Eigenvalue Elasticity Analysis†

Rogelio Oliva

(please do not distribute without permission)

†To appear in:
Rahmandad, H., R. Oliva and N. Osgood (eds). (2015). *Analytical Methods for Dynamic Modelers*. MIT Press: Cambridge, MA (In press).

Eigenvalue Elasticity Analysis

Rogelio Oliva

Abstract

Eigenvalue elasticity analysis (EEA) is a set of methods to assess the effect of structure on behavior in dynamic models. It works by considering observed model behavior as a combination of characteristic behavior modes and by assessing the relative importance of particular elements of system structure in influencing these behavior modes. Elements of the model structure that have a large influence on particular behaviors can provide important clues to the modeler to identify areas for further testing and study, and for policy analysis. The method uses linear systems theory to 1) decompose the observed behavior into its constituent *behavior modes*, such as oscillation, growth, and exponential adjustment, and 2) outline how a particular behavior modes and its appearance in a given system variable depend upon particular parameters and structural elements (links and loops) in the system. In this manner, the method provides a very precise account of the relationship between structure and behavior.

Introduction

The link between system structure and dynamic behavior is one of the defining elements of dynamic modeling. In a sense, a simulation model can be viewed as an explicit and consistent theory of the behavior it exhibits. Although this point of view has certain merits, not least the fact that it lifts the discussion from outcomes to causes of these outcomes and from events to underlying structure (Forrester 1961, Sterman 2000), we are concerned here with a more compact explanation of the system's behavior. In fact, most dynamic modeling projects report their results in terms of simpler explanations of the observed results, typically in terms of dominant feedback loops and, occasionally, external driving forces to the system that produce the salient features of the behavior.

For simple systems with relatively few variables it is usually easy to use intuition and trial and error simulation experiments to explain the dynamic behavior as resulting from particular feedback loops. In larger systems, this method becomes increasingly difficult and the risk of incorrect explanations rises accordingly. There is a need, therefore, for analytical methods that provide some consistency and rigor to this process.

These analytical tools are important to the practitioner because the structure-behavior link is the key to finding leverage points for policy initiatives. And they are important to the theorist because a system dynamics theory of a particular phenomenon is an account of how certain feedback loops cause certain dynamic patterns of behavior to appear. The qualitative understanding of the model behavior is often at least as important as the particular numerical predictions obtained, even in applied studies. Yet the rigor of such an account depends directly on the rigor with which structure-behavior link can be established in a given model.

Eigenvalue elasticity analysis (EEA) is a set of methods to assess the effect of structure on behavior in dynamic models. It works by considering observed model behavior as a combination of characteristic behavior modes and by assessing the relative importance of particular elements of system structure in influencing these behavior modes. Elements of the model structure that have a large influence on particular behaviors can provide important clues to the modeler to identify areas for further testing and study, and for policy analysis. The method represents a high degree of mathematical rigor compared to the traditional experimental simulation methods normally used in the field. The method uses linear systems theory to 1) decompose the observed behavior into its constituent *behavior modes*, such as oscillation, growth, and exponential adjustment, and 2) outline how a particular behavior modes and its appearance in a given system variable depends upon particular parameters and structural elements (links and loops) in the system. In this manner, the method provides a very precise account of the relationship between structure and behavior.

The EEA method enables large-scale models to be analyzed systematically in a manner that is not possible or practical using trial and error simulation. Given the rigor and relative sophistication of the method, it may also provide legitimacy to dynamic model analysis in fields that are traditionally dominated by analytical mathematics, such as economics and econometrics.

The purpose of the elasticity analysis is to analyze the relative importance of structural elements, not to estimate the strength of system elements or values of system parameters. Therefore, the method works very much from a given model structure and parameter set and then tells you something about what would happen if you modify the structure and/or parameters. It, thus, fits mostly in the interpretation and policy analysis stages of model building. It may also prove useful in the model building and testing stage, to the extent that it can help identify structures that produce unwanted or puzzling behavior.

The EEA methods are similar in aims and scope to the Pathway Participation Method (PPM) (Mojtahedzadeh, Andersen, and Richardson 2004). The main difference between the two approaches is that while PPM emphasizes identifying a single “dominant” structure that drives the behavior of a particular variable, and does so relying primarily on partial system structures, the EEA approach provides an overview of the relative influence that different pathways simultaneously have on a variable and does so considering global system properties (see Mojtahedzadeh 2008 for a comparison of the two methods). Duggan and Oliva (2013) summarize other methods that rely on iterative and sensitivity-based approaches to explore dominant structure.

A word of caution is in order: Like any other quantitative method, there is always an element of judgment and interpretation when employing the method in practice that cannot be avoided. Moreover, the results of the EEA may require some work to interpret: since the method involves a translation from patterns of behavior over time to complex number eigenvalues, the results can appear highly abstract. New measures are under way to facilitate more direct interpretation, but these are still in the developmental stage.

Background and Formulation

The method of using eigenvalue elasticities is based on the tools from modern linear systems theory (Chen 1970, Luenberger 1979), applied to a linearized model. The method was first introduced in system dynamics with Nathan Forrester’s doctoral dissertation (1982). He used the method in the context of a macroeconomic model to explore various stabilization policies. However, the method was only peripheral to the dissertation, with most of the emphasis being traditional simulation experiments. Some attempts were made using eigenvalue analysis in the National Model project at MIT, but the limited availability of software and difficulty in interpreting the results prevented the method from gaining extensive use.

In 1996, Kampmann (2012) reintroduced the method by combining it with network and graph theory to reveal some fundamental relationships between feedback loops and eigenvalue elasticities. In particular, he highlighted that there is typically a very large number of alternative loop descriptions of a system and introduced the notion of an independent loop set (ILS), somewhat similar to the set of basis of a vector space, from which all other feedback loops can be said to be derived from. He further demonstrated how eigenvalues or behavior modes[1] are in a sense determined only by the *loop gains* in the independent set while the appearance of these behavior modes in the behavior of individual variables is a function of the *link gains* in the system.

The traceability of eigenvalue elasticity to specific feedback loops, together with the availability of software to support numerical and algebraic analysis (e.g., Mathematica, Maple, Matlab), and the advances of Pathway Participation Method, triggered a stream of research to test and expand the usefulness and applicability of EEA.[2] Kampmann and Oliva (2006) automated some of the computational requirements to perform EEA and tested the method across three types of models. They found that the utility of the method depended on the model structure, and that it was most useful for large-scale quasi-linear models. Güneralp (2006) developed a new measure that takes all model modes into account and proposed a normalization approach for elasticity values. Gonçalves (2009) and Saleh et al. (2010) extended the eigenvalue approach to focus on the overall trajectory of a state variable, and in particular the contribution of the eigenvector, which allows for the analysis of both short- and long-term impact on changes in link and loop gains. These papers, however, have focused on explaining how the method works and guiding the interpretation of results. As such, the authors have chosen simple and well-behaved models in which it is relatively easy to map the method's outcomes with the observed behavior and structure. To date, there is no documented case of the benefits of the application of the EEA methods to a realistic model, where structural dominance analysis is hypothesized to be most effective.

The following subsection provides an analytical description of the EEA method.

Analytical Description

Characterizing linear and nonlinear systems

A dynamic model can be represented mathematically as a set of ordinary differential equations

$$\frac{d\mathbf{x}(t)}{dt} = \dot{\boldsymbol{x}}(t) = \mathbf{f}\big(\mathbf{x}(t), \mathbf{u}(t)\big), \tag{1}$$

where $\mathbf{x}(t)$ is a column vector of n states variables (levels), $\mathbf{u}(t)$ is a column vector of p exogenous variables, $\mathbf{f}$ is a corresponding vector function, and t is simulated time. The system is said to be linear (nonlinear) if $\mathbf{f}$ is a linear (nonlinear) function of its arguments. Given the model structure (1), knowledge of the initial conditions $\mathbf{x}_0$, and the path of the input variables $\mathbf{u}(t)$, the behavior of the model is completely determined. In this sense, the model structure (1) constitutes a "theory" of the behavior $\boldsymbol{x}(t)$.

[1] Since behavior modes are based upon the eigenvalues, the terms "behavior mode" and "eigenvalue" are used interchangeably in the following.
[2] See Duggan and Oliva (2013) for extended bibliography of this research stream.

The approaches considered in this chapter are based on tools from linear systems theory (Chen 1970) and they approximate the nonlinear model (1) with a linearized version, using the first-order Taylor expansion around some operating point $\mathbf{x}_0$, $\mathbf{u}_0$, i.e.,

$$\dot{\mathbf{x}}(t) \approx \mathbf{f}(\mathbf{x_0}, \mathbf{u_0}) + \frac{d\mathbf{f}}{d\mathbf{x}}(\mathbf{x} - \mathbf{x_0}) + \frac{d\mathbf{f}}{d\mathbf{u}}(\mathbf{u} - \mathbf{u_0})$$

or, by redefinition of the variables $\mathbf{x} \to \mathbf{x} - \mathbf{x}_0 - \mathbf{f}(\mathbf{x_0}, \mathbf{u_0})(t - t_0)$ and $\mathbf{u} \to \mathbf{u} - \mathbf{u_0}$,

$$\dot{\mathbf{x}}(t) \approx \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t), \tag{2}$$

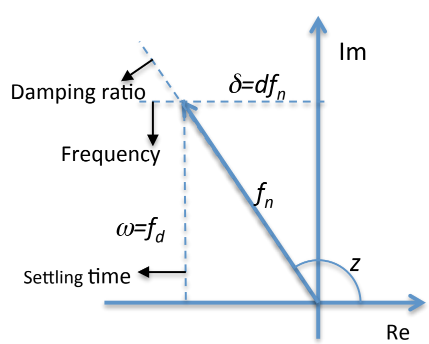
where $\mathbf{A}$ is a constant $n \times n$ matrix of partial derivatives $\partial f_i / \partial x_j$ and $\mathbf{B}$ is a constant $n \times p$ matrix of partial derivatives $\partial f_i / \partial u_j$, and all partial derivatives are evaluated at the operating point. These matrices of first order partial derivatives are know as *Jacobian* matrices. Both the eigenvalue and eigenvector elasticity analysis based are upon this approximated system.

Initially, one may primarily be concerned with the endogenous response of the system, in which case one can set the exogenous or control variables to a constant.[3] In the absence of changes in exogenous inputs, the resulting behavior of any given state variable $x(t)$ can be written as a weighted sum of a set of behavior modes,

$$x_i(t) \ = \ w_{i,0} + w_{i,1}e^{\lambda_1 t} + \cdots + w_{i,n}e^{\lambda_n t}, \tag{3}$$

where the λ's are the eigenvalues of the system Jacobian matrix $\boldsymbol{A}$ and the weights w are constants that depend upon the eigenvectors and the initial conditions of the system (see Saleh et al. 2010 for derivation).

Equation (3) yields three important insights. First, each of the system eigenvalues represents a behavior mode. For real eigenvalues, the behavior mode $e^{\lambda t}$ amounts to an exponential growth ($\lambda > 0$) or adjustment ($\lambda < 0$). Complex eigenvalues appear in conjugate pairs $\delta \pm i\omega$, which give rise to oscillations $e^{\delta t}\sin(\omega t + \theta)$ of frequency ω that are either expanding (if $\delta > 0$) or damped oscillations (if $\delta < 0$). The absolute value of λ is known as the *natural frequency* $f_n = |\lambda| = \sqrt{\delta^2 + \omega^2}$ while the imaginary part of λ is known as *damped frequency* $f_d = \omega$.



[3] See Kampmann and Oliva (2006) for a discussion of when such an approximation is appropriate and useful.

Figure 1. Characterization of eigenvalues in the complex plane

Second, the behavior of every state variable in the system is a *constant* weighted sum of the system behavior modes. That is, the behavior of every state variable in the system is the result of how each of the behavior mode λ is projected into that state variable w. Finally, the system core behavior modes are structurally determined, as they are derived from the eigenvalues λ of the system matrix A.

Different 'flavors' of EEA emphasize each of the three insights in different ways (Kampmann and Oliva 2008, 2009). For example, EEA can be used to develop 'structural explanations of behavior' as it can pinpoint which system elements are responsible for generating a particular behavior mode λ. Alternatively, the tools can be used to derive effective policy recommendations by isolating the system elements that affect the projection of a particular reference mode in a stock (w), or altogether change the system reference modes λ. Before describing in details the tools to perform these analyses, the next subsection presents an example of these computations with a simple model.

Simple Example

Consider the Lotka-Volterra model in Figure 2, defined by the following equations

$$\frac{dx}{dt} = Bx - Dx, \qquad \frac{dy}{dt} = By - Dy,$$

$$Bx = ax, \qquad Dx = bxy, \qquad By = cxy, \qquad Dy = dy$$

Where x represents the prey population, y the predator population and the parameters, a, b, c, and d, determine respectively the natural growth rate of the pray population in the absence of predators, the efficiency of predation, the predator reproduction rate (per available pray), and the natural death rate of predators in the absence of pray. With appropriate initial conditions and parameter values, the model reaches a limit cycle with the two populations rising and falling alternatively (see Figure 2).

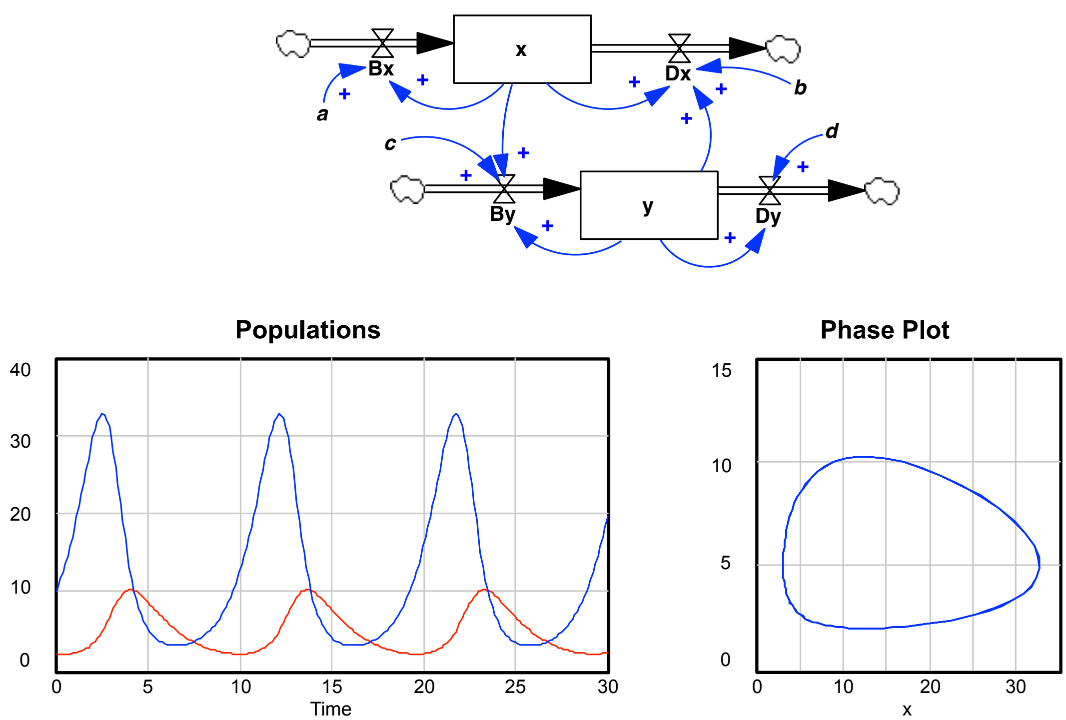


Figure 2 Structure, trace and phase plot of Lotka-Volterra model†

†Parameter values: *a=1, b=0.2, c=0.04,* and *d=0.5;* initial conditions $x_0=10$ and $y_0=2$.

Since the model has no exogenous inputs, the matrix representation of the system can be derived directly:

$$\begin{pmatrix} \dot{x} \\ \dot{y} \end{pmatrix} = \begin{pmatrix} a - by & -bx \\ cy & cx - \delta \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix}$$

The eigenvalues of the *Jacobian* or system matrix **A** are the conjugated pair

$$\left(a - d + cx - by \pm \sqrt{(a + d - cx)^2 - 2by(a + d + cx) + b^2y^2}\right)/2.$$

The eigenvalues are a function of *x* and *y*, so their value will change throughout the simulation. The eigenvalues at time zero are 0.25±0.194i and at that point in time the behavior of the two stocks can be represented by the following equations:

$$x = 20 - 45.018\, e^{0.25t} sin(0.224 - 0.194t)$$
$$y = \ \ 8 - \ \ 9.004\, e^{0.25t} sin(0.729 - 0.194t),$$

where the first term represents a scaling parameter, the coefficient of the second term the weight *w* of the behavior mode on that stock (in this case the two eigenvalues collapsed into a single oscillatory behavior mode), and the first term inside the sin function represents the phase lag of this projection. Although these trajectory equations are only valid around the operating point x=10, y=2, i.e., the trajectory equations will change with the eigenvalues, they give a clear indication of the tendency of the system and it is possible to assess the impact of the system structure on the behavior at that point in time. To fully understand the behavior of the system, it would be necessary to replicate the analysis at different operating points. Before leaving this example, however, it should be noted that for most stock values within the limit cycle, the eigenvalues take complex values indicating an oscillatory behavior mode.

The main strategies for exploiting the information available in this description of the system are described in the following subsections.

Eigenvalue Elasticity and Influence

The EEA is concerned with assessing how system structure affects the behavior modes *(*λ*)* as well as the projections of those behavior modes in a particular stock *(w)*. A measure of the impact on an eigenvalue λ when one changes individual elements a of the system matrix is the eigenvalue elasticity,

$$\varepsilon \ = \frac{\partial \lambda}{\partial a} \frac{a}{\lambda}.$$

The most granular element of system structure is the gain of a link between two variables, i.e. the ratio of the output to the input. For example, in the model above, the gain of the link between *x* and *By* is *cy*. Clearly, all elements *a* of the system matrix **A** are combinations of these individual link gains, and thus it is possible to make assessments of eigenvalue elasticity to each link gain and model parameter.

For a complex-valued eigenvalue, the elasticity measure will also be a complex number. One may define the elasticities of the real and imaginary parts separately, i.e., as the real numbers

$$\varepsilon_\delta \ = \frac{\partial \delta}{\partial a} \frac{a}{\delta}, \ \varepsilon_\omega \ = \frac{\partial \omega}{\partial a} \frac{ga}{\omega},$$

respectively. Note that it is not the case that $\mathrm{Re}\{\varepsilon\} = \varepsilon_\delta$ or $\mathrm{Im}\{\varepsilon\} = \varepsilon_\omega$, since

$$Re\{\varepsilon\} = \frac{\varepsilon_\delta \delta^2 + \varepsilon_\omega \omega^2}{\delta^2 + \omega^2}, \; Im\{\varepsilon\} = \frac{(\varepsilon_\omega - \varepsilon_\delta)\delta\omega}{\delta^2 + \omega^2}.$$

Kampmann and Oliva (2006), however, found that it is often easier to work with the so-called *influence measure* instead, defined as

$$\mu = \frac{\partial \lambda}{\partial a} a, \; \mu_\delta = \frac{\partial \delta}{\partial a} a, \; \mu_\omega = \frac{\partial \omega}{\partial a} a. \qquad (4)$$

For the influence measures, it is indeed the case that $\mathrm{Re}[\mu] = \mu_\delta$ and $\mathrm{Im}[\mu] = \mu_\omega$. In addition to simplifying interpretation, the influence measures also remove technical difficulties involved when eigenvalues are close to zero[4].

Loop Eigenvalue Elasticity Analysis (LEEA)

Kampmann (2012) showed that it was possible to express the characteristic polynomial[5] of the system matrix **A**, that is, the polynomial whose zeros are the eigenvalues of **A**, in terms of the gains of the loops in what he termed an *independent loop set* (ILS). The loop gain is defined as the product of the gains of its constituent links; for example, in the model above, the loop gain of the loop formed by {*x, y, x*} is (-*bcxy)*. An independent loop set is a maximal set of loops whose gains can be determined or changed independently of each other through an appropriate assignment or change in the link gains of the system. The gain of any loop outside this set is then dependent upon the loop gains in the ILS. Put differently, the ILS is a *complete description* of the feedback structure of the system, where the many additional feedback loops are redundant.

Once an ILS has been identified (see Oliva 2004, Kampmann 2012 for procedures), it is straightforward to calculate the gain g of each loop in the set and then use those gains as the basis for exploration of the behavior of the eigenvalues. Specifically, the loop eigenvalue elasticity and the loop influence metrics are defined as

$$\varepsilon = \frac{\partial \lambda}{\partial g}\frac{g}{\lambda}, \text{ and } \mu = \frac{\partial \lambda}{\partial g} g. \qquad (5)$$

While the ILS is not unique in a model, this decomposition focuses the analysis in a relevant subset of loops. In particular, changes in relationships in the model that are not part of a feedback loop will have no effect upon the system eigenvalues. Thus, one can interpret the elasticities or influence measures in terms of how they affect the gains of a set of (independent) feedback loops in the system. Alternatively, one can assess the relative importance of particular feedback loops in generating a particular mode of behavior, where loops with large elasticities (or influence) are considered important for the behavior mode in question.

---

[4] While the elasticity measure has the advantage that it is a dimensionless measure and hence independent of the choice of units in the model, the influence measure has the dimension $time^{-1}$, and so depends upon the chosen time unit. However, it is still independent of the choice of the other units in the model.

[5] The characteristic polynomial is defined as $P(\lambda)=\det(\lambda\mathbf{I}-\mathbf{A})$, where **I** is the identity matrix. The eigenvalues of **A** are the roots of $P(\lambda)=0$.

Dynamic Decomposition Weight Analysis (DDWA)

The Eigenvector, or Dynamic Decomposition Weight Analysis, introduced by Saleh et al. (2010) is concerned with what happens to the weights w in (3) when changes are made to the system elements (parameters and link gains). As with the LEEA, one may express the relationship either as influence measures or elasticities. Specifically,

$$\varepsilon_w = \frac{\partial w}{\partial a}\frac{a}{w}, \; \mu_w = \frac{\partial w}{\partial a}a.$$

Unlike in LEEA, however, where only those links in the model that are part of feedback loops will have any significance, all the links in the model are potentially relevant in the determination of the DDWs. Figure 3 presents the main analyses covered by these methods, as well as the inputs and outputs required by each.

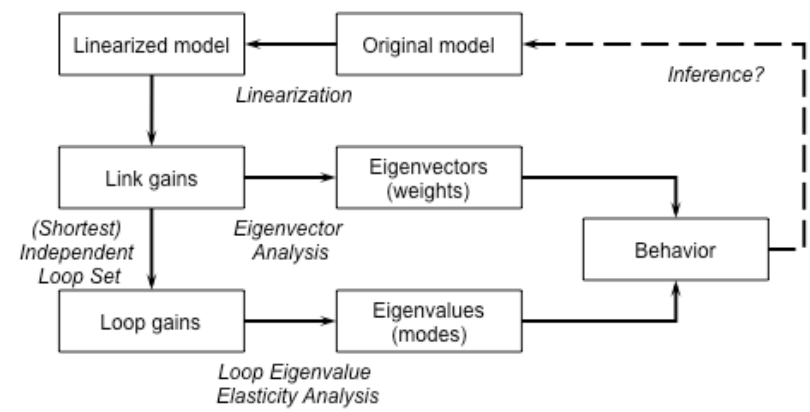


Figure 3. Schematic representation of EEA process[†]

[†]Adapted from Saleh et al. (2010)

As discussed above, interpretation of the results from these analyses is not necessarily straightforward, in particular because the methods could be used for different purposes, e.g., identifying structural explanations of behavior or policy design. As such, the outcomes of these analyses have not been standardized. In the examples below, I will use different representations of the outputs (i.e., the eigenvalues and eigenvectors, and the loop, link and parameter influences on them) that have proven useful to explain observed behavior in terms of system structure (feedback loops) and develop policy recommendations.

In order to conduct a meaningful policy analysis, it is necessary to specify a set of criteria for what constitutes a successful policy change. Forrester (1982) discusses different measures of stabilizing policies and their possible tradeoffs. This issue, however, is difficult to treat in general, since the policy criteria are linked to the purpose of the model and the problem definition, which may involve transient behaviors like overshoot and collapse (e.g., in the World model), or the settlement in the system to undesirable end states (e.g., in the Urban Dynamics model). In this paper, I focus on policies that reduce the oscillatory tendencies of the system, since the model presented is designed to address this issue, and since, as was demonstrated by Kampmann and Oliva (2010), it appears to be one of areas where the eigenvalue analysis shows the most promise.

In the context of unwanted instabilities (oscillations), effective policies are normally defined as those that either increase the damping of oscillatory behavior modes by making the real part more negative or decrease the frequency of oscillation. The LEEA can aid in finding the changes that have those desired effects and explain why the effect occurs in terms of the changes in feedback loop gains they imply. Another perspective, afforded by the DDWA, is to make changes that reduce the weights w of the oscillatory behavior modes in a particular system variable, i.e., reduce the amplitude of the variable's oscillations. Another aspect addressed by the DDWA is the degree to which external disturbances (from the exogenous variables) can be absorbed and dampened by the system. I have chosen to relegate this aspect to subsequent work.

Detailed Example

Both the LEEA and DDWA analysis methods have been implemented in Mathematica® routines and are available in the book electronic supplement with the example models in Vensim® and text parsing routines that generate the appropriate Mathematica® files from a Vensim® model file — the tools are also available online (Oliva and Kampmann 2010). Names of files available in the electronic supplement will be listed using a fixed-width font, e.g., `model.mdl`.

The model I use to illustrate the method is Nathan Forrester's macroeconomic model, used in his Ph.D. thesis (1982). The model serves this purpose well, both because it is close to linear and because the main emphasis of the model is to understand macroeconomic instabilities such as business cycles or longer cycles, and develop policies to stabilize these cycles. The purpose of the original model was to investigate various suggestions for fiscal and monetary policies to stabilize the economy. The model, which is shown in Figure 4, represents the relationships that are all part of standard macroeconomic theory, such as the consumption multiplier, the permanent income hypothesis, the Phillips curve, and the investment accelerator. For the reader unfamiliar with macroeconomics, detailed description of this theory can be found in any standard textbook, such as Dornbusch et al. (2010).

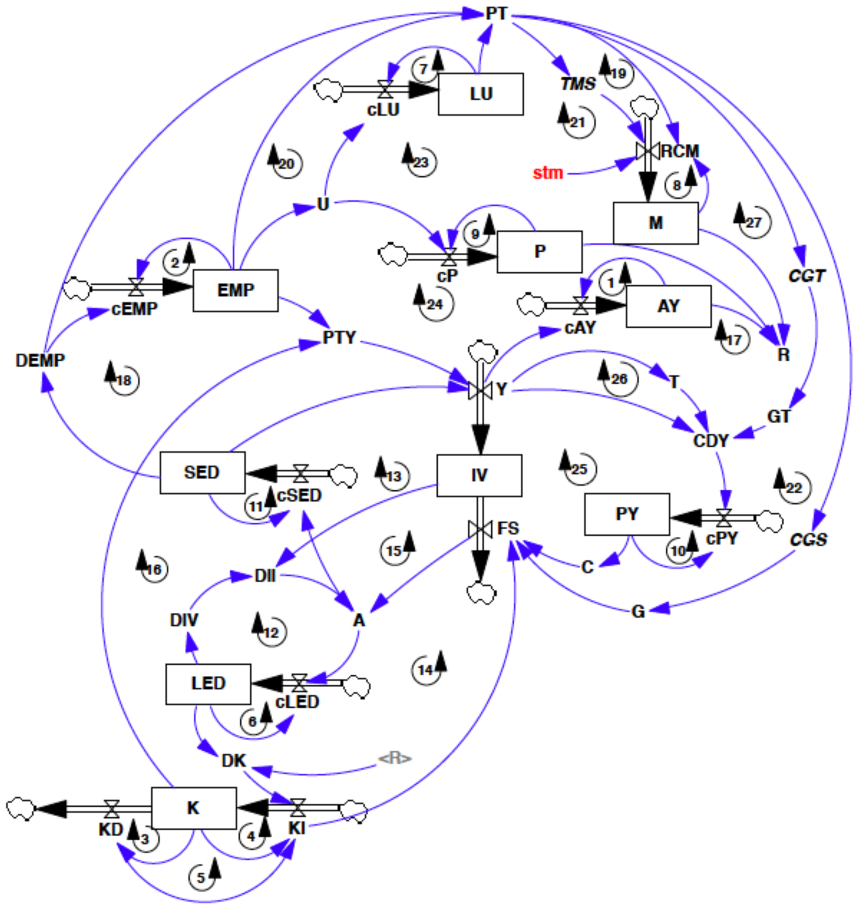


Figure 4. Diagram of the Forrester macroeconomic model

The model (`NF_model.mdl`) has a total of 10 levels. Four of these (LU, SED, LED, AY, and PY) are first-order delays of unemployment, short and long term expected demand, aggregate production, and disposable income, respectively. Moreover, employment (EMP) and capital stock (K) are also effectively first-order adjustments to desired employment (DEMP) and desired capital (DK), respectively. Thus, only the price level (P), the inventory level (IV), and the money supply (M) are not first-order delays. (The latter is de-activated in the base run of the model.)

The model used in our study is identical to the one listed in Forrester's dissertation with the following three minor exceptions.

- All smooth functions have been replaced by the explicit formulation of the stock adjustment process
- Stock values have been scaled by a factor of 10e8 to avoid very large negative exponents, which the software may otherwise truncate when doing the analysis.

- Instead of random noise, the IV stock is initialized about 3% below its equilibrium level. This generates smooth trajectories that are easier to interpret and understand[6].

For our base simulation (`Base.vdf`) all policy levers are inactive, and the behavior of the model is the same as reported in chapter IV of Forrester's dissertation. Figure 5 shows the characteristic response of the system, in this case aggregate output (Y) and the inventory (IV), capital (K) and employment (EMP) stocks. The inventory and employment stocks show a damped oscillation with a period of approximately 4 years and the capital stock shows a dampened oscillation with a period of about 30 years (not visible in the figure). The peaks and troughs of inventory stock lag the peaks and troughs of output by approximately two and half years. As such, the model appears to do a good job of replicating the salient features of the business and the capital cycle.

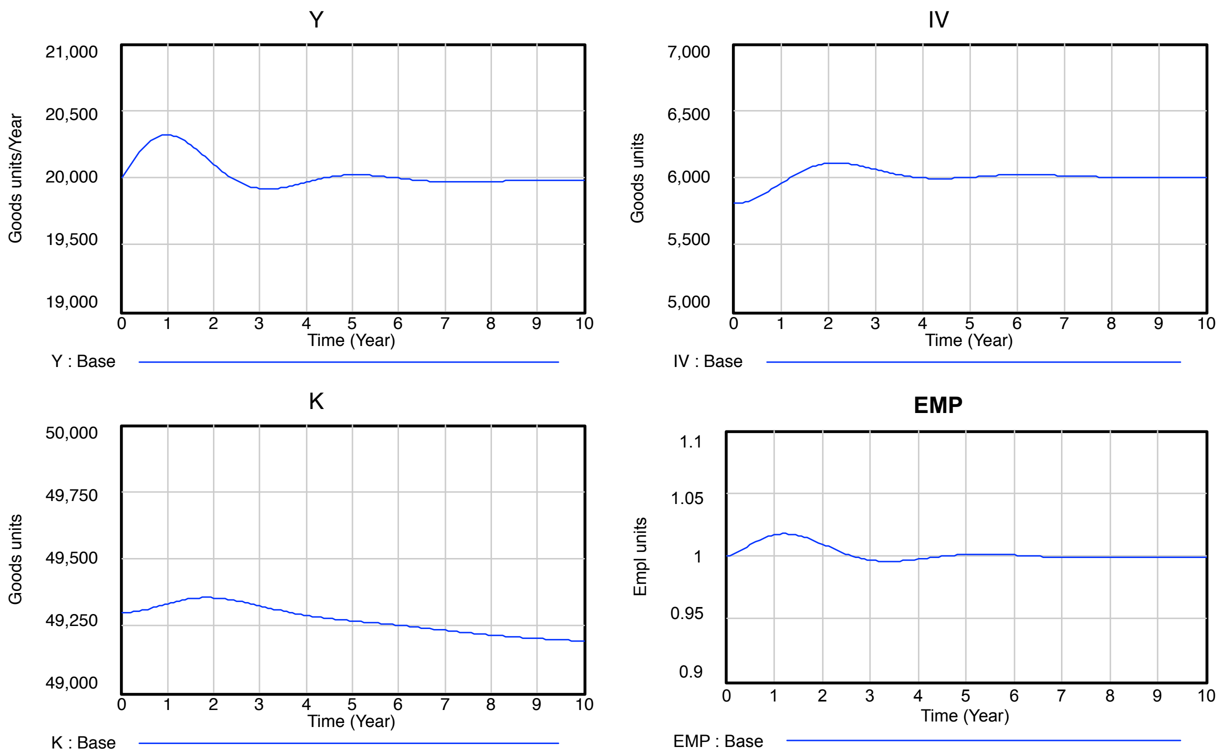


Figure 5: Base run of the Forrester model

[6] In Forrester's original model, the aggregated output (Y) and potential output (PTY) are modified with an additive random noise. The net effect of this noise is to perturb the system and prevents it from reaching equilibrium (the system is heavily dampened). While the EEA methods can work through these perturbations (see Oliva 2014), removing them allows us to understand the transient behavior from the point of linearization as if the system was not perturbed. This is a valid approach as the eigenvalues of the system are very stable throughout the simulation horizon (see following section).

LEEA

Base Model

The Mathematica® implementation of 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. The electronic supplement (`Appendix.pdf`) provides detailed instructions for preparing these files from our sample model (`NF_model.mdl`), as well as executing the analysis. The supplement also lists the current limitations of the implementation.

The first two sections of the LEEA notebook (`LEEA.nb`) contain the instructions to use the utility and the commands to import the data. The following five sections are the computational core of the utility. Section 3 derives the graph representation of the model structure (see Oliva 2004); section 4 identifies a Shortest Independent Loop Set (SILS) (Oliva 2004) that will be used as the base description of the loops in the model; sections 5 and 6 symbolically derive the link and loop gains as well as the *Jacobian* matrix for the model; and section 7 calculates the loop elasticities for the time periods in the data table. While it is possible to explore the intermediate steps in each of these sections, the sections are not intended for user inspection. Instead, the last two sections present the analysis output in easy to interpret formats.

Section 8 reports the evolution of eigenvalues through the different time instances where they were evaluated. The real and imaginary parts of the eigenvalues can be inspected in tabular and graphical form and it is also possible to obtain a plot with the eigenvalues in the complex plane for each time frame.

Tables 1 and 2 report the real and imaginary parts of the 10 eigenvalues (one per independent state variable) of our sample model across the 10 annual evaluations. There are several things to note in these tables. First, one of the eigenvalues is 0 throughout the simulation. This is consistent with the fact that one of the stocks in the model (M) does not change in the base run. Second, all eigenvalues have a negative real part, which means that the system is dampened and all perturbations will eventually die off. This is consistent with the behavior observed in the two main stocks (Figure 5). Third, there are two pairs of complex eigenvalues {3,4} and {7,8} each representing a different frequency of oscillation. The first pair represents an oscillatory behavior mode with a period of 4.27 years ($2\pi/\mathrm{Im}[\lambda] = 2\pi/1.47$) that corresponds to the business cycle, and the other corresponds to the capital cycle with a period of almost 30 years ($2\pi/0.21$).

Eigenvalue		1	2	3	4	5	6	7	8	9	10
Time	0	-16.000	-3.208	-0.571	-0.571	-0.400	-0.398	-0.160	-0.160	-0.022	0.000
	1	-16.000	-3.217	-0.488	-0.488	-0.406	-0.400	-0.227	-0.227	-0.029	0.000
	2	-16.000	-3.212	-0.537	-0.537	-0.401	-0.400	-0.188	-0.188	-0.025	0.000
	3	-16.000	-3.208	-0.579	-0.579	-0.400	-0.398	-0.154	-0.154	-0.021	0.000
	4	-16.000	-3.208	-0.578	-0.578	-0.400	-0.398	-0.154	-0.154	-0.021	0.000
	5	-16.000	-3.209	-0.569	-0.569	-0.400	-0.398	-0.162	-0.162	-0.022	0.000
	6	-16.000	-3.208	-0.570	-0.570	-0.400	-0.398	-0.161	-0.161	-0.022	0.000
	7	-16.000	-3.208	-0.575	-0.575	-0.400	-0.398	-0.157	-0.157	-0.022	0.000
	8	-16.000	-3.208	-0.575	-0.575	-0.400	-0.398	-0.157	-0.157	-0.022	0.000
	9	-16.000	-3.208	-0.574	-0.574	-0.400	-0.398	-0.158	-0.158	-0.022	0.000

Table 1. System Eigenvalues (real part) – evaluated annually

Eigenvalue		1	2	3	4	5	6	7	8	9	10
Time	0	0.000	0.000	1.469	-1.469	0.000	0.000	0.210	-0.210	0.000	0.000
	1	0.000	0.000	1.455	-1.455	0.000	0.000	0.232	-0.232	0.000	0.000
	2	0.000	0.000	1.462	-1.462	0.000	0.000	0.222	-0.222	0.000	0.000
	3	0.000	0.000	1.472	-1.472	0.000	0.000	0.205	-0.205	0.000	0.000
	4	0.000	0.000	1.471	-1.471	0.000	0.000	0.205	-0.205	0.000	0.000
	5	0.000	0.000	1.469	-1.469	0.000	0.000	0.210	-0.210	0.000	0.000
	6	0.000	0.000	1.469	-1.469	0.000	0.000	0.209	-0.209	0.000	0.000
	7	0.000	0.000	1.470	-1.470	0.000	0.000	0.207	-0.207	0.000	0.000
	8	0.000	0.000	1.470	-1.470	0.000	0.000	0.207	-0.207	0.000	0.000
	9	0.000	0.000	1.470	-1.470	0.000	0.000	0.208	-0.208	0.000	0.000

Table 2. System Eigenvalues (imaginary part) – evaluated annually

Finally, it should be noted that all eigenvalues are very stable throughout the simulation, meaning that there are no significant transitions in the model. This stability simplifies the analysis of the linkages between structure and behavior, as there are no significant changes of loop dominance in the trajectories of the base case. Kampmann and Oliva (2006) present a case of a model with significant changes in loop dominance and illustrate how the tools are useful in that context.

Since eigenvalues are fairly stable through out the simulation, I focus on reporting the results of the analysis of loop dominance at time 5, the midpoint of the simulation. The utility, however, is capable of instantaneously generating similar reporting for each of the instances when the computations were realized. Figure 6 shows the system eigenvalues in the complex plain.

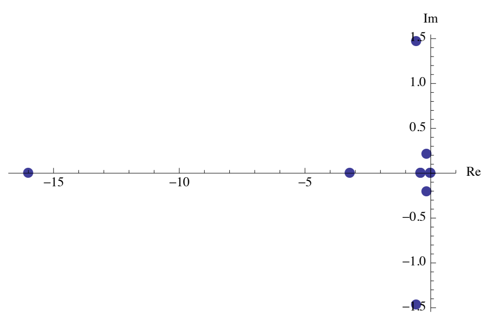


Figure 6. System Eigenvalues at time 5

Section 9 of the LEEA utility reports the impact of the feedback loop structure on the reference modes represented by each of the eigenvalues. For our sample model, the utility identified 27 loops in the SILS (Oliva 2004); see table 3.

Loop 1	AY>cAY
Loop 2	EMP>cEMP
Loop 3	K>KD
Loop 4	K>KI
Loop 5	K>KD>KI
Loop 6	LED>cLED
Loop 7	LU>cLU
Loop 8	M>RCM
Loop 9	P>cP
Loop 10	PY>cPY
Loop 11	SED>cSED
Loop 12	LED>DIV>DII>A>cLED
Loop 13	IV>DII>A>cSED>SED>Y
Loop 14	LED>DK>KI>FS>A>cLED
Loop 15	IV>DII>A>cLED>LED>DK>KI>FS
Loop 16	IV>DII>A>cLED>LED>DK>KI>K>PTY>Y
Loop 17	AY>R>DK>KI>K>PTY>Y>cAY
Loop 18	EMP>PTY>Y>IV>DII>A>cSED>SED>DEMP>cEMP
Loop 19	M>R>DK>KI>FS>A>cSED>SED>DEMP>PT>RCM
Loop 20	EMP>PT>RCM>M>R>DK>KI>FS>A>cSED>SED>DEMP>cEMP
Loop 21	M>R>DK>KI>FS>A>cSED>SED>DEMP>PT>TMS>RCM
Loop 22	SED>DEMP>PT>CGS>G>FS>A>cSED
Loop 23	EMP>U>cLU>LU>PT>CGS>G>FS>A>cSED>SED>DEMP>cEMP
Loop 24	EMP>U>cP>P>R>DK>KI>FS>A>cSED>SED>DEMP>cEMP
Loop 25	PY>C>FS>A>cSED>SED>Y>CDY>cPY
Loop 26	PY>C>FS>A>cSED>SED>Y>T>CDY>cPY
Loop 27	PY>C>FS>A>cSED>SED>DEMP>PT>CGT>GT>CDY>cPY

Table 3. A Shortest Independent Loop Set

The utility offers the option to report the loop gains for all the loops in the SILS. The gain values, however, are contingent on the magnitudes of the variables traversed by the loop, and it becomes difficult to make meaningful comparisons between loops. The influence metric reported in the following subsection addresses this shortcoming. Nonetheless, it should be noted that in the base case, loop 8, loops 19 through 23, and loop 27 have a gain of zero as the policy trigger (PT) is inactive and there are no changes in the M stock.

As discussed above, the focus of the analysis will be on the model's oscillatory behavior modes, i.e., the two pairs of complex eigenvalues. Figure 7 shows the LEEA utility's output for the loop influence on eigenvalue 3—the behavior mode with a 4.27 years period representing the business cycle. The utility has an option to control the number of loops to be included in this report; in this case the figure includes the top 10 most influential loops as measured by the absolute value of the influence metric (see eq. 5)—regardless of its sign, loops with large influence metrics are more influential. To simplify the analysis, the utility sorts the loops in descending order of influence and reports the influence measure in the first row of the table.

Loop	11	13	2	18	15	25	14	12	24	26
Abs	1.062	0.69	0.548	0.499	0.237	0.231	0.209	0.108	0.076	0.069
Real Part	-0.938	0.113	-0.536	0.274	-0.196	-0.225	0.207	0.087	0.074	0.067

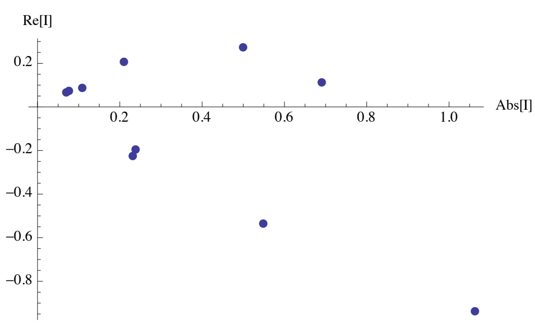


Figure 7. Loop Influence (top 10 loops) on Eigenvalue 3 at time 5

It is also of interest to determine whether a particular loop acts as a stabilizing or destabilizing influence on the behavior mode. The sign of the real part of the influence metric determines this role, i.e., a negative real part implies a stabilizing (dampening) effect on the behavior mode, while a positive real part implies a destabilizing (exponential growth) behavior mode. The real part of the influence metric is reported in the second row of the output table. To facilitate interpretation of the role of each loops and its relationship to other influential loops, the utility generates a scatter plot of the influence metrics placing the absolute value of the influence in the x-axis and the real part in the y-axis. The further from the origin a loop is the more influential it is. Points below (above) the x-axis represent stabilizing (destabilizing) loops.

From Figure 7, it is clear that the business cycle is destabilized by loops 13 and 18 (the rapid adjustment of inventory and the interaction between inventory and employment) and is stabilized by loops 11 and 2 (the slow adjustments of expected demand and inventory). Each of the stabilizing loops has links within the destabilizing loops.

Similar analysis (see Figure 8) reveals that the capital cycle (the oscillatory behavior mode with a period of 30 years represented by eigenvalue 7) is destabilized by loops 15, 25 and 11 (the multiplier effects of capital and consumption, and the short term response to estimated demand) and it is stabilized by loops 6, 10 and 13 (the smoothing process to adjust Long Term Estimated Demand, the Adjustment of Permanent Income, and the short term adjustment of inventory).

Loop	15	25	11	6	10	13	4	2	14	18
Abs	0.229	0.222	0.211	0.202	0.196	0.195	0.151	0.115	0.111	0.081
Real Part	0.225	0.21	0.171	-0.192	-0.097	-0.19	-0.141	0.111	-0.105	-0.041

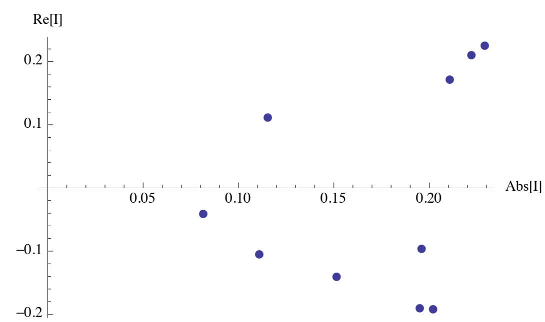


Figure 8. Loop Influence (top 10 loops) on Eigenvalue 7 at time 5

This explanation of the two oscillatory behavior modes is consistent with the explanation provided by Forrester in his thesis (1982), and also by the explanation obtained from analyzing similar models that include the labor and capital interactions (Mass 1975, Oliva and Kampmann 2010). However, it should be noted that all these insights and a full structural explanation of the behavior of the system was generated out of a single run of the model as opposed to the exhaustive exploration through sensitivity analysis.

From this analysis one could derive policy recommendations within the structure of the system. That is, the analyses reveals what loops need to be weakened or strengthen in order to bring more stability to the system. It is easy to identify the parameters that control the gain for each of the loops. For example, decreasing the parameter values of time to smooth short-term demand (tssd) and time to adjust employment (tae) would increase the gain of loops 11 and 2 respectively. By increasing the gain of these two loops that have a stabilizing influence in the business cycle, one would further dampen the business cycle.

The above analysis, however, has the disadvantage that it only identifies leverage points within the active structure of the system. That is, LEEA cannot “see” beyond the active elements of the system and none of influential loops are part of what Forrester described as part of the policy levers — not surprising since in this initial simulation the gain on those loops was set to zero.

The ability of the method to assess the full model behavior at once, however, sets up the opportunity to assess several potential policy recommendations at once. While Forrester (1982) tested each of his five policies individually, given this tool one can assess them all at once and get the full benefit of understanding their interactions. I turn to this possibility in the following section.

Full Model

A full version of the Forrester model, including the activation of the policy triggers, is also available in the electronic supplement (`NF_model_full.mdl`). The model is structurally the same as the model used in the base case, but switches and parameter have been updated to activate feedback loops 8, 19 through 23, and 27. As with the base case, the model is initialized in disequilibrium—IV stock 3% below its equilibrium level. Figure 9 shows the behavior of aggregate output and the inventory, capital and employment stocks under the full model (`full.vdf`) and compares each to the base case simulation (`base.vdf`).

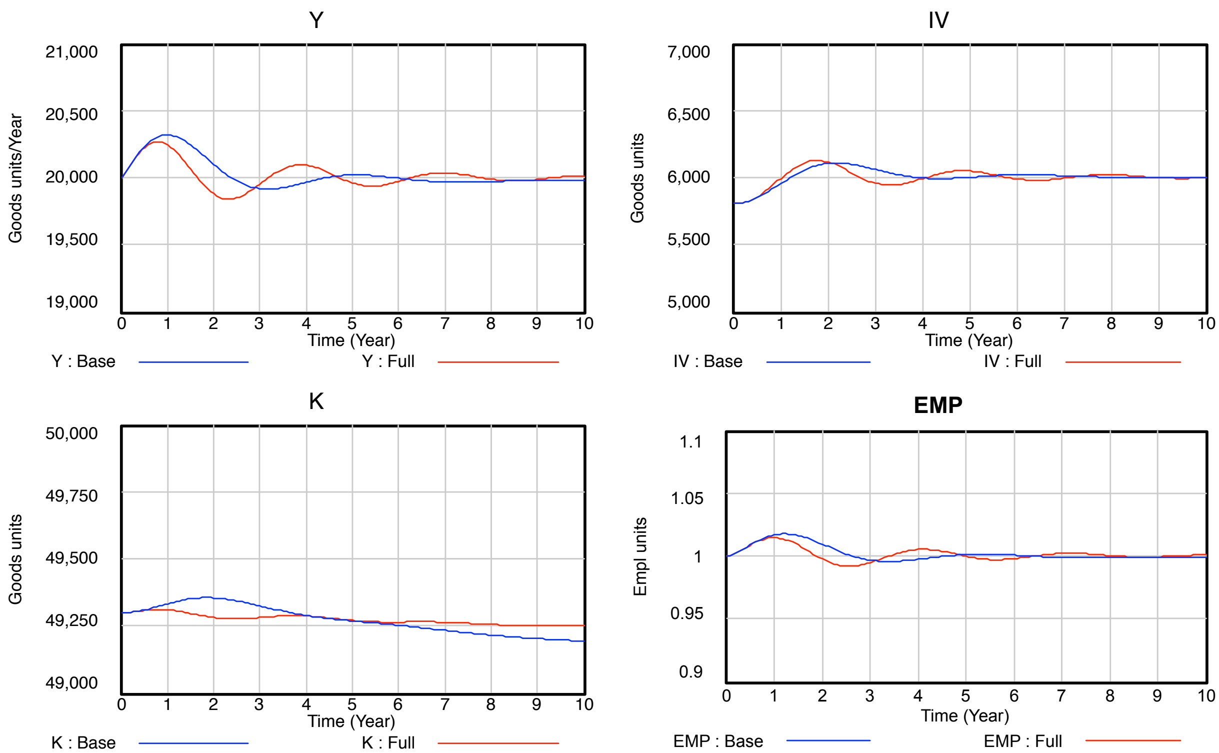


Figure 9. Full run of the Forrester model

With all the intervention policies active at the same time, the frequency of the system's response increases and the dampening ratio decreases relative to the base case. That is, the combination of all the five policies acting simultaneously makes the system respond faster and more aggressively to deviations from equilibrium. While the faster response prevents the capital stock (K) to deviate from equilibrium as much as in the base case, as a result of this aggressiveness, the system overreacts to those deviations and now the stocks in the business cycle (IV and EMP) take longer to reach equilibrium. Aggregate production (Y) follows closer the response of the business cycle in almost the same phase as the employment stock.

As discussed above, to conduct a meaningful policy analysis, it is necessary to specify a set of criteria for what constitutes a successful policy change. I will avoid this challenge, and instead focus on eliminating unwanted instabilities (oscillations), by either increasing the damping of oscillatory behavior modes by making the real part more negative or decrease the frequency of oscillation. From this perspective, the combination of policies introduced in the full model seem to be effective in increasing the dampening and reducing the frequency of the capital cycle, but have the opposite effect on the business cycle. The LEEA analysis can help us understand why these tradeoffs are taking place and focus on the subset of policies that might yield a better balance of these tradeoffs.

After preparing the Vensim® files into the appropriate format (`NF_model_full.mdl` → `nf_model_full.nb` and `full.vdf` → `full.tab`) I ran them through the LEEA Mathematica® utility performing the linearization and full analysis of elasticities at one year intervals.

Again, the system eigenvalues are stable through the simulation, thus I focus on a single period. Table 4 reports the eigenvalues of the system at time 5. As this time the M stock is active in the system, all eigenvalues are non-zero. All real parts of the eigenvalues remain negative, thus the system remains dampened. More interestingly, the system now shows three complex pairs of eigenvalues representing three separate oscillatory behavior modes for the system. Eigenvalue pairs {1,2}, {4,5} and {8,9} denote oscillatory modes with a periods ($2\pi/\mathrm{Im}[\lambda]$) of 5.96, 3.15 and 37.6 years respectively.

Eigenvalue	1	2	3	4	5	6	7	8	9	10
Real	-16.104	-16.104	-3.055	-0.440	-0.440	-0.454	-0.385	-0.250	-0.250	-0.007
Imaginary	1.054	-1.054	0.000	1.993	-1.993	0.000	0.000	0.167	-0.167	0.000

Table 4. System Eigenvalues (full model) at time 5

Analysis of the loops influencing these reference modes reveals the impact of the implemented policies. Figure 10 shows the most influential loops on eigenvalue 1 (period 5.96 years). Only loops 8 and 7 (long term labor adjustment and monetary adjustment) have a significant influence on this behavior mode—both loops are stabilizing. Loop 8, as mentioned above, was not active in the base simulation, and while loop 7 was active, long-term unemployment (LU) had no further effect on the model, as the policy trigger (PT) was not activated. Thus, this behavior mode is strictly the result of the policies introduced. This is in itself an interesting finding, as in addition to affecting the two existing oscillatory behavior modes (see discussion below), the policy implementation in itself introduces an oscillatory pattern with a period of six years. The two influential loops in this reference mode are first order delays on stocks that determine the intensity of the policy trigger (LU), or the speed of the policy adjustment (M), and their gain is determined by identical time constants.

Loop	8	7	22	23	21	2	15	14	11	19
Abs	7.894	7.863	0.523	0.521	0.474	0.096	0.094	0.089	0.073	0.048
Real Part	-7.888	-7.827	0.07	-0.101	-0.064	-0.026	0.025	-0.024	-0.019	-0.007

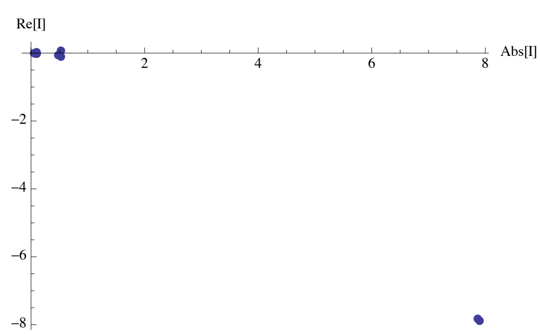


Figure 10. Loop Influence (top 10 loops) on Eigenvalue 1 at time 5 (full model)

This oscillatory behavior mode represented by eigenvalues {4,5} (period 3.15 years) is most influenced by two stabilizing loops (11 and 2) (see Figure 11). These are the same two loops identified as dampening the business cycle on the base simulation and they have almost the same effect on the eigenvalue. Comparing this influence scatter plot with the one for the business cycle in base case (Figure 7), two major differences become apparent. First, loop 13, the short-term adjustment of inventory, changes from destabilizing to marginally stabilizing. Both the magnitude and the destabilizing effect (the size of the real part) of this loop have been reduced with the introduction of the policies. It is interesting to note that loop 13 has no active links to the 7 loops introduced with the policy recommendations but its change in relative influence is the result of other loop interactions. Second, loop 23, the counter cyclical government spending policy, is now a significant destabilizing influence in the business cycle. While the policy increases aggregate demand (A) in order to affect demand expectations and long term employment (EMP) towards faster equilibrium, the policy has a short term effect of depleting the existing inventory (IV)—through final sales (FS)—that further exacerbates the disequilibrium between aggregate production and demand, causing inventories to drop.

Loop	11	2	13	7	23	18	14	15	8	22
Abs	1.008	0.685	0.557	0.455	0.446	0.349	0.300	0.262	0.241	0.239
Real Part	-0.898	-0.630	-0.021	-0.110	0.052	0.211	-0.175	0.242	-0.034	-0.047

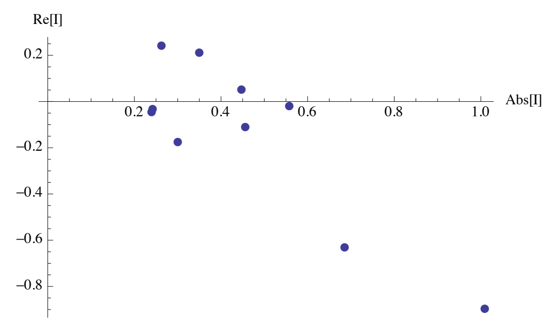


Figure 11. Loop Influence (top 10 loops) on Eigenvalue 4 at time 5 (full model)

This behavior mode from eigenvalues {8,9} is the same as the behavior mode captured by eigenvalues {7,8} in the base simulation. Figure 12 shows the 10 most influential loops on this reference mode. Comparing the this scatter plot with the one from the base case (Figure 8), it is clear that the destabilizing influence of loops 15 (capital adjustment), 25 (demand from disposable income), and 11 (short term demand expectations) has been significantly reduced – the real part of the influence metric of these three loops has been reduced by 24%, 35%, and 70% of their original values. The diminished role of loops 25 and 11 implies that stabilizing loops 10 and 13 also loose their influence, as loop 10 shares links with loop 25 and loop 13 works interacts with loop 11. Loops 6 (capital investment) and 4 (long term demand expectations) retain their influence in the stabilizing effect on this behavior mode as the two first order loops dampen the response of all capital acquisition. The net effect of the elimination of the destabilizing loop is the reduction of frequency of the capital cycle (the period increases from 30 to 37 years).

Loop	6	11	4	2	25	13	15	26	1	16
Abs	0.183	0.147	0.139	0.098	0.077	0.056	0.056	0.051	0.044	0.042
Real Part	-0.176	0.058	-0.139	0.054	0.075	-0.021	0.039	-0.049	0.028	-0.013

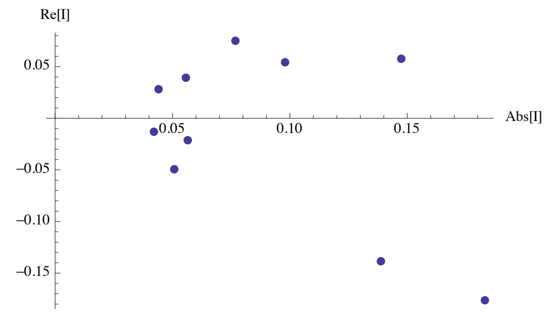


Figure 12. Loop Influence (top 10 loops) on Eigenvalue 8 at time 5 (full model)

Again, this analysis of the full model can be used to design the policy that would improve the results, looking, for instance, to reduce the strength of the countercyclical government spending as to mitigate its destabilizing effect on the business cycle while retaining the stabilizing effect of all policies in the capital cycle. The fact that the gains of all the loops that are being analyzed can be independently set (since this is what defines Independent Loop Set) immediately focuses our attention on the parameter that uniquely affects the gain of the countercyclical government spending loop, i.e. the Strength of the Countercyclical Government Spending (SCGS).

A quick test reveals that reducing the strength of SCGS does indeed have the desired effect in the system, reducing the frequency of the oscillations in the inventory stock, without affecting the oscillations in the capital stock. Figure 13 shows the effect of changing SCGS from 1.00 to 0.25 on the behavior of these two stocks. While the effect of the parameter change is in the right direction, it is clear that its leverage on the actual behavior on the stocks of interest is very limited as a reduction of 75% of the strength of this policy only marginally reduces the frequency of the oscillations in the business cycle. An exploration of the projection of each of the eigenvalues in the stocks of interest, and an assessment of the impact of the parameter values in this projection has been found to be much more effective for policy design. The next section addresses the details of the Dynamic Decomposition Weight (DDW) analysis.

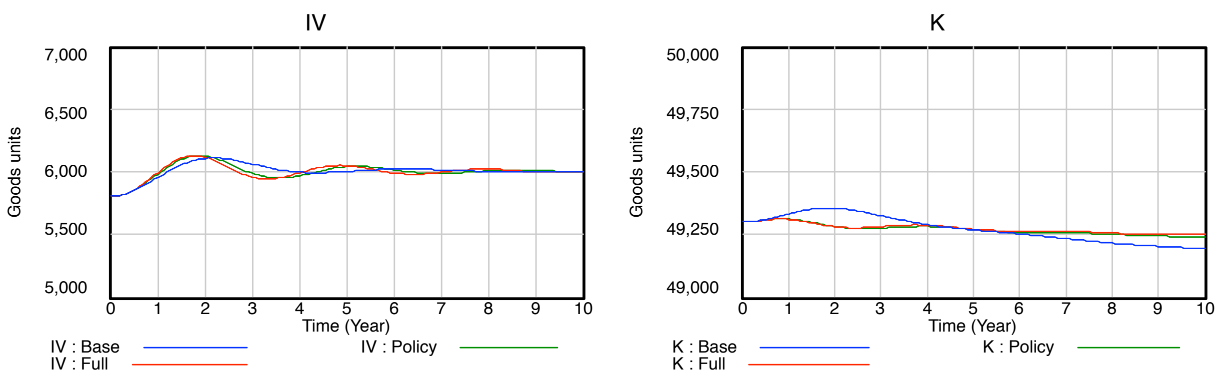


Figure 13. Effect of policy to reduce the strength of SCGS from 1.00 to 0.25

DDWA

Full Model

Unlike the LEEA utility, the Mathematica® implementation of the DDWA utility (`DDWA.nb`) only requires the Mathematica® version of the model[7], as the elasticities of all parameters are estimated from the model's initial conditions. The electronic supplement provides detailed instructions for preparing this input file.

The first two sections of the DDWA notebook contain the instructions to use the utility and some functions required by the utility. The following three sections are identical to corresponding sections in the LEEA utility. Section 3 derives the graph representation of the model structure. Sections 4 and 5 symbolically derive the edge and loop gains as well as the *Jacobian* matrix for the model[8]. Section 6 performs the computations for the dynamic decomposition weight (see eq. 3 above) and reports graphs of the decomposition of the behavior of each of the stocks decomposed to each of the behavior modes represented by the eigenvalues. The graphs are available in absolute stock values, or normalized by dividing by the constant term of eq. 3, thus making the contribution of each eigenvalue comparable.

Section 7 performs the core computations to determine the elasticities of the weights (*w* in Eq. 3) to parameters values and links and section 8 provides different reporting options for these computations. There are three options to report the parameter and link elasticity tables: i) reporting by stock, ii) reporting by behavior mode, and iii) reporting with eigenvalue elasticity. The first option allows the user to focus on a particular stock and a table with the elasticity of all DDWs of that stock (the *w*s in equation 3) to all parameters (links) is displayed. The second option allows the user to focus on a particular behavior model (eigenvalue) and a table of the elasticity of all DDWs of that eigenvalue to all parameters (links) is displayed. The final option reports, for a selected stock and behavior mode, the weight elasticity as well as the elasticities of the real and imaginary parts of an eigenvalue to each parameter (link).

From the LEEA of the full model, we are interested in reducing the frequency of oscillation of the business cycle ($-0.44+1.99i$)[9]. The stocks involved in the feedback loops responsible for this behavior mode are inventory (IV), short-term expectation of demand (SED) and employment (EMP) (loop 18). Table 5 reports, in descending order, the elasticities of the DDW on the inventory stock and real and imaginary part of the business cycle eigenvalues to all the model parameters.

---

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

[8] This is clearly a replication of computational effort. I plan to integrate these two analyses into a single utility in the near future.

[9] Note that the value of the eigenvalue reported in table 5 is slightly different from the value reported in table 4. This is because table 4 reports the eigenvalues at time 5 whereas the analysis in table 5 is based on the values of the eigenvalues at time 0. The easiest way to perform the DDW analysis for that particular point in time would be to initialize the model at the values the full simulation shows at that point in time. This is something that should be addressed once the two analyses are incorporated into a single platform.

Stock IV

Ref. Mode $-0.439089 + 1.99189i$

Parameter	Weight	Re[Eigenvalue]	Im[Eigenvalue]
tai	1.2254	1.1389	-0.438051
tssd	1.0904	-1.4915	-0.397835
ee	0.254223	0.697591	-0.362646
tsld	-0.222606	0.370957	0.0956871
eyvm	-0.182839	-0.170836	0.237425
ep	0.182604	0.17063	-0.23703
alk	-0.136176	-0.377402	0.152125
tae	-0.122809	-0.736998	-0.228529
iem	0.110846	0.165356	-0.129712
lr	-0.104886	-0.130388	0.119828
scms	-0.071869	-0.00533037	0.107508
scgs	-0.0625046	-0.0122594	0.104935
egs	-0.0625046	-0.0122594	0.104935
tak	0.059149	0.224975	-0.0590739
nic	0.0530342	-0.215016	-0.0223606
sgyt	-0.0430347	-0.191731	0.02593
tsu	-0.0323641	-0.132002	-0.000771187
fcu	-0.0313859	0.628563	0.124721
alpha	0.0295238	0.0122131	-0.0831417
tam	-0.0271463	-0.0701878	0.0121287
stm	0.0271116	0.0701027	-0.0121211
tsay	0.0251853	0.0494694	-0.0134502
yem	-0.0193851	-0.0591123	0.0108077
ey	-0.0170366	-0.535925	0.0571362
scgt	-0.0140052	-0.0994481	-0.000994437
egt	-0.0140052	-0.0994481	-0.000994437
spc	-0.0123833	-0.100541	0.000594068
scmg	-0.00729961	-0.000539694	0.0109192
tsy	-0.00663401	0.0137172	0.0143647
nru	-0.00305242	-0.0888955	-0.0113284
sdc	0.	0.	0.

Table 5. Elasticity of business cycle eigenvalue and DDW on inventory stock to model parameters

Quick inspection of the table reveals the reason for the limited impact of the changes in SCGS on the frequency of interest discussed in the previous section. The parameter has only a marginal effect of the DDW on the inventory stock (elasticity = -0.0625, ranked 12th in the table), and the elasticity of the eigenvalue to this parameter is only marginal. Parameters higher in the table will have a more dramatic impact on the specific stock behavior mode combination. As an example, Figure 14 shows the effect of increasing the length of the time to adjust inventory (TAI), the top parameter in the list, form 0.4 to 0.8 on the inventory and capital stocks.

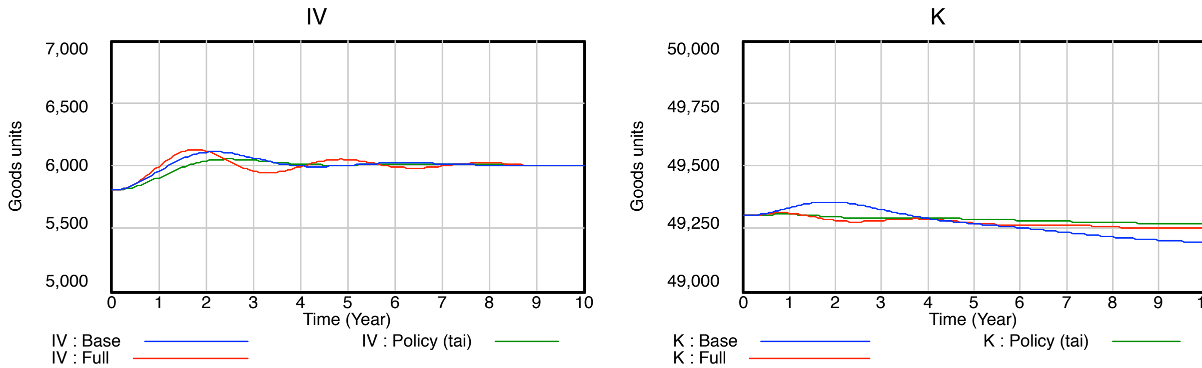


Figure 14. Effect of policy to increase the time to adjust inventory (tai) from 0.4 to 0.8

Clearly, the tool cannot judge the feasibility of implementation of the different policy changes, but the comprehensive assessment of the impact of all parameters (links) on all stocks and reference modes allows for a very rapid identification of the parameters with the highest leverage on a particular behavior mode of a specific stock.

Discussion and Conclusion

The two methods described in this chapter, Loop Eigenvalue Elasticity Analysis (LEEA) and Dynamic Decomposition Weight Analysis (DDWA) are predicated on the idea that it is possible to linearize, around a particular operating point, a non-linear model. Once that linearization takes place, it is possible to characterize the system behavior in terms of the eigenvalues of the system matrix. LEEA identifies the feedback loops that are responsible for each of the behavior modes represented by each eigenvalue. As such, it is a powerful tool to formally establish the relationships between model structure and observed behavior. This linkage between model structure and behavior is critical to dynamic modelers, not only in that we now have certainty on what are the structural elements responsible for the behavior, but also as a general map of the feedback loops that are crucial for policy analysis.

DDWA uses the system matrix eigenvectors to identify a closed form projection (weight) of each behavior mode on the state variables. By assessing the elasticity of these weights to model parameters (links) it is possible to formally identify the parameters (policies) with the highest leverage on a particular behavior mode of a state variable.

While the linearization of the model represents an approximation, our experience from having analyzed dozens of models is that the approximation is quite valid whenever the model is reasonably close to the linearization of the operating point and the proposed applications (identification of dominant structure and high leverage policies) do not require the numerical precision that is affected by this approximation.

The benefits of the method are significant. First, it is a formal analysis and linkage of the model structure and behavior. The closed form solutions used by these methods make the analyses traceable, programmable and replicable. This means that even novice modelers can benefit from the power of the insights generated by the methods. Second, the methods are comprehensive in that they assess the overall behavior of the system and its structure. The simultaneous evaluation of all model structure and behavior allows assessment of how different pieces of structure behave in the context of the overall system, i.e., with other structural pieces in place, something that is clearly lost with partial model simulations. Third, the methods are efficient; the analysis of a single simulation reveals model structural insights that used to take hundreds of simulations and sensitivity analysis to develop. Finally, the methods are effective in that they have consistently replicated previous analysis, behavior narratives, and policy design analysis.

The methods, however, have some clear limitations and disadvantages. The first limitation, and probably the most significant in terms of hampering the broad adoption of the methods, is that the interpretation of the analysis output requires some basic understanding of control theory and linear algebra. Second, the linearization step requires that all table functions must be analytical and continuously differentiable (C^1). Future model parsers could eventually address this limitation and while in most cases this only requires an additional step to express the table function, this requirement reduces the flexibility and spontaneity of testing different formulations in the model. Third, although there are no known computational limitations, the calculation of eigenvalues and eigenvectors, especially through multiple linearization points, might be computational intensive. Finally, as currently implemented, the parser does not support macros, arrays, and most dynamic functions.

The above limitations still make the analysis an "expensive" undertaking. In a recent analysis of a previously-build, large (13 stocks, 44 auxiliaries, 33 parameters, 34 loops in the ILS) model with random inputs, it took ~90 seconds to run the utilities to perform the LEEA and DDWA and about 45 minutes to interpret the results and prepare a report (Oliva 2014). However, preparing the model for the utilities (i.e., formulating SMOOTH functions explicitly, eliminating MIN, MAX and IF_THEN_ELSE statements, and replacing table functions with analytical forms) took more than 8 hours. While 8 hours is significantly shorter time than the weeks it took the author to develop an intuition for the model behavior, this kind of overhead makes these analyses a post-modeling exercise, rather than an integral part of the model-building/learning process. While the 'model preparation' stage would have been much shorter if the software limitations had been considered when building the model, adherence to these requirements would have limited the developer's appetite for testing the model and exploring alternative formulations. Clearly, reducing the overhead imposed by the current limitations of the experimental tools here presented is a major leverage point for the broader adoption of these methods.[10] Adoption of formal analysis of models' behavior will not only make modelers and analysts more efficient, but would also improve the overall quality of the dynamic modeling work.

Challenge

The following is a series of challenges for the reader to develop a better intuition of the analyses output as well as a way to explore the different reporting options of the utilities.

- Loops 17 and 24 are the only loops that contain the interest rate (R) that are active in the base case. What is the role of these loops in that base run? What behavior modes do they affect?
- How does the role of R change in the full model? What behavior models is it affecting?
- What parameter changes (policies) would you introduce to augment (diminish) the impact of the interest rate (R) on the business cycle? On the capital cycle?

Answers to these questions and a description of a strategy on how to go about answering them are available in the `Challenge.pdf` document in the electronic supplement.

[10] All Mathematica® notebooks are open for inspection of algorithms and partial output, and the translator Perl code is available for downloading in the utility's website.

Acknowledgments

The development of these tools and approaches to EEA has been the result of a long-term collaboration with my friend Christian E. Kampmann. He could not participate in the preparation of this chapter, but I gratefully acknowledge his contributions to these ideas. All errors and omissions are my own. I also thank Burak Güneralp, Alejandro Serrano, and the handbook editors for comments that greatly improved the manuscript.

Bibliography

Chen, C. 1970. *Introduction to Linear System Theory*. New York: Holt, Rinnehart and Winston.

Dornbusch, R., S. Fischer, and R. Startz. 2010. *Macroeconomics*. 11th ed, *The McGraw-Hill Series Economics*. New York: McGraw-Hill.

Duggan, J., and R. Oliva. 2013. "Methods for identifying structural dominance—Introduction to the model analysis virtual issue." *System Dynamics Review* (Virtual Issue):http://onlinelibrary.wiley.com/journal/10.1002/(ISSN)1099-1727/homepage/VirtualIssuesPage.html.

Forrester, J.W. 1961. *Industrial Dynamics*. Cambridge, MA: Productivity Press.

Forrester, N.B. 1982. "A Dynamic Synthesis of Basic Macroeconomic Theory: Implications for Stabilization Policy Analysis."PhD Thesis, Massachusetts Institute of Technology.

Gonçalves, P. 2009. "Behavior modes, pathways and overall trajectories: eigenvector and eigenvalue analysis of dynamic systems." *System Dynamics Review* 25 (1):35-62.

Güneralp, B. 2006. "Towards coherent loop dominance analysis: progress in eigenvalue elasticity analysis." *System Dynamics Review* 22 (3):263-289.

Kampmann, C.E. 2012. "Feedback loop gains and system behavior (1996)." *System Dynamics Review* 28 (4):370-395.

Kampmann, C.E., and R. Oliva. 2006. "Loop eigenvalue elasticity analysis: Three case studies." *System Dynamics Review* 22 (2):141-162.

Kampmann, C.E., and R. Oliva. 2008. "Structural dominance analysis and theory building in system dynamics." *Systems Research and Behavioral Science* 25 (4):505-519.

Kampmann, C.E., and R. Oliva. 2009. "Analytical methods for structural dominance analysis in system dynamics." In *Encyclopedia of Complexity and Systems Science*, edited by R. Meyers, 8948-8967. New York: Springer.

Luenberger, D. 1979. *Introduction to Dynamic Systems: Theory, Models and Applications*. New York: Wiley.

Mass, N.J. 1975. *Economic Cycles: An Analysis of Underlying Causes*. Cambridge, MA: Productivity Press.

Mojtahedzadeh, M.T. 2008. "Do parallel lines meet? How can pathway participation metrics and eigenvalue analysis produce similar results?" *System Dynamics Review* 24 (4):451-478.

Mojtahedzadeh, M.T., D.F. Andersen, and G.P. Richardson. 2004. "Using *Digest* to implement the pathway participation method for detecting influential system structure." *System Dynamics Review* 20 (1):1-20.

Oliva, R. 2004. "Model structure analysis through graph theory: Partition heuristics and feedback structure decomposition." *System Dynamics Review* 20 (4):313-336.

Oliva, R. 2014. "Structural dominance in large and stochastic system dynamics models." International System Dynamics Conference, Delft, The Netherlands, July, 2014.

Oliva, R., and C.E. Kampmann. 2010. "Toolset for eigenvalue elasticity analysis." Accessed Apr. 15, 2014. http://iops.tamu.edu/faculty/roliva/research/sd/leea/toolset.html.

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

Sterman, J.D. 2000. *Business Dynamics: Systems Thinking and Modeling for a Complex World*. New York: Irwin McGraw-Hill.