A Comprehensive Eigenvalue Analysis of System Dynamics Models

Mohamed Saleh
Decision Support Department
Cairo University
Email: m.saleh@fci-cu.edu.eg

Pål Davidsen
System Dynamics Group
University of Bergen
Email: davidsen@ifi.uib.no

Khaled Bayoumi
Information Technology Institute
Cairo, Egypt
Email: kbayoumi@iti.net.eg

Abstract

In this paper, we develop a comprehensive eigenvalue analysis for linear models, in order to identify the leverage points in models. The analysis is comprehensive as we develop a closed-form analytic function relating the behavior of any state variable to all parameters in the model. Moreover, by decomposing the behavior into several modes of behavior – each characterized by an eigenvalue and an eigenvector – it is possible to develop a closed-form analytic function relating a certain mode of behavior to all parameters in the model. In the first section of this paper, we explain the mathematical foundation of eigenvalue analysis. In the second section we identify the structural origin of the modes of behavior. This enables us to pinpoint the leverage points of the model. Finally, in the last section, for illustration purpose, we apply the method to a linearized version of the market growth model. The analysis of this linearized model enables us to explain the model behavior as a superimposition of a number of behavior modes, and set the stage for analyzing the original, non-linear version of the market growth model.

Keywords: linear model analysis, eigenvalue analysis**,** leverage points

I. Mathematical Background

The first step in the analysis is to express the behavior as a summation of several modes of behavior, i.e. to decompose the behavior into several modes of behavior. In this paper, we will decompose the time trajectory of a state variable into several modes of behavior. The time trajectory of the state variable is a mathematical function that specifies the value of the state variable at any time instant.

The point of departure is the structure of the model. The structure of any linear model can be represented by the following compact matrix equation:

$$\dot{\bar{x}} = G\,\bar{x} + \bar{b} \qquad (1)$$

Where:
$\bar{x}$ is the vector of state variables.
$\dot{\bar{x}}$ is the vector of first time derivates of state variables, which is also called the slope vector, $\bar{s}$.
$\bar{b}$ is a constant vector.
G is the gain matrix, which is defined as follows:

$$G = \begin{bmatrix} \partial\dot{x}_1/\partial x_1 & \partial\dot{x}_1/\partial x_2 & . & \partial\dot{x}_1/\partial x_n \\ \partial\dot{x}_2/\partial x_1 & & & . \\ . & & & . \\ \partial\dot{x}_n/\partial x_1 & . & . & \partial\dot{x}_n/\partial x_n \end{bmatrix}$$

Each element, g_{ij} ($\partial\dot{x}_i/\partial x_j$), in the above matrix constitutes a compact gain; i.e. the change in the net rate (slope) of state variable i, in response to an infinitesimal change in the level (value) of state variable j, in the model.

Note that, in this paper, we denote scalars using variables in small letters; vectors in over-scored, small letters; and matrices in capital letters.

Differentiating equation (1) with respect to time yields:

$$\ddot{\bar{x}} = G\,\dot{\bar{x}}$$

That is:

$$\bar{c} = G\,\bar{s} \qquad (2)$$

Where
$\bar{c}$ is the curvature vector, which is the vector of second time derivates of state variables; i.e. $\ddot{\bar{x}}$.
$\bar{s}$ is the slope vector, which is the vector of first time derivates of state variables; i.e. $\dot{\bar{x}}$.

That is the gain matrix, G, relates the slope vector to the curvature vector. In other words, the gain matrix, G, transforms the $\bar{s}$ vector into the $\bar{c}$ vector in an n-dimensional standard space. The solution to the system of differential equations, specified by equation (2), provides us with the time trajectory of the slope vector of the model. In general, each curvature, c, is a linear combination of all the slopes (i.e. function of the entire slope vector, $\bar{s}$). The implication is that the curvature, c, associated with a single coordinate (axis) in the standard space is determined by the slopes associated with all the coordinates (component mixing). Therefore, to be able to solve for the time trajectory of the slope, we change the coordinate system using the eigenvalue method.

From the gain matrix, G, we can derive the eigenvalues and the right eigenvectors, as, per definition;

$$G\ \bar{r}_k = \lambda_k\ \bar{r}_k$$

If G is an n x n matrix (i.e. we have n state variables), we will have n eigenvalues, λ_k, each associated with a right eigenvector, $\bar{r}_k$. The default case is to have n distinct eigenvalues, and consequently the right eigenvectors will be linearly independent (Luenberger, 1979), and span an n-dimensional space, R^n; i.e. the right eigenvectors will form a new coordinate system, which is known as the "*eigen-space*". Note that, in general, the coordinates in this system are not orthogonal, yet they are of unity length. The eigenvectors only specify a variety of directions in space along which the dynamics of the model unfold (as explained below).

As the right eigenvectors span the n-dimensional space, then the slope vector, $\bar{s}$, (at any point of time) can be expressed as a linear combination of the right eigenvectors; i.e.:

$$\bar{s} = \alpha_1\ \bar{r}_1 + \alpha_2\ \bar{r}_2 + \ .... + \alpha_n\ \bar{r}_n \qquad (3)$$

In this new coordinate system (eigen-space) the alphas (α_k) will be the *new* components of the slope vector.

Differentiating equation (3) with respect to time yields:

$$\bar{c} = \dot{\alpha}_1\bar{r}_1 + \dot{\alpha}_2\bar{r}_2 + ... + \dot{\alpha}_n\bar{r}_n$$

In this new coordinate system, $\dot{\alpha}_i$ will be the *new* components of the curvature vector.

Substituting $\bar{s}$ -- as expressed in equation (3) -- into equation (2) yields:

$$\bar{c} = G\ [\ \alpha_1\ \bar{r}_1 + \alpha_2\ \bar{r}_2 + \ .... + \alpha_n\ \bar{r}_n\ ]$$

By rearranging, we obtain:

$$\bar{c} = \alpha_1 G\ \bar{r}_1 + \alpha_2 G\ \bar{r}_2 + \ldots + \alpha_n G\ \bar{r}_n$$

By utilizing the fact that $G\ \bar{r}_k = \lambda_k\ \bar{r}_k$, we obtain:

$$\bar{c} = \alpha_1 \lambda_1\ \bar{r}_1 + \alpha_2 \lambda_2\ \bar{r}_2 + \ldots + \alpha_n \lambda_n\ \bar{r}_n$$

And as:

$$\bar{c} = \dot{\alpha}_1 \bar{r}_1 + \dot{\alpha}_2 \bar{r}_2 + \ldots + \dot{\alpha}_n \bar{r}_n$$

Then, along a particular coordinate (specified by a right eigenvector), the dynamics that takes place, can be described by the following differential equation:

$$\dot{\alpha}_k = \lambda_k \alpha_k$$

The solution of the above differential equation is:

$$\alpha_k = \alpha_k^0\, e^{\lambda k\ (t-\tau)}$$

Where τ is the initial time (i.e. the starting time of the analysis period); and α_k^0 are the initial values of α_k (at time τ).

It is clear that the only factor determining the dynamics along a particular coordinate (i.e. a right eigenvector) is the eigenvalue associated with that coordinate itself. Substituting the solution of the dynamic behavior of each alpha (α_k), into equation (3) yields the time trajectory of the slope, i.e.:

$$\bar{s} = \alpha_1^0\, e^{\lambda 1\ (t-\tau)}\ \bar{r}_1 + \alpha_2^0\, e^{\lambda 2\ (t-\tau)}\ \bar{r}_2 + \ldots + \alpha_n^0\, e^{\lambda n\ (t-\tau)}\ \bar{r}_n \qquad (4)$$

Integrating the above slope trajectory equation, with respect to time, yields:

$$\bar{x} = (\alpha_1^0/\lambda_1)\ \{ e^{\lambda 1\ (t-\tau)} - 1\}\ \bar{r}_1 + \ldots + (\alpha_n^0/\lambda_n)\ \{ e^{\lambda n\ (t-\tau)} - 1\}\ \bar{r}_n + \bar{x}_o \qquad (5)$$

Where $\bar{x}_o$ is a constant vector that contains the initial values of the state variables.

The above equation is our target mathematical function that decomposes the state trajectory into several modes of behavior, each expressed by an eigenvalue with its associated right eigenvector. In the next section, we will interpret this equation.

II. Structural Foundation of behavior

In the first section of this paper, we developed a mathematical equation for the state trajectory behavior (equation 5). In this section, we will identify the origin of each component of the state trajectory. As portrayed in the figure below, the basic components are:

• The eigenvalues (λ_κ)
• The right eigenvectors ($\bar{r}_k$)
• The initial values of alphas (α_k^0)

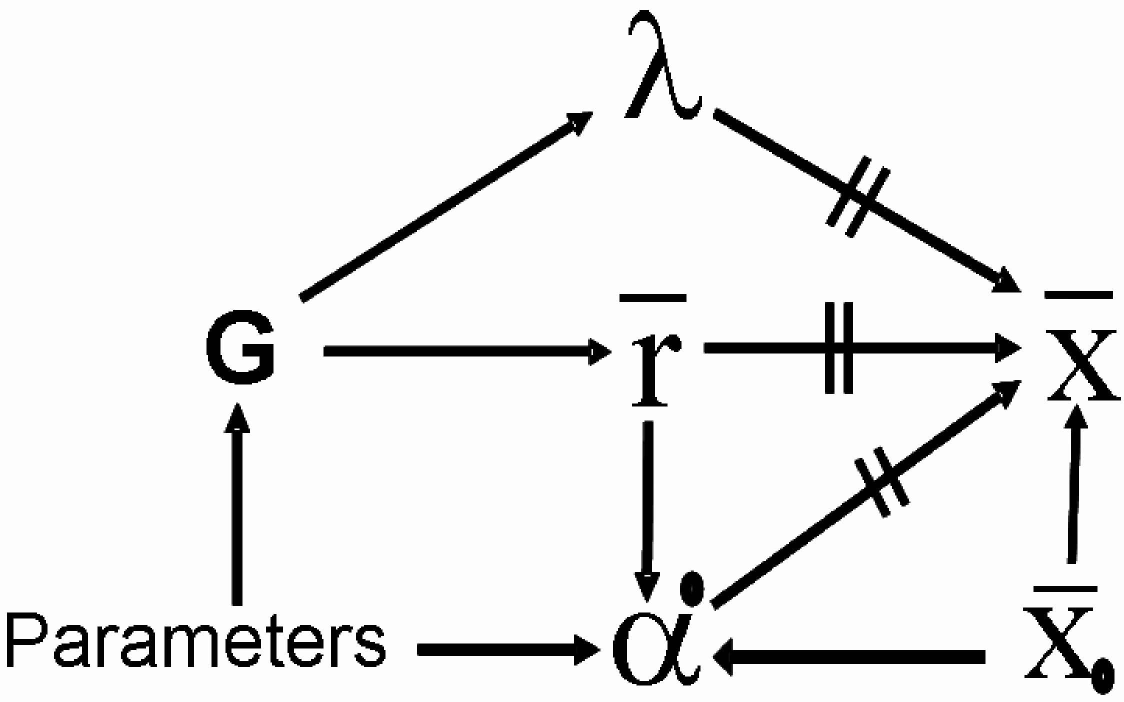

Fig. 1: The origin of the components of the state trajectory

Note that we used delayed links, in the above figure, to indicate that the values of eigenvalues, right eigenvectors, and alphas control the future trajectory of state variables (as indicated by equation 5).

Note that, in our previous, work we have been focusing our efforts solely on identifying the origin of eigenvalues, and ignoring the origin of the eigenvectors (and consequently also ignoring the origin of the alphas). For example in Saleh & Davidsen (2001b) we identified the marginal impact of each parameter, link, and loop, in the business cycles model, to the dominant eigenvalue that was generating the oscillatory behavior observed, in the model. Nevertheless, this partial analysis (i.e. by solely focusing on the dominant eigenvalue) gave us valuable insights about the marginal impacts of each parameter, link, and loop, in the model, on the frequency of oscillation, and on the damping ratio of the envelope within which the oscillatory behavior is seen to unfold. However, in general, eigenvectors can play a significant role in shaping behavior. For example, Saleh & Davidsen (2001a) demonstrated that a simple linear model that has a single positive feedback loop can generate exponential decay behavior, rather than exponential growth; if the initial slope vector was orthogonal to the right eigenvector associated with the positive eigenvalue (that is responsible of generating the exponential growth mode of behavior). For this reason, in this paper, we included the eigenvectors in our analysis.

In the rest of this section, we will explain the origin of each of the basic components of the state trajectory.

Regarding eigenvalues, they originate solely from the structure of the model; or more specifically, from the gain matrix, G. Recall that the gain matrix is used as a condensed representation of the model structure, and was used, in section 1, as the point of departure to drive the state trajectory. Also note that (as portrayed in figure 1), in linear models, the gain matrix, G, depends on the model's parameters (constants in the model). Hence, for each eigenvalue, it is possible to formulate a mathematical function relating the eigenvalue (dependent variable) to the model's parameters (independent variables). Moreover, for simplification, instead of formulating a single complicated function relating an eigenvalue to all parameters in the model, it is possible to develop many mathematical functions; where each function relates the eigenvalue to a single parameter (this is done by substituting the values of the rest of the parameters, in the gain matrix, G). In our research, we utilized the symbolic toolbox in Matlab to automate the above process.

Similarly, as eigenvectors also originate solely from the gain matrix, it is possible to formulate a mathematical function relating any eigenvector (dependent variable) to any parameter (independent variable).

The initial value of each of the alphas (i.e. α_k^0) represents the projection of the initial slope vector along a particular right eigenvector (i.e. a particular coordinate in the eigenspace); i.e. the initial values of the alphas are dependent on the initial values of net rates (the values at the start of the simulation), and the right eigenvectors. The initial values of net rates, in turn, are functions of the initial state of the model (i.e. $\overline{x}_o$), and the model parameters. The right eigenvectors, in turn, are functions of the model parameters (as indicated above). Hence, it is possible to formulate a mathematical function, relating the initial value of each alpha (dependent variable) to any parameter (independent variable). Note that the initial values of sates variable are substituted into this mathematical function.

To summarize, for each component in the state trajectory (i.e. equation 5), it is possible to formulate a mathematical function relating the component to any parameter. This enables us to develop a compound mathematical function relating the future value, at any point of time, of a state variable (dependent variable) to any parameter (independent variable). The partial differentiation, at any point of time, of this compound function with respect to the parameter, yields the sensitivity of the future value (of the state variable) to that parameter. To automate the above process we developed a full-fledged program using the symbolic toolbox in "Matlab" software.

III. Analyzing the Market Growth Model

III.I. Model Overview

The market growth model represents the nature of a single firm competing in a potentially boundless market. The firm consists of three sectors, sales force, order fulfillment, and capacity acquisition. Figure 2 shows the relationship between the firm three sectors and the market sector. The initial version of the market growth model, analyzed in this paper, is taken from the CD of the Business Dynamics book (Sterman 2000); the name of the file is “mktgrow1.sim” (Powersim model). The model is described in details in chapter 15 in the book. We recommend the reader to review the model, and the associated explanation, in the Business Dynamics book. However, we made some modifications to this model (mktgrow1.sim) in order to linearize it. These modifications are described below.

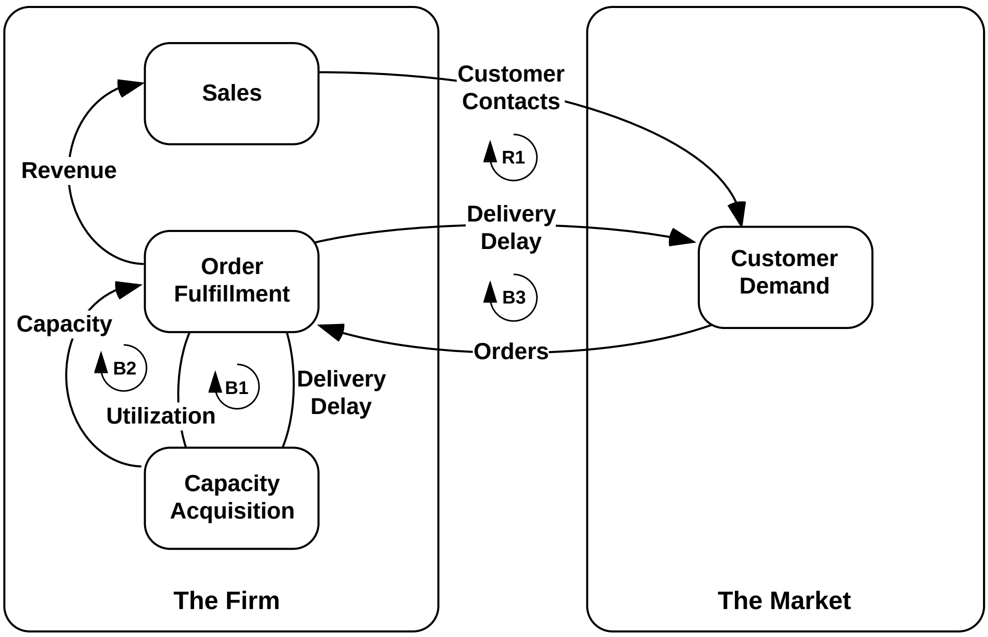

Fig. 2: The sectors of the market growth model

III.II. Model Simplification

Here is the list of modifications which was introduced in order to linearize the model.

1. We assumed that "sales effectiveness" is a constant variable in the model, rather than an endogenous variable whose value is determined by the current state of the model.
2. We assumed that "capacity utilization" is a constant variable in the model, rather than an endogenous variable.
3. We assumed that the "pressure to expand capacity" is based on the "perceived backlog" (by the company) rather than the "perceived delivery delay".

4. We linearized the table function of the "effect of pressure to expand capacity on the desired capacity".
5. We used an additive formulation rather than a multiplicative one for "desired capacity".

For more information about this linearized model the reader may refer to the supplementary powersim model, which is attached to this paper.

III.III. Analysis Results

In this study, we will focus on the Backlog state variable, whose behavior is plotted in the figure below.

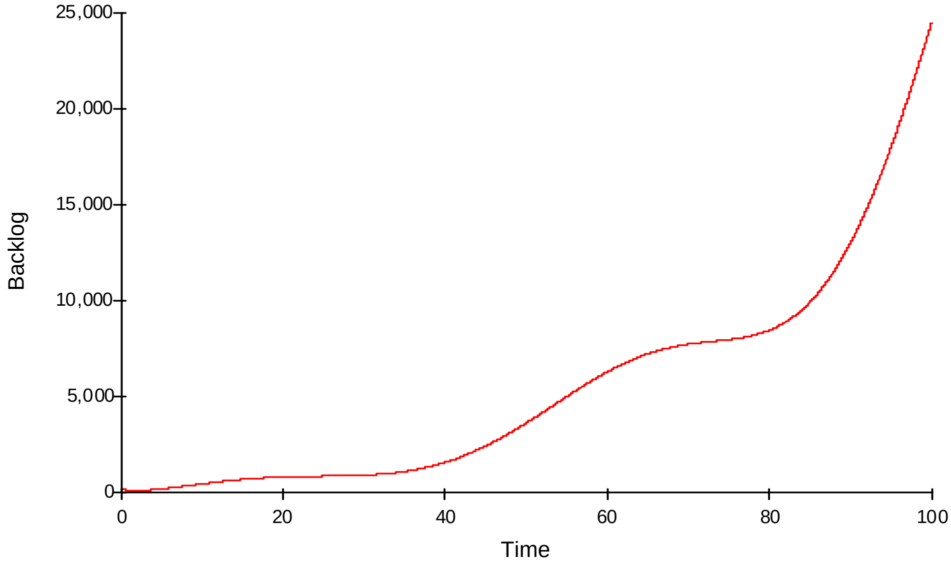

Fig. 3: The behavior of the backlog.

Using the eigenvalue approach, the backlog behavior, can be decomposed into the following modes of behavior:

1. An exponential growth monotonic mode of behavior, shown in figure 4 (characterized by a single eigenvalue).
2. An expanding oscillation mode of behavior, shown in figure 5 (characterized by a pair of complex conjugate eigenvalues).
3. A decaying oscillation mode of behavior, shown in figure 6 (characterized by a pair of complex conjugate eigenvalues).
4. Another decaying oscillation mode of behavior, shown in figure 7 (characterized by a pair of complex conjugate eigenvalues).

Note that superimposing these four modes of behavior yields the total backlog behavior.

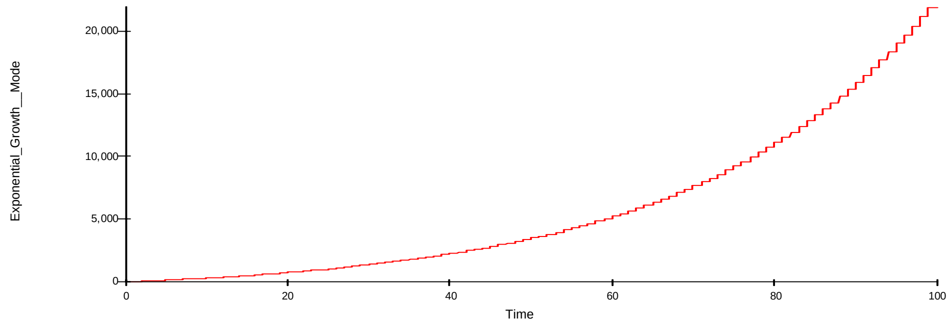

Fig. 4: The exponential growth mode associated with behavior of the backlog

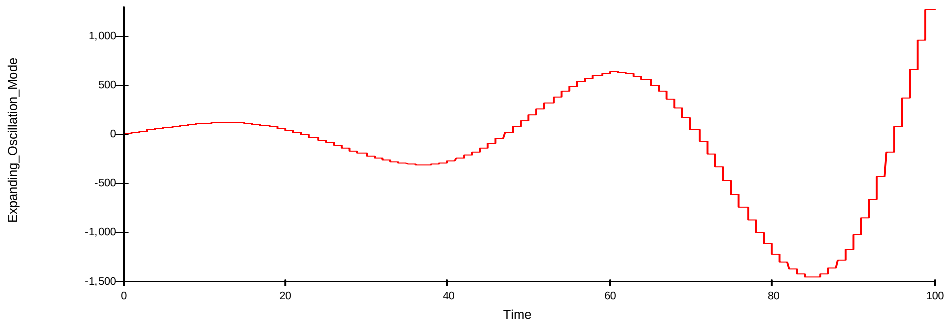

Fig. 5: The expanding oscillation mode associated with behavior of the backlog

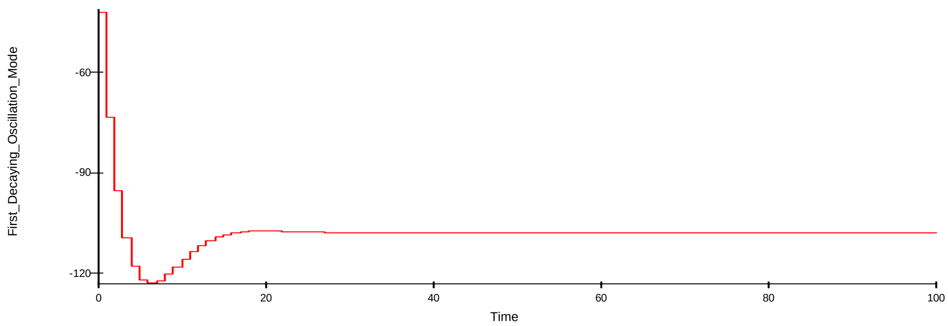

Fig. 6: The first decaying oscillation mode associated with behavior of the backlog

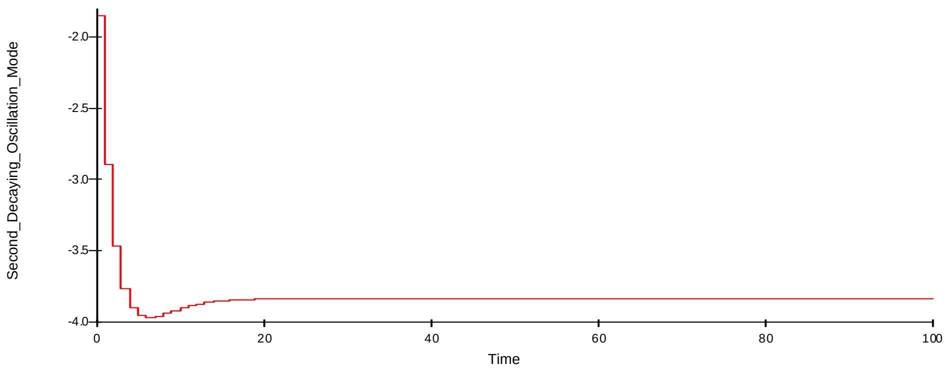

Fig. 7: The second decaying oscillation mode associated with behavior of the backlog

Now, one can compute the sensitivities of the total backlog behavior to the model parameters; or one can compute the sensitivities of a certain mode of behavior (e.g. the exponential growth mode) to the model parameters. In this study, we focused on the sensitivities of the total backlog behavior to the model parameters. However, in principle, one can easily choose a certain mode of behavior to study rather than the total behavior. ***Such a deep analysis can only be performed by using the eigenvalue approach.***

As we mentioned in the previous section, the Matlab algorithm that we developed compute the sensitivity – across time – of the backlog behavior (i.e. the sensitivity of all future values of backlog) to a change in any parameter at the start time of the simulation. In this paper, for demonstration purpose, we will plot the sensitivities of the total backlog behavior to the following seven parameters in this model:

1. CapacityAcquisitionDelay
2. CompanyGoalForBacklog
3. SalesEffectivness
4. CostPerSalesRepresentative
5. Price
6. RevenueReportingDelay
7. SalesForceAdjustmentTime

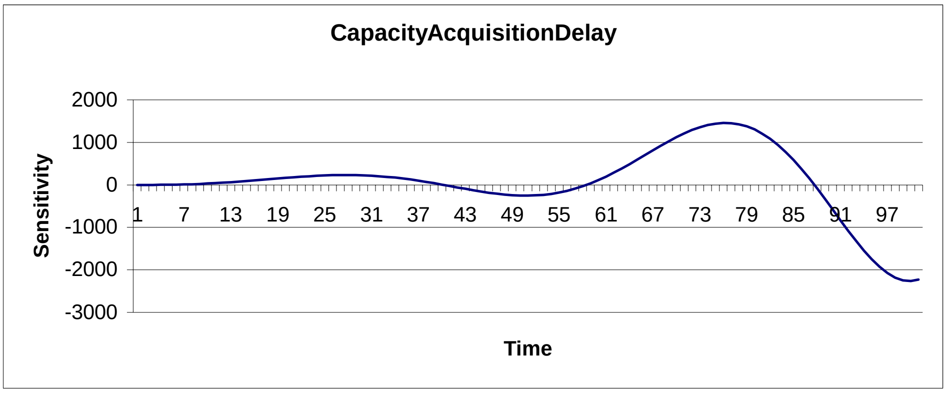

Fig. 8: The sensitivity – across time – of the backlog behavior to CapacityAcquisitionDelay

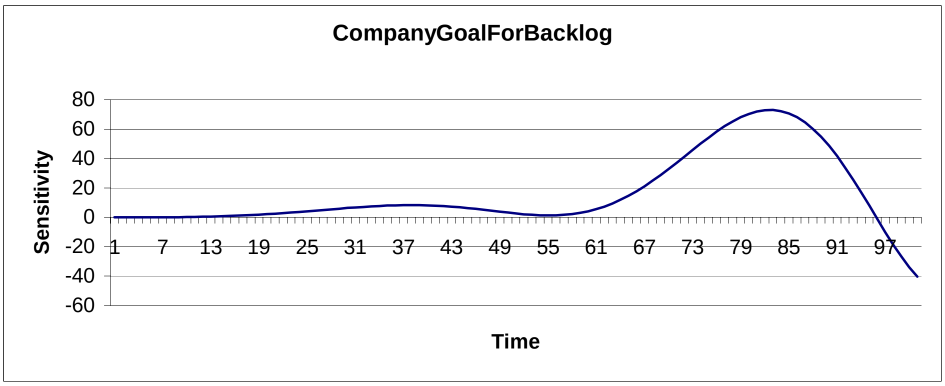

Fig. 9: The sensitivity – across time – of the backlog behavior to CompanyGoalForBacklog

Fig. 10: The sensitivity – across time – of the backlog behavior to SalesEffectivness

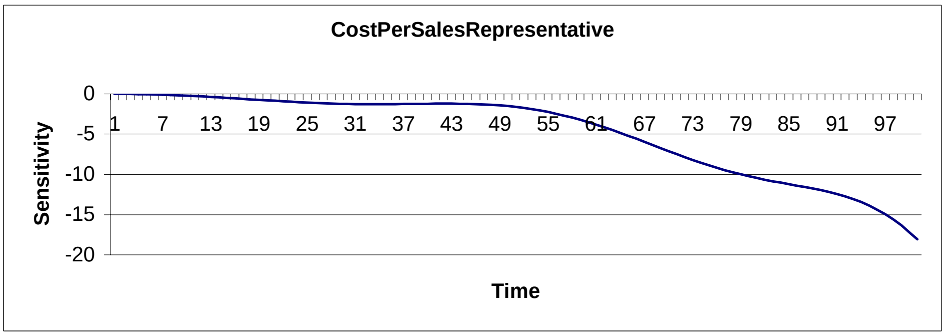

Fig. 11: The sensitivity – across time – of the backlog behavior to CostPerSalesRepresentative

Fig. 12: The sensitivity – across time – of the backlog behavior to Price

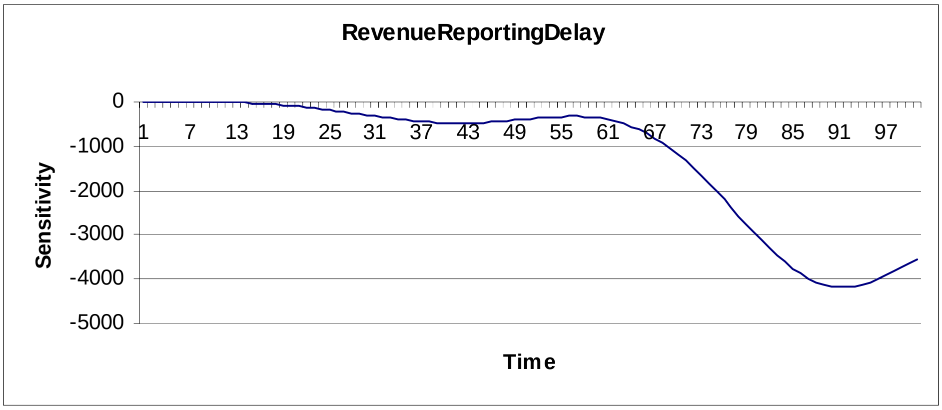

Fig. 13: The sensitivity – across time – of the backlog behavior to RevenueReportingDelay

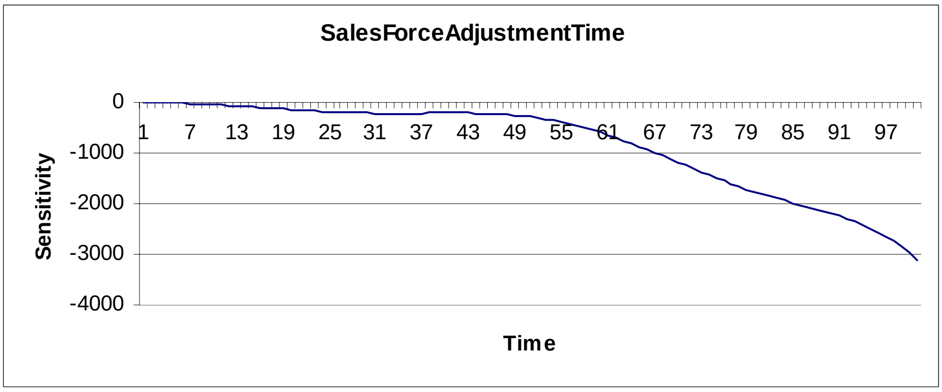

Fig. 14: The sensitivity – across time – of the backlog behavior to SalesForceAdjustmentTime

References

Abdel-Aleem, B. (2004): An automated system to analyze system dynamics models -- Case Study: Commodity Production Cycles. M. Phil. Thesis. University of Bergen. Norway.

Abdel-Gawad, A. (2004): An automated system to analyze system dynamics models. M. Phil. Thesis. University of Bergen. Norway.

Barlas, Y.; Kanar K. (2000): Structure-oriented Behvior Tests in Model Validation. Proceedings of the 2000 International System Dynamics Conference. Bergen.

Davidsen, P. (1991): The Structure-Behavior Graph. The System Dynamics Group, MIT. Cambridge.

Eberlein, R. (1984): Simplifying Dynamic Models by Retaining Selected Behavior Modes. Ph.D. Thesis, M.I.T., Cambridge, MA.

Forrester. N. (1982): A Dynamic Synthesis of Basic Macroeconomic Policy: Implications for Stabilization Policy Analysis. Ph.D. Thesis, M.I.T., Cambridge, MA.

Forrester. N. (1983): Eigenvalue Analysis of Dominant Feedback Loops. The 1983 International System Dynamics Conference, Plenary Session Papers, pp. 178-202.

Ford, D. (1999): A Behavioral Approach to Feedback Loop Dominance Analysis. System Dynamics Review. Volume 15, Issue 1, pp. 3-36.

Goncalves, P.; Lertpattaraong, C.; Hines, J. (2000): Implementing Formal Model Analysis. Proceedings of the 2000 International System Dynamics Conference. Bergen.

Kampmann, C. (1996): Feedback Loop Gains and System Behavior. Proceedings of the 1996 International System Dynamics Conference. Boston.

Kreyszig, E. (1979): Advanced Engineering Mathematics. Fourth Edition. John Wiley & Sons.

Luenberger, D. (1979): Introduction to Dynamic Systems: Theory, Models and Applications. John Wiley & Sons.

Mojtahedzadeh, M. (1996): A Path Taken: Computer-Assisted Heuristics for Understanding Dynamic Systems. Ph.D. Thesis. Rockefeller College of Public Affairs and Policy. Albany NY.

Myrtveit, M.; Saleh, M. (2000): Superimposing Dynamic Behavior on Causal Loop Diagram of System Dynamics Models. Proceedings of the 2000 International System Dynamics Conference. Bergen.

Ogata, K. (1997): Modern Control Engineering. Third Edition. Prentice-Hall.

Press, W.; Teukolsky, S.; Vetterling, W.; Flannery, B. (1992): Numerical Recipes in C. Second Edition. Cambridge Univ. Press.

Reinschke, K. (1988): Multivariable Control: A Graph-theoretical Approach. Lecture Notes in Control and Information Sciences. Springer-Verlag.

Richardson, G. (1984): Loop polarity, loop dominance, and the concept of dominant polarity. System Dynamics Review Vol. 11; pp. 67-88.

Saleh, M.; Davidsen P. (2001a): The origins of behavior patterns. Proceedings of the 2001 International System Dynamics Conference. Atlanta, USA.

Saleh, M.; Davidsen P. (2001b): The origins of business cycles. Proceedings of the 2001 International System Dynamics Conference. Atlanta, USA.

Sterman, J. (2000): Business Dynamics: Systems Thinking and Modeling for a Complex World. McGraw-Hill.