Eigenvalue Elasticity Analysis

Rogelio Oliva

Loop Eigenvalue Elasticity Analysis

The Mathematica® implementation of the LEEA Utility (`LEEA.nb`) requires that the `FeedbackLoops.m` package (also available in the electronic supplement) is loadable into Mathematica®. See §A.8.2 of the Mathematica Book (Wolfram, 2003) for directory locations for the file to be loadable.

The LEEA utility requires two inputs to perform the analysis of a model: 1) a Mathematica® version of the model, and 2) a data file with the values of the system state variables at all the points in time that have been chosen for performing the analysis.

A utility to translate a Vensim® model into a model description suitable for the Mathematica® routines to perform LEEA is available at http://tools.systemdynamics.org/loop_eigen/.[1] The utility takes as input a Vensim® (`*.mdl`) format file, and generates a Mathematica® (`*.nb`) file with the following nine subsections:

- The first subsection reports the source of the file
- The following seven sections describe the model
  a) a list of level variables
  b) a list of initial conditions for level variables
  c) a list of auxiliary variables
  d) a list of model parameters
  e) a list of table functions
  f) a list of simulation parameters
  g) a list of summary variables needed by LLEA & DDWA modules
- The last subsection reports the model equations that could not be processes.

Each element of the level and auxiliary variable lists is itself a list with the {lhs,rhs} format. The rest of the lists contain assignations var->value. The utility has some limitations—all listed in the utility website. Some limitations reflect the current state of development of the tool, but in theory they should be implementable without major problems (e.g., variables and function naming conventions, dynamic functions, macros, arrays).[2] A second group of limitations, however, are imposed to allow the LEEA utility to symbolically linearize the system and derive its Jacobian matrix. The most significant of this latter type of constraints is the need to represent the lookup functions as continuous differentiable functions. Our sample model does not contain any table functions, but examples of these representations of table functions can be seen in previous studies (e.g., Kampmann and Oliva, 2009; Oliva and Kampmann, 2010; Saleh *et al*., 2010). In our sample model (`NF_model.mdl`) we adapted variable names to reflect the utilities constraints, thus the translator output file (`NF_model.nb`) is ready for utilization by the LEEA utility.

Strictly speaking, it is not necessary to supply anything more than the values of the state variables for the software to calculate the values of all the auxiliary variables and rates. However, the easiest way to export the data from Vensim® is to use the Model>Dataset Export … command

[1] Retrieved March 25, 2014.

[2] The code of the translator utility (Pearl) is available from the author. Users are encouraged to expand its functionality.

(use the "tab" and "time running down" options of this command to obtain the file format expected by the LEAA utility). The LEEA Utility will calculate the LEEA for every time step stored in the *.tab file. Use Vensim® Saveper option to control the frequency of calculations. For our base case (Base.tab) we use a Saveper of 1 year, meaning that we will perform the LEEA analysis once a year.

To execute the LEEA analysis, in Mathematica® fist evaluate (Evaluation>Evaluate Notebook) the model notebook (`NF_model.nb`) to load the model structure into memory – there is no visible output from this evaluation. Once the model has been loaded into memory, evaluate the LEEA utility notebook (`LEEA.nb`). The utility will ask for the tab-delimited file with the data (`Base.tab`). Use the normal navigation commands to locate the file. Once the file loads, the utility will perform the analysis.

Dynamic Decomposition Weights Analysis

Like the LEEA utility, the Mathematica® implementation of the DDWA utility (`DDWA.nb`) requires that the `FeedbackLoops.m` package (also available in the electronic supplement) is loadable into Mathematica®. See §A.8.2 of the Mathematica Book (Wolfram, 2003) for directory locations for the file to be loadable. Unlike the LEEA utility, the DDWA utility only requires the Mathematica® version of the model[3], as the elasticities of all parameters are estimated from the model's initial conditions.

To execute the DDWA analysis, in Mathematica® fist evaluate (Evaluation>Evaluate Notebook) the model notebook (`NF_model_full.nb`) to load the model structure into memory – there is no visible output from this evaluation. Once the model has been loaded into memory, evaluate the DDWA utility notebook (`DDWA.nb`).

Bibliography

Kampmann CE, R Oliva. 2009. Analytical methods for structural dominance analysis in system dynamics. In Meyers R. (ed.), *Encyclopedia of Complexity and Systems Science*. Springer, New York, pp. 8948-8967.

Oliva R, CE Kampmann. 2010. Toolset for eigenvalue elasticity analysis. Retrieved Jan. 15, 2014. Available from http://iops.tamu.edu/faculty/roliva/research/sd/leea/toolset.html.

Saleh M, R Oliva, CE Kampmann, PI Davidsen. 2010. A comprehensive analytical approach for policy analysis of system dynamics models. *European Journal of Operational Research* **203**(3): 673-683.

Wolfram S. 2003. *The Mathematica Book*. 5th ed. Wolfram Media/Cambride University Press, Champaing, IL.

[3] See LEEA section above for a description of the utility to translate a Vensim® model into a Mathematica® version suitable for these analysis.