$

2. fill each output message slot with projected features pointed by the projection map. For each output message slot i :

$$output[i] = projection[map_{action}[i]]$$

When an extended classifier input filter slot contains #, it means that both 0 or 1 can be stored in the associated projection area slot. The same way, if a classifier output filter slot also contains # then it will store the content of the corresponding projection area slot in the output message.

Note that the part of the map covering the conditions is of type

message position \rightarrow projection area position

although the part used for output projection is of the form

projection area position \rightarrow message position

Such a scheme ensures that no output message will be underspecified and that crossover and mutation will never produce invalid maps. Moreover, the proposed rewriting process can be performed in parallel because it avoids conflicts during memory transfers.

It is interesting to note that GA driven modifications on rule connection maps and condition/action patterns do not alter their meaning in a disrupting way. For instance, if we consider two # symbols placed respectively in the condition and in the action part of a rule. If they are linked by the connection map, it means that we have a *variable*, local to the rule, which is bound at rule fire time. If one of these symbols is turned by mutation into 1 or 0, the variable disappear and is replaced by a constant. In the later case we can consider the transformation – as it is the case of classifier systems – as a rule specialization. Differently, if the connection map is altered but the variable maintained, that is one of the variable binding site have been changed, then we can say that the variable has been substituted by an other one.

7.1 Animat's rules with connection maps

The rewriting, using connection maps, of our last example of animat behavior will lead to the following rule set :

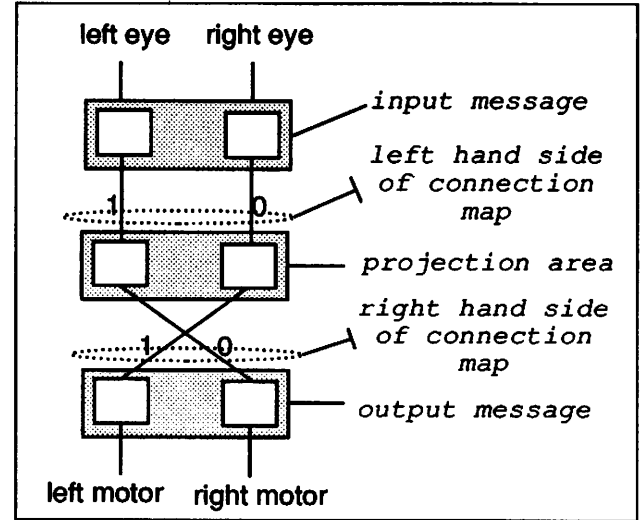


Figure 5: Connections for a light seeking animat.

```
if not (TOUCH)
then
  LEFT_MOTOR = RIGHT_EYE
  RIGHT_MOTOR = LEFT_EYE
endif
```

```
if not (TOUCH)
then
  LEFT_MOTOR = false
  RIGHT_MOTOR = false
endif
```

The figure 5 displays how the connection map creates crossing links between the input part and the output part of the light seeking rule. Note that the first three rules given in section 6.1 are replaced by only one equivalent rule.

Although we have no experimental results on the performances of such rule system, one can see that the size of the rule space is dramatically increased by the adjunction of connection maps to rules. The set of every possible rules of length N becomes $(\frac{3N}{2})^N$ instead of 3^N for the traditional classifiers.

8 Conclusion

About GA processing, the exploration of such huge amount of rules can be shorten if we initialize the system with a set of predefined rules. We also hope that the greater message rewriting abilities of the rules will allow us to reduce the size of the rule population. This reduction is of great interest if we wish to disassemble the classifiers to a human readable form.

On the GA side, many issues must be fixed. For instance, which kind of crossover is preferable? Is it a good thing to separate crossovers on the connection map from crossovers on rule condition/action filters?

The automated production of systems able to challenge the human analytic abilities of modeling behavior, to surprise peo-

ple and to keep them interested may become a growing field. The long term consequences of work on this domain could have a strong impact on fields, such as the entertainment business, which seems to be reserved to human beings. In fact, it would be a natural evolution for this kind of activities, which nowadays are industries, to take advantage of automated creation.

References

- [1] I. Aoki. Internal dynamics of fish schools in relation to inter-fish distance. *Bulletin of the Japanese Society of Scientific Fisheries*, 50(5):751–758, 1984.
- [2] E. Berlekamp, J.H. Conway, and R. Guy. *Winning Ways for Your Mathematical Plays*. Academic Press, New York, 1982.
- [3] V. Braitenberg. *Vehicles: Experiments in Synthetic Psychology*. MIT Press, Cambridge, 1984.
- [4] A.G. Cairns-Smith. The first organisms. *Scientific American*, 252(6):90–100, June 1985.
- [5] R. Dawkins. *The Blind Watchmaker*. W.W. Norton, New York, 1986.
- [6] Adele Goldberg and A. Robson. *Smalltalk-80: The language and its implementation*. Addison-Wesley, Reading, Massachusetts, 1983.
- [7] J. H. Holland. *Adaptation in Natural and Artificial Systems*. University of Michigan Press, Ann Arbor, 1975.
- [8] John H. Holland, Keith J. Holyoak, Richard E. Nisbett, and Paul R. Thagard. *Induction*. The MIT Press, 1986.
- [9] W.S. McCulloch and W. Pitts. A logical calculus of the idea immanent in nervous activity. *Bulletin of Mathematical Biophysics*, 5:115–133, 1943.
- [10] F. Moyson and B. Manderick. *The Collective Behavior of Ants: An Example of Self-Organization in Massive Parallelism*. Technical Report, Vrije Universiteit Brussels, 1988.
- [11] Symbolics and Withney Demos Production. Stanley and stella in breaking the ice. Film, 1987.
- [12] S. Wilson. Classifier systems and the animat problem. *Machine learning*, (2):199–228, 1987.