

# **An eigenvalue approach to feedback loop dominance analysis in non-linear dynamic models**

**Mohamed Saleh**

*Information Science Department  
University of Bergen  
E-mail: mohamed.saleh@powersim.no*

**Pål Davidsen**

*Information Science Department  
University of Bergen  
E-mail: davidsen@ifi.uib.no*

## **Abstract**

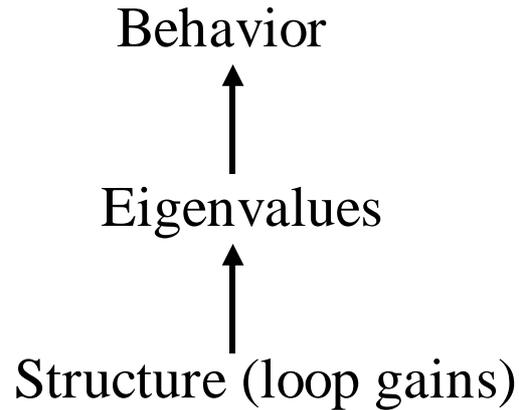
*The purpose of the System Dynamics method is to study the relationship between structure and behavior in non-linear, dynamic systems. In such systems, the significance of various structural components to the behavior pattern exhibited, changes as the behavior unfolds. Changes in structural significance, in turn modifies that behavior pattern, which, in turn, feeds back to change the relative significance of structural components. We develop a mathematical framework within which we can study the characteristics of this feedback between structure and behavior. This framework is based on a piecewise observation of the model over time, a characterization of the behavior pattern exhibited using eigenvalue analysis, and an identification of the relative contribution of each of the loop in the model to each of the eigenvalues that characterize the total behavior, and thus to the total behavior. This work is an extension of the work by Nathan Forrester and Christian Kampmann on the use of eigenvalue analysis in system dynamics. Our main contribution, in this paper, is embedding eigenvalue analysis in a broader analytic framework to capture the transient as well as long-term behavior of non-linear models in general. The mathematical framework developed has been implemented in the form of computer algorithms and tested in a variety of cases.*

*Using this method, we can classify, at any point in time, the feedback loops in a system with respect to their relative significance to the system behavior. This allows us to offer a structural interpretation of the behavior exhibited. Moreover, the method is a key to managing such systems because it allows us to rank, at any point in time, the loops of a model with respect to their significance to the behavior of that model. Thus, as a basis for our management of the system, we may identify the loops that contribute most significantly to the model behavior in a favorable or in an unfavorable way, and, consequently, the loops to strengthen and weaken, respectively, while managing the system.*

## **1 Introduction**

System dynamics is the theory of the relationship between structure and behavior in dynamic systems. One of the most challenging tasks in system dynamics has been to understand how behavior emerges from the underlying structure, i.e. how behavior is created in non-linear models and how that behavior feeds back to change the relative significance of the various loops of the underlying model structure.

In this paper, we will identify the units of analysis of the structure and the behavior of a non-linear model and develop a method by which we can identify the causal relationship between the two. More specifically, we want to attribute the properties of the model behavior, characterized by the ***Behavior Pattern Indexes***, to properties of the underlying model structure, characterized by the ***gains*** of the loops in the model. As indicated in figure 1, we will demonstrate that the ***eigenvalues*** of the model constitute the link between model structure and behavior. On the one hand, they originate solely from the ***loops*** of the structure of the model and, on the other hand, they characterize the behavior of the model.

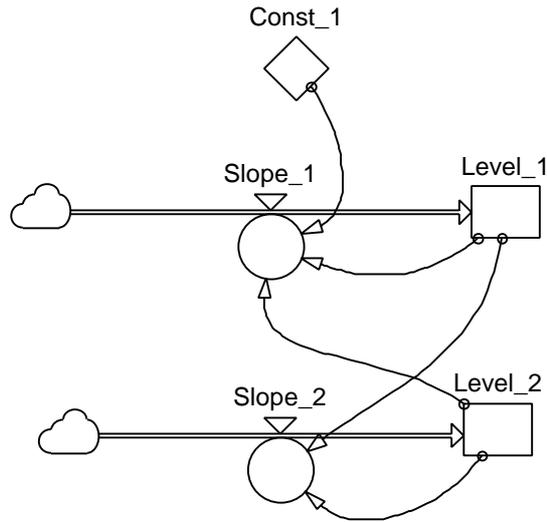


**Fig. 1: Structure drives behavior through the intermediate eigenvalues link.**

We will illustrate the theory presented in this paper with several examples. A simple main example --example 1-- has been incorporated throughout the text to demonstrate the elements of the theory. In addition, we use other, more specific examples to illustrate particular issues raised. At the end of the paper, we present the “yeast cells generation” example to demonstrate all the steps of the procedure that we recommend be followed in the analysis of complex, dynamic systems.

Our main example, example 1, described below, is inspired by a similar example given by Mojtahedzadeh (Mojtahedzadeh, 1996, p. 38) to demonstrate that a traditional eigenvalue analysis, alone, is not sufficient to explain the transient behavior of even such a simple model. He argued that eigenvalue analysis could only be used to study long-term behavior. Our contribution, in this respect, is to demonstrate that eigenvalue analysis can be embedded in a broader analytic framework to capture the transient as well as long-term behavior of non-linear models in general.

Example 1, presented in figure 2 and table 1, is a simple linear, second order model with state variables, Level<sub>1</sub> and 2, governed by the rates Slope<sub>1</sub> and <sub>2</sub>, respectively.



**Fig. 2: Example 1, stock and flow diagram**

**init** Level\_1 = 7  
**flow** Level\_1 = Slope\_1

**init** Level\_2 = 3  
**flow** Level\_2 = Slope\_2

**Slope\_1** =  $(-0.15 * \text{Level}_1) + (-0.2 * \text{Level}_2) + \text{const}_1$   
**Slope\_2** =  $(-0.2 * \text{Level}_1) + (-0.15 * \text{Level}_2)$

**Const\_1** = 0.1

**Simulation Setup Parameters:**  
**Start Time**=0; **Stop Time** =45; **Simulation Time Step** = 0.1

**Table 1: Example 1, equations**

Note: In this paper, we denote *scalars* using variables in small letters; *vectors* in bold, underscored, small letters; and *matrices* in bold, capital letters.

## 2 Behavior pattern indexes

### 2.1 Definitions and justification

The properties of the behavior that we will be focusing on are the slope (s), and the curvature (c) of each state variable (x). These are defined, in this paper, as follows:

- the slope, s, is defined as the (time) derivative,  $\dot{x}$ , of that state variable, x; and

- the curvature,  $c$ , is defined as the double (time) derivative,  $\ddot{x}$ , of that state variable.

The convergence/divergence of a state variable at any instant of time is defined as the rate of change of the absolute value of the slope,  $s$ , of this state variable, i.e.  $d|s|/dt$ . If the state variable is in a convergent mode, then  $d|s|/dt$  will be negative, i.e. the absolute value of the slope,  $|s|$ , is decreasing with time. If the state variable is in a divergent mode, then  $d|s|/dt$  will be positive, i.e. the absolute value of the slope vector,  $|s|$ , is increasing with time. Note that  $|s|=0$  is a partial condition for equilibrium.

As the state of a model changes over time, so do the slopes associated with each of the state variables. Consequently, for a state variable  $x$ , we may express the new slope, i.e.  $s_{x \text{ new}}$ , in terms of the original slope,  $s_{x \text{ current}}$ , and the rate of change in the slope over the subsequent period of time, i.e. the curvature,  $c_x$ .

$$s_{x \text{ new}} = s_{x \text{ current}} + \Delta t * c_x$$

We may now investigate the properties of the ratio:

$$c_x / s_{x \text{ current}}$$

We want to illustrate that this ratio can be considered as a proxy for the  $d|s_x|/dt$  indicator. Note that if  $c_x$  and  $s_{x \text{ current}}$  have the same sign, then this ratio will be positive, and the absolute value of the slope will increase (i.e.  $d|s_x|/dt > 0$ ), - an indicator of divergent behavior. If, on the other hand, the two have opposite signs, then the ratio will be negative, and the absolute value of the slope will decrease (i.e.  $d|s|/dt < 0$ ), - an indicator of convergent behavior. If  $c = 0$ , then the ratio will be 0, so that the absolute value of the slope will not change, i.e.  $d|s_x|/dt = 0$ . In such a case, if  $c_x$  changes its value from  $-\epsilon$  ( $\epsilon \rightarrow 0$ ) to  $+\epsilon$ , then the ratio ( $c_x / s_{x \text{ current}}$ ) will change from negative to positive -- or the reverse, depending on the sign of  $s_{x \text{ current}}$ . As a consequence, the value of  $d|s_x|/dt$  will also change from negative to positive -- or the reverse, - an indicator of a transition in the mode of behavior from convergence to divergence -- or the reverse. If  $s_{x \text{ current}}$  changes its value from  $-\epsilon$  to  $+\epsilon$ , then the ratio ( $c_x / s_{x \text{ current}}$ ) will change from  $-\infty$  to  $+\infty$  -- or the reverse, depending on the sign of  $c_x$ . That would be an indicator of a discontinuity in  $d|s_x|/dt$ , and thus, also in this case, a transition in the mode of behavior from convergence to divergence - or the reverse.

Since the ratio  $c_x / s_{x \text{ current}}$  is characterization of the mode of behavior exhibited by the state  $x$ , we define an indicator of that behavior in the form of a **Behavior Pattern Index (BPI)**, associated with a particular state variable,  $x$ , as:

$$\text{BPI}_x = c_x / s_x$$

$\text{BPI}_x$  serves as a normalized proxy for the  $d|s_x|/dt$ .

For *the model as a whole*, there is an array of slopes, the  $\underline{s}(\dot{\mathbf{x}})$  vector, and curvatures, the  $\underline{c}(\ddot{\mathbf{x}})$  vector. For a model, we define the **overall BPI** to be the angle between the two vectors,  $\underline{s}$  and  $\underline{c}$ . Generally, in a  $n^{\text{th}}$  order model, say, a 300 states model the  $\underline{s}$  and  $\underline{c}$  vectors cannot be visualized (in  $\mathbb{R}^{300}$ ). Yet, any two vectors in  $\mathbb{R}^{300}$  will, however, span a two-dimensional  $\mathbb{R}^2$  subspace of  $\mathbb{R}^{300}$ ; i.e. a plane<sup>1</sup>. In this plane, the angle between the two vectors is real and well defined (measurable). In fact, the angle ( $\theta$ ) between the two vectors  $\underline{s}$  and  $\underline{c}$  is given as:

$$\text{Cos } \theta = \frac{\underline{s} \cdot \underline{c}}{|\underline{s}| * |\underline{c}|} \text{ (Watkins, 1991, p. 137)}$$

Where  $\underline{s} \cdot \underline{c}$  is the inner (scalar) product of the two vectors;  $|\underline{s}|$  is the (Euclidian) length of slope vector; and  $|\underline{c}|$  is the (Euclidian) length of curvature vector.

The angle between  $\underline{s}$  and  $\underline{c}$  is an indication of the divergence or convergence of the model as a whole at any time instance. If the angle (between  $\underline{s}$  and  $\underline{c}$ ) is in the range of  $[0^\circ, 90^\circ)$ , then the model, as a whole, will be in a divergent mode. If, on the other hand, this angle is in the range of  $(90^\circ, 180^\circ]$ , then the model, as a whole, will be in a convergent mode. The condition of a “sustained mode” for the model as a whole is a persistent angle of  $90^\circ$ .

The logic behind the idea of using the angle (between  $\underline{s}$  and  $\underline{c}$ ) as an overall BPI is based on a generalization of the one-state variable case discussed above and goes as follows:

---

<sup>1</sup> The two vectors will span a one-dimensional sub-space (i.e. a line) if they were linearly dependent.

The first step is to explicitly define what we mean by a divergent or a convergent behavior mode for *the model as a whole*. A direct measure of the convergence or divergence of a model at any instant is the rate of change of the length of the slope vector,  $d|\underline{s}|/dt$ . If the whole model is in a *convergent mode*, then  $d|\underline{s}|/dt$  will be negative, i.e. the length of the slope vector,  $|\underline{s}|$ , is decreasing with time. Note that a *necessary condition* for equilibrium of the model is  $|\underline{s}|=0$ . If the whole model is in a *divergent mode*, then  $d|\underline{s}|/dt$  will be positive, i.e. the length of the slope vector,  $|\underline{s}|$ , is increasing with time. And if the whole model is in a *sustained mode*, then the length of the slope vector,  $|\underline{s}|$ , is constant over time.

The next step is to understand how the angle  $\theta$  between the vectors  $\underline{s}$  and  $\underline{c}$  can be seen as a characterization of the rate of change of the length of the slope vector,  $d|\underline{s}|/dt$ , in any  $n^{\text{th}}$  order model. Now, in a very small time interval;

$$\underline{s}_{\text{new}} = \underline{s}_{\text{old}} + \Delta t * \underline{c} \quad (\text{a vector summation}).$$

In the same plane spanned by  $\underline{c}$  and  $\underline{s}$ , we can decompose the vector  $\underline{c}$  into two components, one parallel to the vector  $\underline{s}$ , and the other perpendicular to the vector  $\underline{s}$ . The parallel component is;

$$|\underline{c}| \cos(\theta) (\underline{s}/|\underline{s}|),$$

Where  $(\underline{s}/|\underline{s}|)$  is a unit vector along the same direction as the slope vector. The *length* of the perpendicular component is given as:

$$|\underline{c}| \sin(\theta).$$

If one takes the limit of the equation ( $\underline{s}_{\text{new}} = \underline{s}_{\text{old}} + \Delta t * \underline{c}$ ) as  $\Delta t \rightarrow 0$ , then the perpendicular component of  $\underline{c}$ , will not affect the length of the vector  $\underline{s}$ . It will just shift the direction of the vector  $\underline{s}$ , i.e. it will make vector  $\underline{s}$  rotate around the origo <sup>2</sup>.

---

<sup>2</sup> The angular velocity of the slope vector  $\omega_{\text{slope}}$  (the speed of rotation around the origo) can be calculated from the perpendicular component of  $\underline{c}$ . See example 3 for illustration.

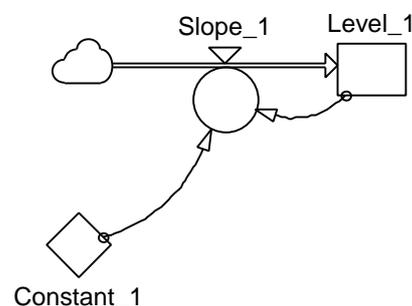
And the only component (of vector  $\underline{c}$ ) that will affect the length of the slope vector,  $|\underline{s}|$ , is the parallel component, i.e.  $|\underline{c}| \cos(\theta) (\underline{s} / |\underline{s}|)$ . Consequently, we can conclude that if  $\theta$  is in the range of  $[0^\circ, 90^\circ)$ , then  $\cos(\theta) > 0$  and  $|\underline{s}|$  will increase, i.e.  $d|\underline{s}|/dt$  is positive. If, on the other hand,  $\theta$  is in the range of  $(90^\circ, 180^\circ]$ , then  $\cos(\theta) < 0$  and  $|\underline{s}|$  will decrease, i.e.  $d|\underline{s}|/dt$  is negative. Moreover, if  $\theta$  equals  $90^\circ$ , then  $\cos(\theta) = 0$  and  $|\underline{s}|$  will be constant, i.e.  $d|\underline{s}|/dt$  is zero.

We may use the overall BPI (angle  $\theta$ ) to characterize the mode of behavior of a model. This is because, using the angle  $\theta$ , one can characterize qualitatively (pos. / neg.) the rate of change of the length of the slope vector,  $d|\underline{s}|/dt$ . Thus the angle  $\theta$  can be considered a proxy for  $d|\underline{s}|/dt$ .

In the rest of this section, we will present several examples to illustrate the use of behavior pattern indexes in SD models.

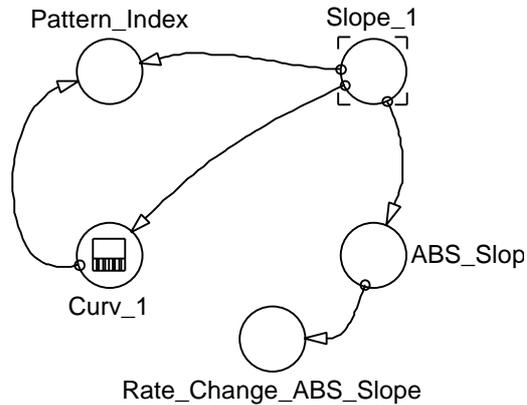
## 2.2 Behavior pattern indexes –example 2

Example 2 is a simple first order model with the state variable Level\_1, governed by the Slope\_1, - a model that can exhibit exponential growth or decay, depending on the value of the parameter Constant\_1. A positive value implies exponential growth, while a negative one implies exponential decay.



**Fig. 3: Example 2, stock and flow diagram of the model**

In order to analyze the model in example 2, we need to add an auxiliary structure, see figure 4, that calculates the pattern index, *Pattern\_Index*, for the state variable based on the first derivative, *Slope\_1*, and second derivative, *Curv\_1*; it also calculates the rate of change of the absolute value of the slope, *Rate\_Change\_ABS\_Slope*:



**Fig. 4: Example 2, stock and flow diagram of the calculation of the behavior pattern index, and the rate of change of the absolute value of the slope**

Below is the table of equations for example 2.

**init** *Level\_1* = 1  
**flow** *Level\_1* = *Rate\_1*

**Slope\_1** = *Constant\_1*\**Level\_1*  
**Curv\_1** = DERIVN(*Slope\_1*)  
*Note: DERIVN is the time derivative function.*

**Constant\_1** = 2 (or -2)

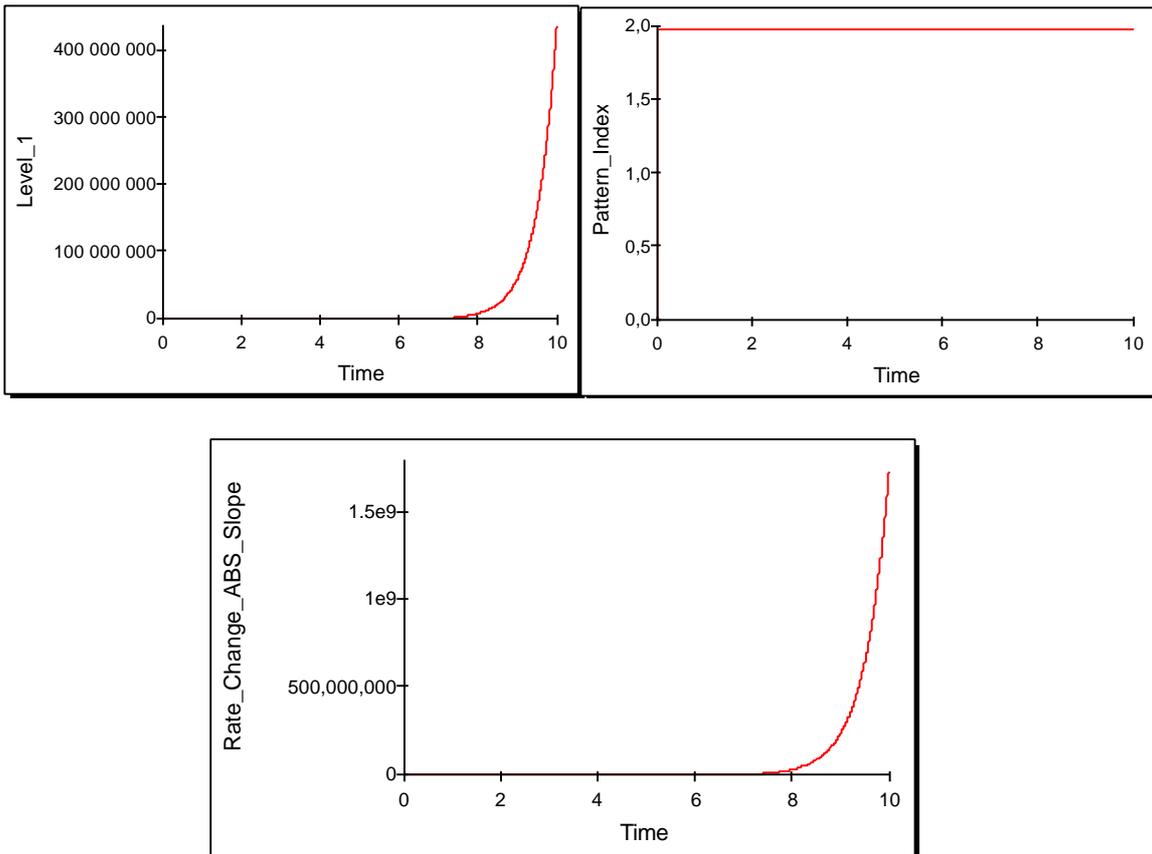
**Pattern\_Index** = *Slope\_1*/*Curv\_1*  
**ABS\_Slope** =ABS(*Slope\_1*)  
*Note: ABS is the absolute function.*

**Rate\_Change\_ABS\_Slope** = DERIVN(*ABS\_Slope*)

**Simulation Setup Parameters:**  
**Start Time=0; Stop Time =10; Simulation Time Step = 0.01**

**Table 2: Example 2, equations**

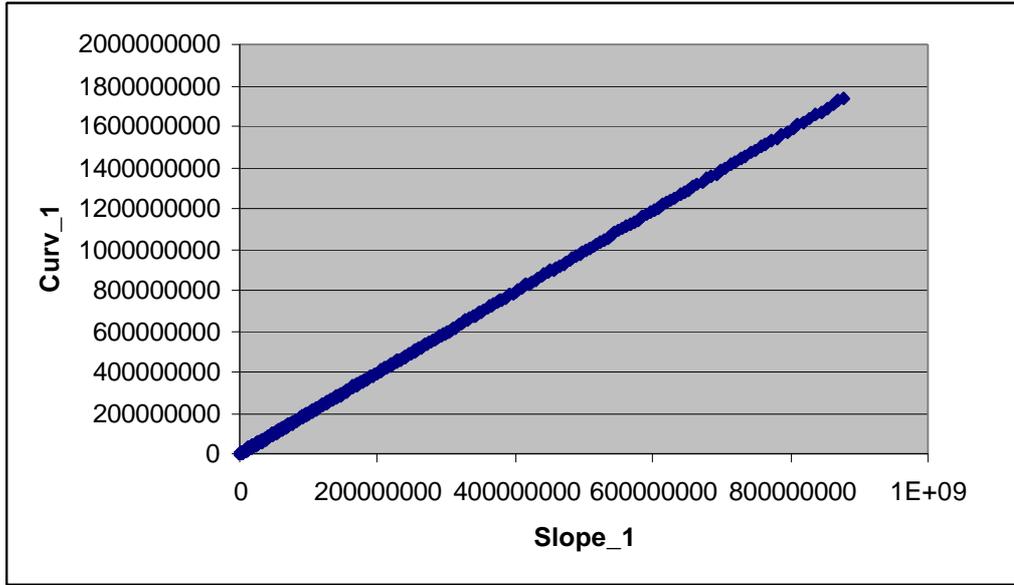
For *Constant\_1* = 2, this model exhibits the following behavior with respect to the state variable (*Level\_1*),  $BPI_{Level_1}$ , and  $d|s|/dt$ :



**Fig. 5: Example 2, behavior, Constant<sub>1</sub>= 2**

Note that the graphs for  $BPI_{Level_1}$  and  $d|s|/dt$  qualitatively provide the same information about the state variable  $Level_1$ , -- that the state variable is divergent since both of them take positive values.  $BPI_{Level_1}$  is constant because it is a normalized expression of the divergence, i.e. of curvature,  $c$  with respect to slope,  $s$ .  $BPI_{Level_1}$  is thus a compact characterization of the mode of behavior exhibited by the state variable.

In figure 6, we portray the relationship between the slope and curvature of  $Level_1$ . This is a linear relationship reflecting precisely the fact that there is a constant positive  $BPI_{Level_1}$ , - i.e. a constant ratio between curvature and slope.



**Fig. 6: Example 2, the relationship between the slope and the curvature of the state variable, Constant\_1= 2**

We now turn to a characterization of the model in example 2 *as a whole*, recognizing the fact that this is a model with only one state variable. In this example, if;

$\underline{s} = [v1]$  is a vector of one element

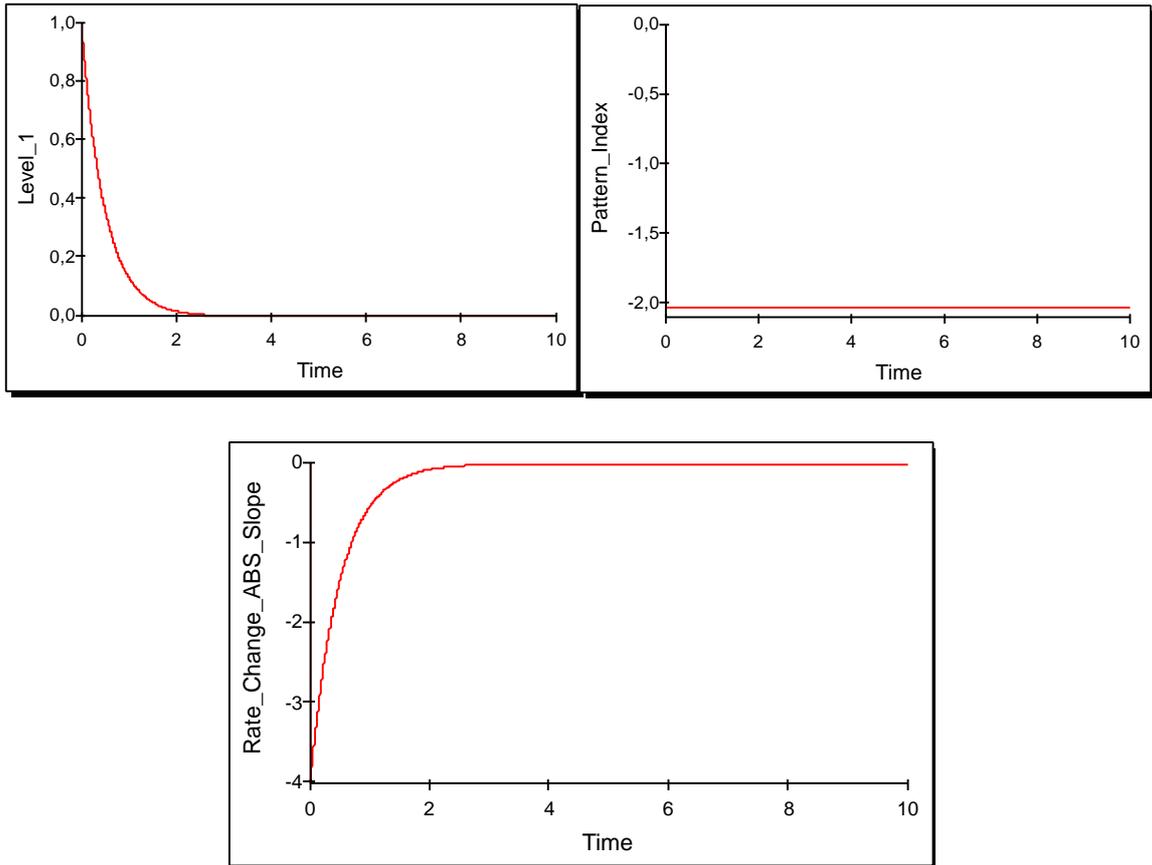
$\therefore \underline{c} = [2 * v1]$  is a vector of one element,

then;

$$\text{Cos } \theta = \frac{\underline{s} \cdot \underline{c}}{|\underline{s}| * |\underline{c}|} = \frac{v1 * 2 * v1}{|v1| * |2 * v1|} = 1$$

Note that the overall BPI for the entire model is defined as the angle  $\theta$  between the curvature and slope vectors. Thus the overall BPI is  $0^\circ$ , which is a qualitative indication of a divergent behavior mode exhibited by the model.

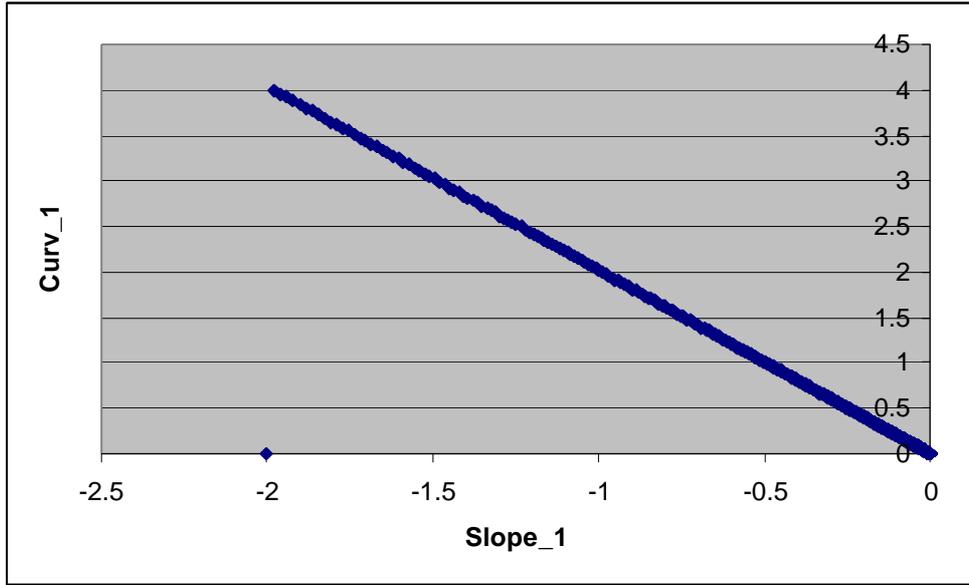
Now, if, in this model, we change the value of Constant\_1 from “2” to “-2”, then we will obtain the following behavior with respect to the state variable (Level\_1),  $\text{BPI}_{\text{Level}_1}$ , and  $d|s|/dt$ :



**Fig. 7: Example 2, behavior, Constant\_1= -2**

Again note that the graphs for  $BPI_{Level_1}$  and  $d|s|/dt$  qualitatively provide the same information about the state variable  $Level_1$ , -- as they both take negative values, indicating that the state variable is convergent.

In figure 8, we once more portray the relationship between the slope and curvature of  $level_1$ . And again there is a linear relationship reflecting the fact that we have a constant, this time negative,  $BPI_{Level_1}$ , - i.e. a constant ratio between curvature and slope.



**Fig. 8: Example 2, the relationship between the slope and the curvature of the state variable, Constant\_1= -2**

Once more, we turn to a characterization of the model in example 2 *as a whole*, - this time for Constant\_1 = -2. In this example, if;

$$\underline{s} = [v1] \text{ is a vector of one element}$$

$$\therefore \underline{c} = [-2 * v1] \text{ is a vector of one element,}$$

then;

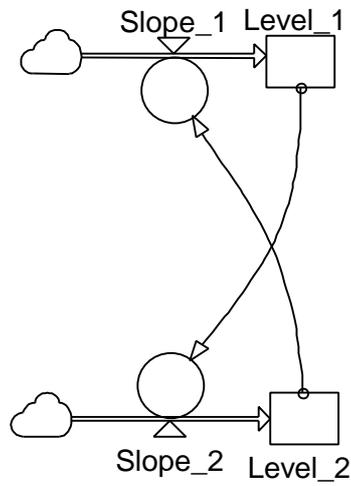
$$\cos \theta = \frac{\underline{s} \cdot \underline{c}}{|\underline{s}| * |\underline{c}|} = \frac{v1 * -2 * v1}{|v1| * |-2 * v1|} = -1$$

Note that the overall BPI for the entire model is defined as the angle  $\theta$  between the curvature and slope vectors. Thus the overall BPI is  $180^\circ$ , which is a qualitative indication of the convergent behavior mode exhibited by the model.

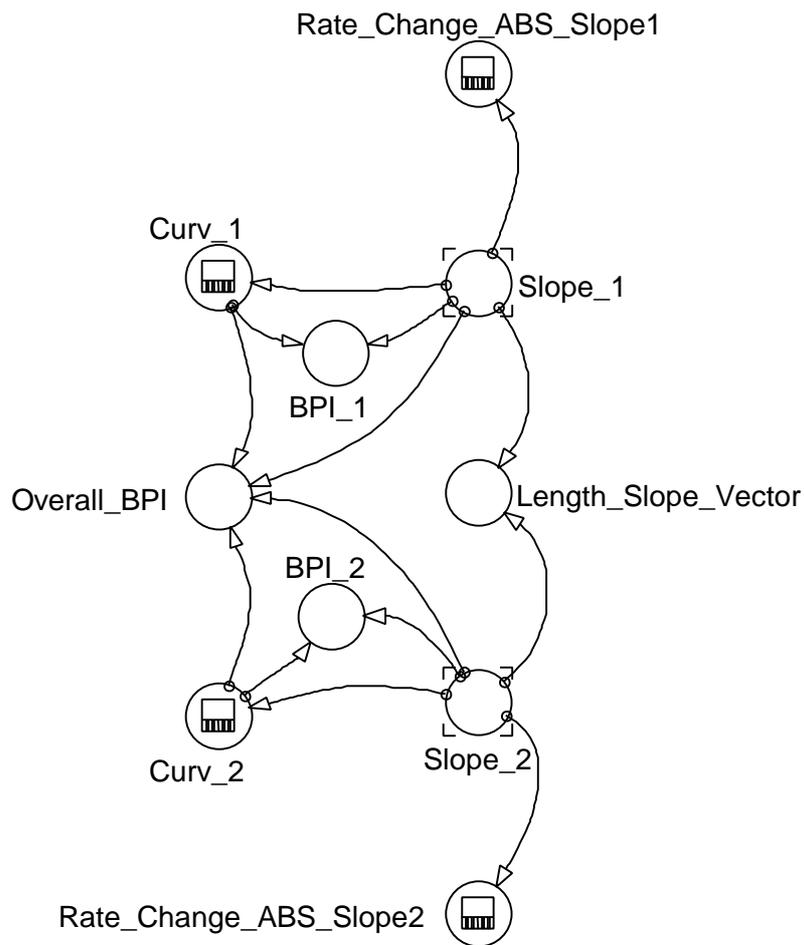
### 2.3 Behavior pattern indexes –example 3

Our next example, example 3 (fig. 9), is a second order model with the state variables Level\_1 and 2, governed by the rates Slope\_1 and \_2, respectively. This example provides us with an opportunity to comment on the distinction between the behaviors of the individual state variables, characterized by BPI\_1 (BPI<sub>Level\_1</sub>) and BPI\_2 (BPI

Level\_2) respectively, and the behavior of the model as a whole, characterized by an overall BPI.



**Fig. 9: Example 3, stock and flow diagram of the model**



**Fig. 10: Example 3, stock and flow diagram of the calculation of the behavior pattern indexes and the length of the slope vector.**

Fig. 10 illustrates the auxiliary structure in example 3, that is responsible for calculating the behavior pattern indexes and the length of the slope vector. Below is the table of equations for example 3.

```

init Level_1 = 1
flow Level_1 = Slope_1

init Level_2 = 1
flow Level_2 = Slope_2

Slope_1 = 0.1*Level_2
Slope_2 = -0.1*Level_1

Curv_1 = DERIVN(Slope_1)
Curv_2 = DERIVN(Slope_2)

BPI_1 = Curv_1/Slope_1
BPI_2 = Curv_2/Slope_2

Overall_BPI = (180/PI)*ARCCOS( ( (Slope_1*Curv_1)+(Slope_2*Curv_2) ) /
                (SQRT(Slope_1^2+Slope_2^2)*SQRT(Curv_1^2+Curv_2^2) ) )

Length_Slope_Vector=SQRT( (Slope_1^2)+(Slope_2^2))

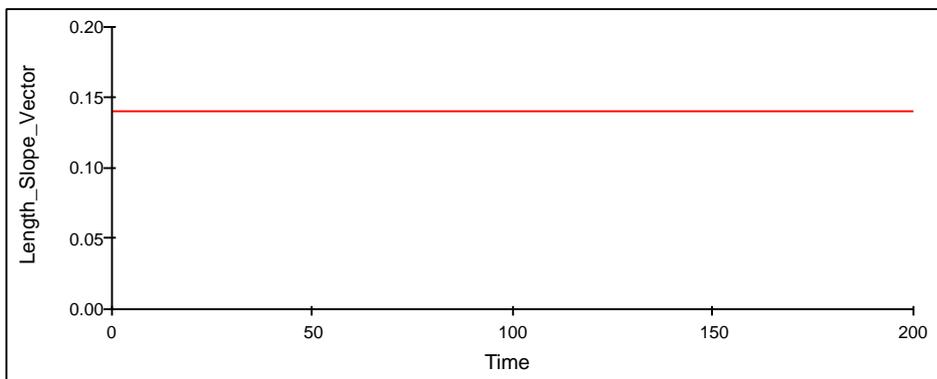
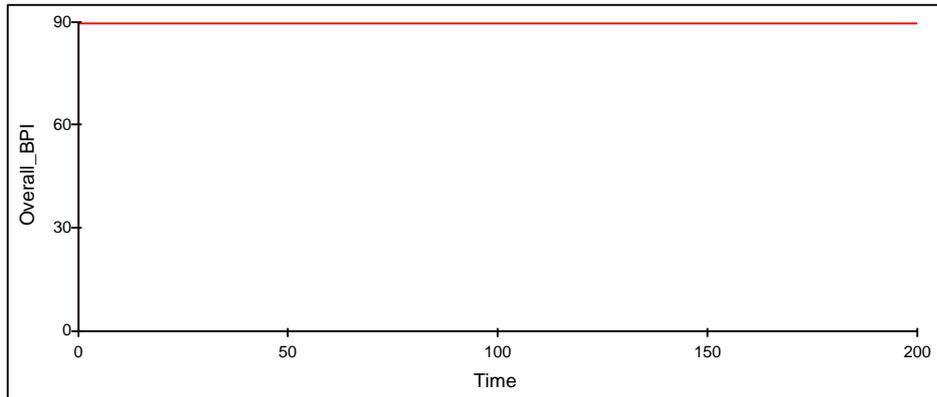
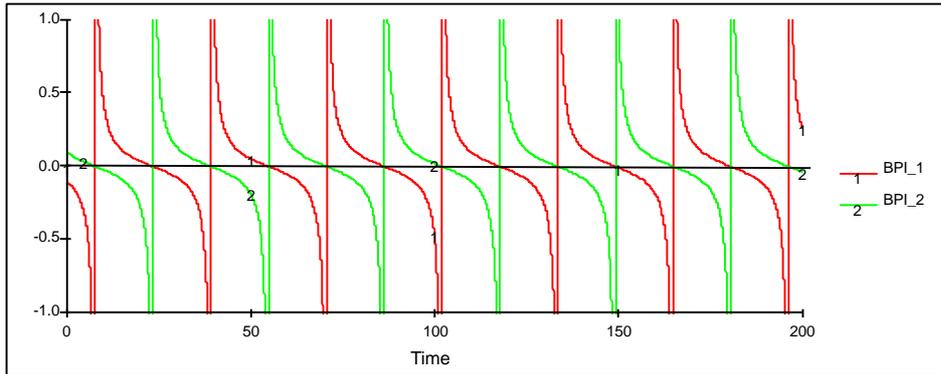
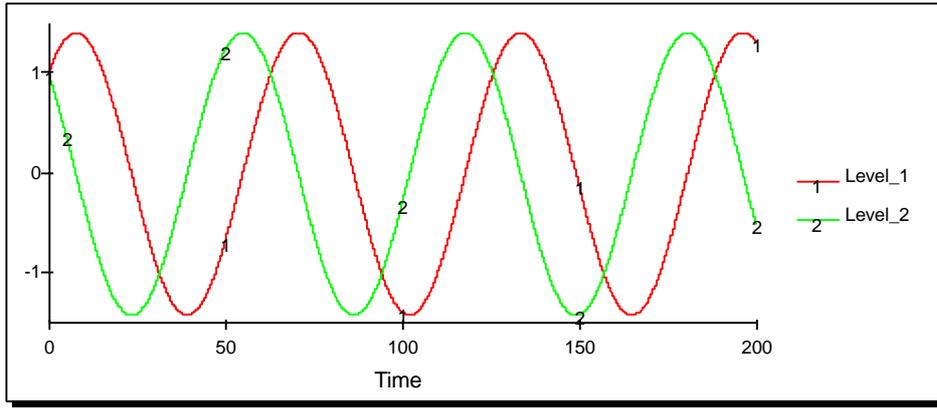
Rate_Change_ABS_Slope1 = DERIVN(ABS(Slope_1))
Rate_Change_ABS_Slope2 = DERIVN(ABS(Slope_2))

Simulation Setup Parameters:
Start Time=0; Stop Time =200; Simulation Time Step = 0.01.

```

**Table 3: Example 3, equations**

The behavior of this model is exhibited in figure 11 portraying the behavior of the state variables, Level\_1, and 2; the individual BPIs, BPI\_1 and BPI\_2; the overall BPI (in this case constant); and the  $|\underline{s}|$  (in this case constant, indicating a zero change in the length of the slope vector). In this model, while as each of the state variables are alternating between a converging and a diverging behavior mode, as indicated by the dynamics of the associated BPIs, the model as a whole exhibits a sustained mode of behavior (i.e. a sustained oscillation) as indicated by the overall BPI, that permanently takes the value  $90^\circ$  as well as the fact that the length of the slope vector,  $\underline{s}$ , is constant.



**Fig. 11: Example 3, behavior**

We may now provide an explanation for the constant length of the slope vector in the case of a sustained oscillations based upon a decomposition of the curvature vector,  $c$ , into two components, - one along the slope vector and one perpendicular to that vector. In this case, the projection along the slope vector amounts to 0 ( $\cos 90^\circ = 0$ ) and, therefore, the only change in the slope vector that takes place is a displacement perpendicular to that vector. This displacement is infinitely small and, consequently, does not cause a change in the length of the vector. Moreover, thus having changed direction by an infinitely small angle, a second, infinitely small displacement can take place, also that perpendicular to the new vector, - again causing no change in the length of the vector. This series of infinitely small adjustments of the direction of the vector, with no change in the length of the vector, causes the slope vector to move along a trajectory that constitutes an arc of a circle with the radius equal to the length of the slope vector.

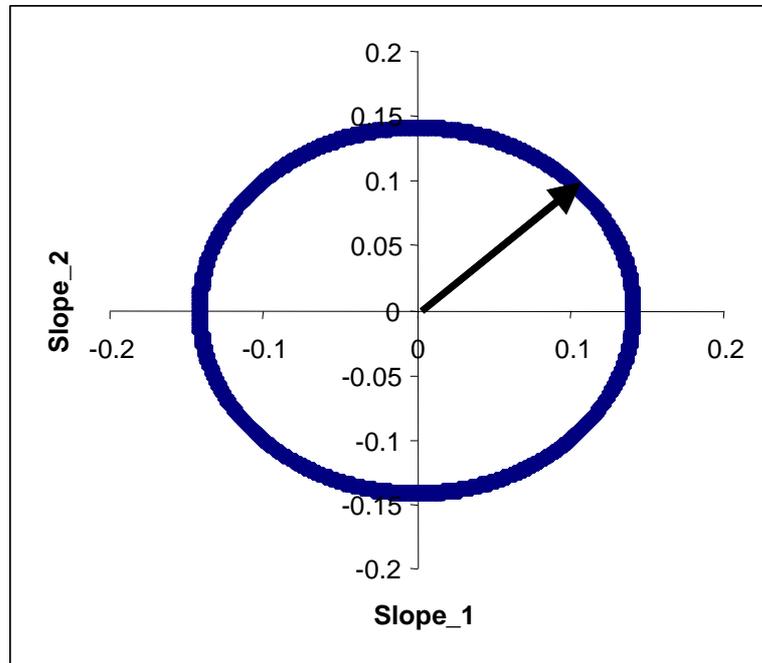
This can be seen from figure 12, where the slope vector is portrayed in a space spanned by the slopes of the individual state variables of the model. The slope vector rotates clockwise along the circle centered at origo. In figure 13, we illustrate the trajectory of the curvature vector that constitutes the rate of change of the slope vector. For illustration purpose, we consider the slope vector and the corresponding curvature vector at a particular point in time. As seen from those figures, the curvature vector is perpendicular to the slope vector at that particular point of time. As a consequence, the slope vector is displaced clockwise along the tangent of the circle portrayed in figure 12, with a tangent velocity that is equal to  $|c| \sin 90^\circ$ , i.e.  $|c|$  (the length of the curvature vector). Moreover, the curvature vector, remaining perpendicular to the slope vector, traverses clockwise the circle in figure 13. Over time, a series of infinitely small displacements, perpendicular to the slope vector, take place so as to move the slope vector along the circle portrayed in figure 12, and, thus, the curvature vector along the circle portrayed in figure 13. As the tangent velocity of the slope vector equals  $|c|$ , then the angular velocity  $\omega_{\text{slope}}$  (radians per time unit) of the slope vector will equal to  $|c|/|s|$ . As, in this case, both  $|s|$  and  $|c|$  are constants, then  $\omega_{\text{slope}}$  will in turn be constant. In example 3, the calculation of  $\omega_{\text{slope}}$  goes as follows:

in figure 12 as  $s=[0.1,0.1]$ , then  $|s| = 0.1 * \sqrt{2}$  ; and

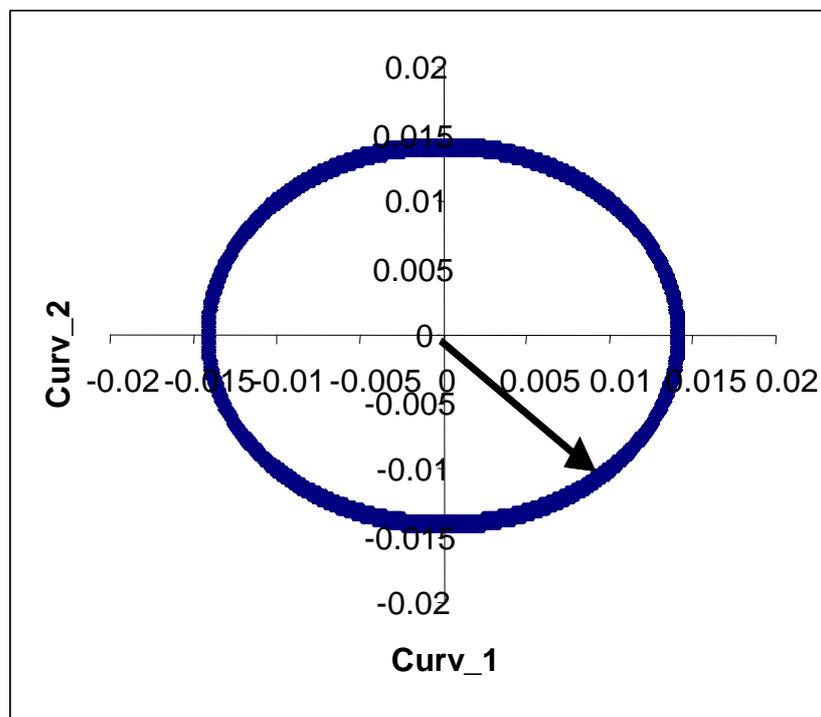
in figure 13 as  $c=[0.01,-0.01]$ , then  $|c| = 0.01 * \sqrt{2}$  ; thus

$$\omega_{\text{slope}} = |\underline{c}|/|\underline{s}| = (0.01 * \sqrt{2}) / (0.1 * \sqrt{2}) = 0.1$$

If we denote T to be the duration of one rotation around the origo (the period of one cycle); then,  $T = 2\pi / \omega_{\text{slope}} = 20\pi$ ; which is approximately 63 time units.



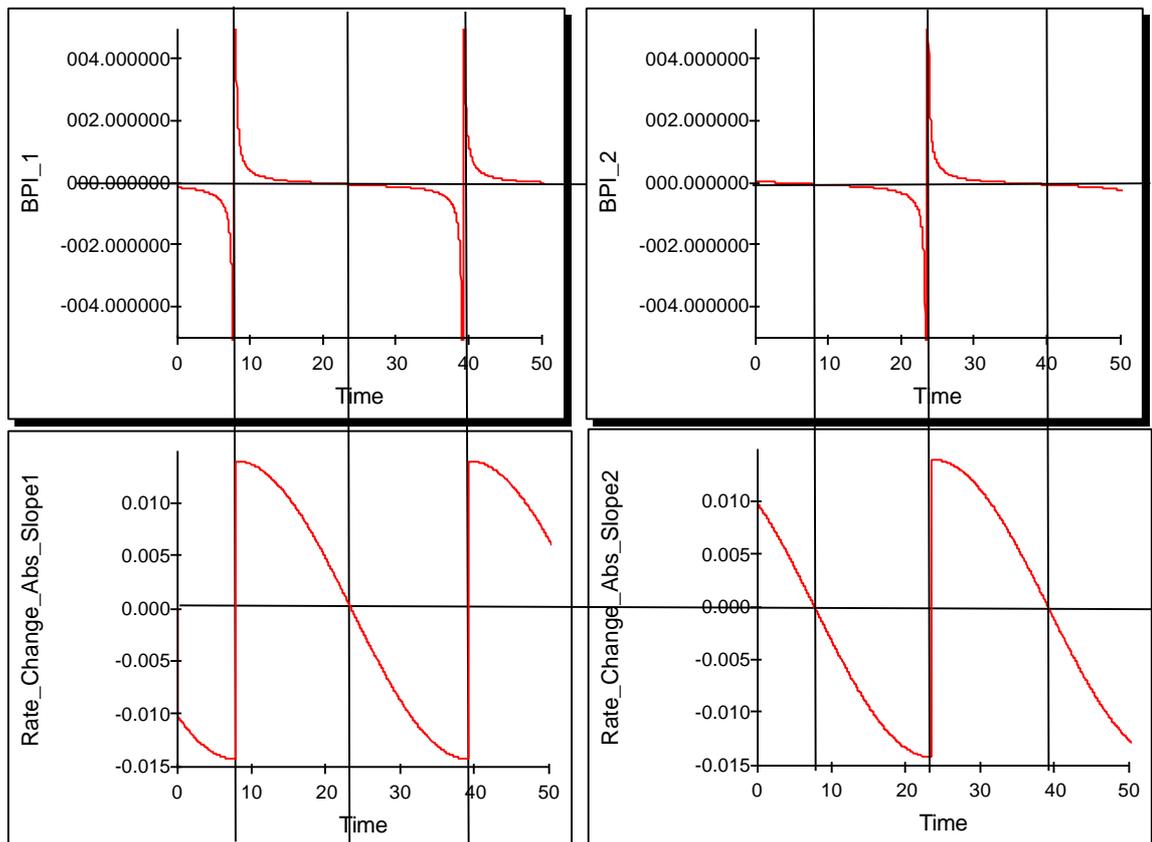
**Fig. 12: Example 3, the trajectory of the slope vector in the phase space**



**Fig. 13: Example 3, the trajectory of the curvature vector in the phase space**

Also here we may graph  $BPI_1$  and  $BPI_2$ ; as well as  $d|s_{Level_1}|/dt$  and  $d|s_{Level_2}|/dt$  to demonstrate that they, qualitatively, provide the same information about the behavior mode of each state variable, -- the fact that they alternate between a divergent and a convergent mode of behavior. This characterization of the individual behavior modes as alternating (between divergent and convergent behavior) is not in conflict with, but rather consistent with the characterization of the model behavior mode, as a sustained mode (of oscillation).

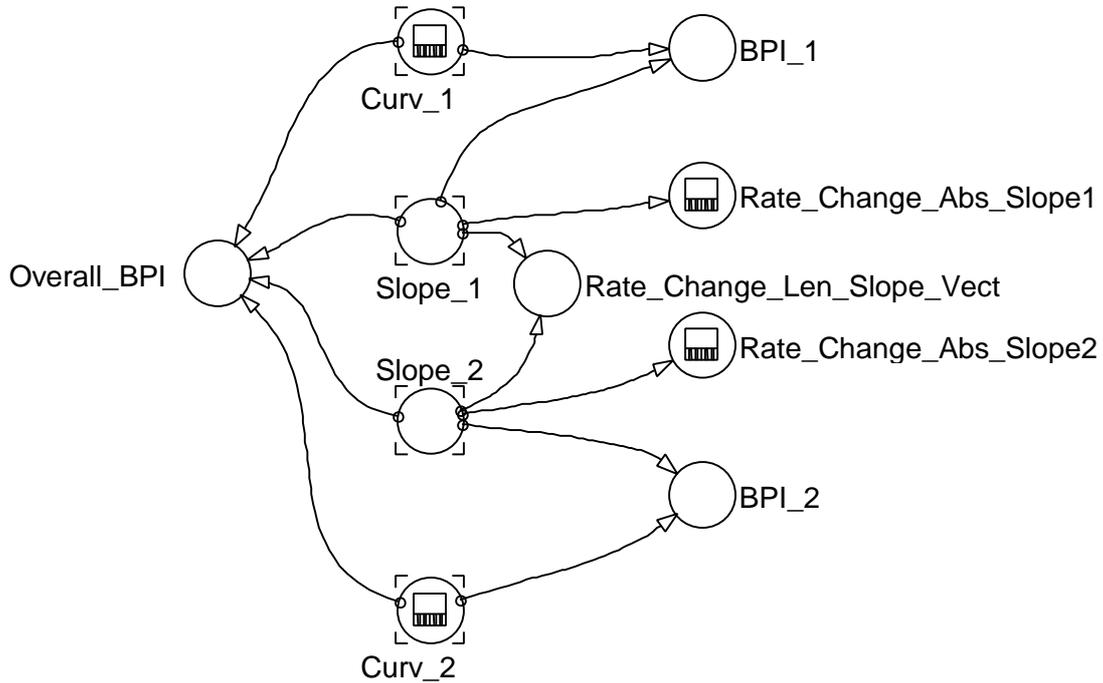
Note from the figure below that, in this example,  $BPI_x$  and  $d|s_x|/dt$  will always have the same sign, so that they qualitatively provide the same information about the modes of behavior of the state variables of the model.



**Fig. 14: Example 3, the relationships between the  $BPI_x$  and the rates of change of the absolute values of the slopes for the state variables.**

## 2.4 Behavior pattern indexes –back to example 1

The discussion of the relationship between the individual BPIs, associated with state variables, and the overall BPI, can be generalized to models exhibiting non-sustainable behavior. For that purpose, consider the behavior of the original model (Example 1, fig. 1). As indicated in figure 15 and table 4, we first add an auxiliary structure to the model; in order to calculate the individual and the overall BPIs; and also to calculate the rate of change of the length of the slope vector and its components:



**Fig. 15: Example 1, stock and flow diagram of the calculation of BPI\_1 and \_2 (BPI<sub>Level\_1 and \_2</sub>), the Overall BPI represented by the angle, and the Rate of change in the length of the slope vector and its components.**

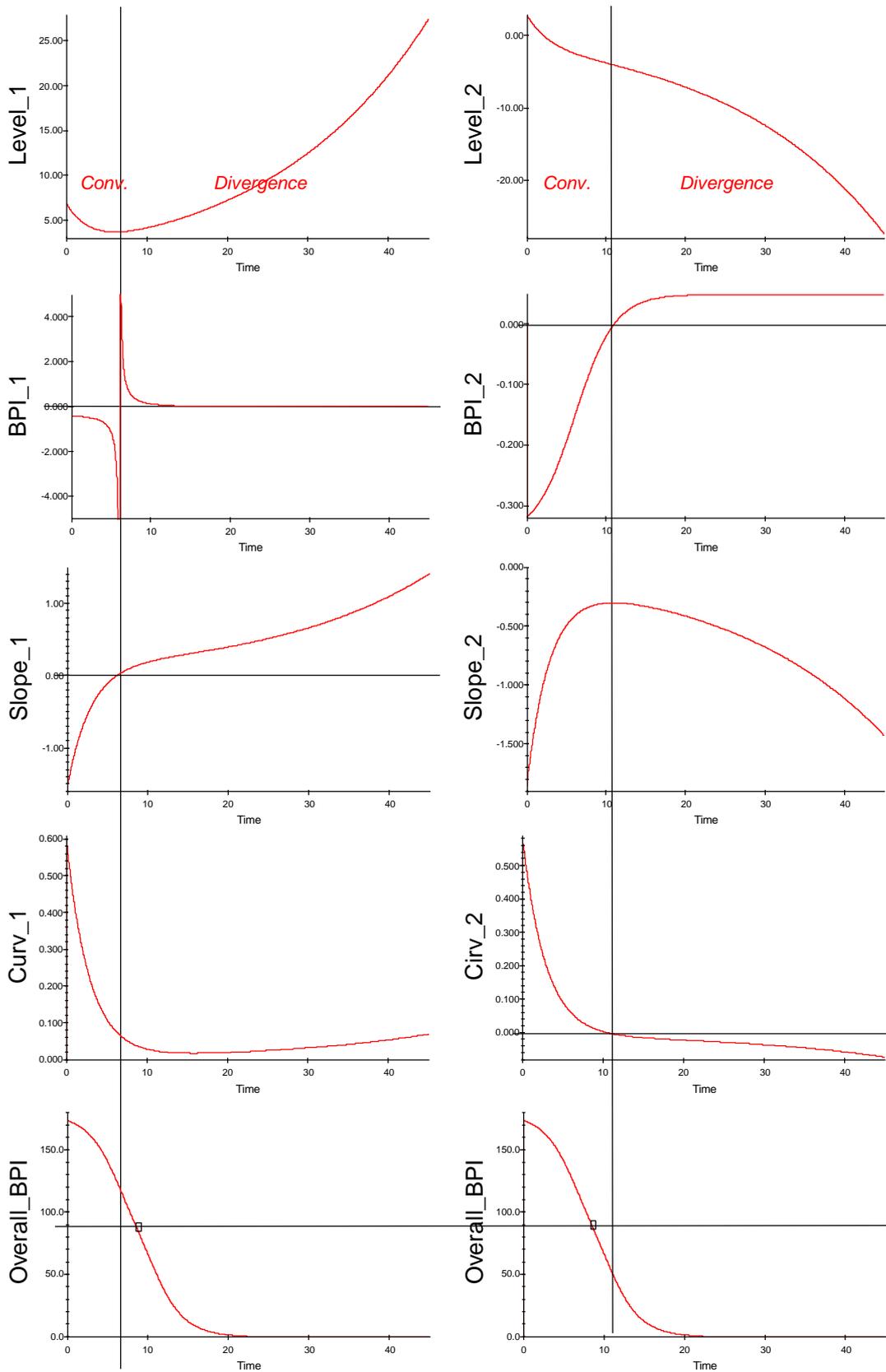
$$\begin{aligned} \text{BPI}_1 &= \text{Curv}_1 / \text{Slope}_1 \\ \text{BPI}_2 &= \text{Curv}_2 / \text{Slope}_2 \end{aligned}$$

$$\text{Overall\_BPI} = (180/\text{PI}) * \text{ARCCOS} ( ( (\text{Slope}_1 * \text{Curv}_1) + (\text{Slope}_2 * \text{Curv}_2) ) / (\text{SQRT}(\text{Slope}_1^2 + \text{Slope}_2^2) * \text{SQRT}(\text{Curv}_1^2 + \text{Curv}_2^2) ) )$$

$$\text{Rate\_Change\_Len\_Slope\_Vect} = \text{DERIVN} ( \text{SQRT} ( \text{Slope}_1^2 + \text{Slope}_2^2 ) )$$

$$\begin{aligned} \text{Rate\_Change\_Abs\_Slope1} &= \text{DERIVN}(\text{ABS}(\text{Slope}_1)) \\ \text{Rate\_Change\_Abs\_Slope2} &= \text{DERIVN}(\text{ABS}(\text{Slope}_2)) \end{aligned}$$

**Table. 4: Example 1, equations for BPI\_1 and \_2, the Overall BPI, and the rate of change of the length of the slope vector and its components.**



**Fig. 16: Example 1, Behavior**

As seen in figure 16, we may consider one state variable at a time. Beginning with Level\_1, we observe that, in the first phase, the slope,  $s$ , is negative and approaching 0, while the curvature,  $c$ , is positive, and diminishing, yet not reaching 0. The implication is that that  $BPI_1 = c/s$  approaches  $-\infty$ . Thereafter, the slope,  $s$ , is positive and increases, while the curvature,  $c$ , continues to approach zero (yet it never reaches zero). Then the curvature increases slowly. Between these two modes of behavior, a transition takes place at time 6.07, whereby  $BPI_1$  changes from  $-\infty$  to  $+\infty$ , i.e., and thus, more significantly, from a negative to a positive value.

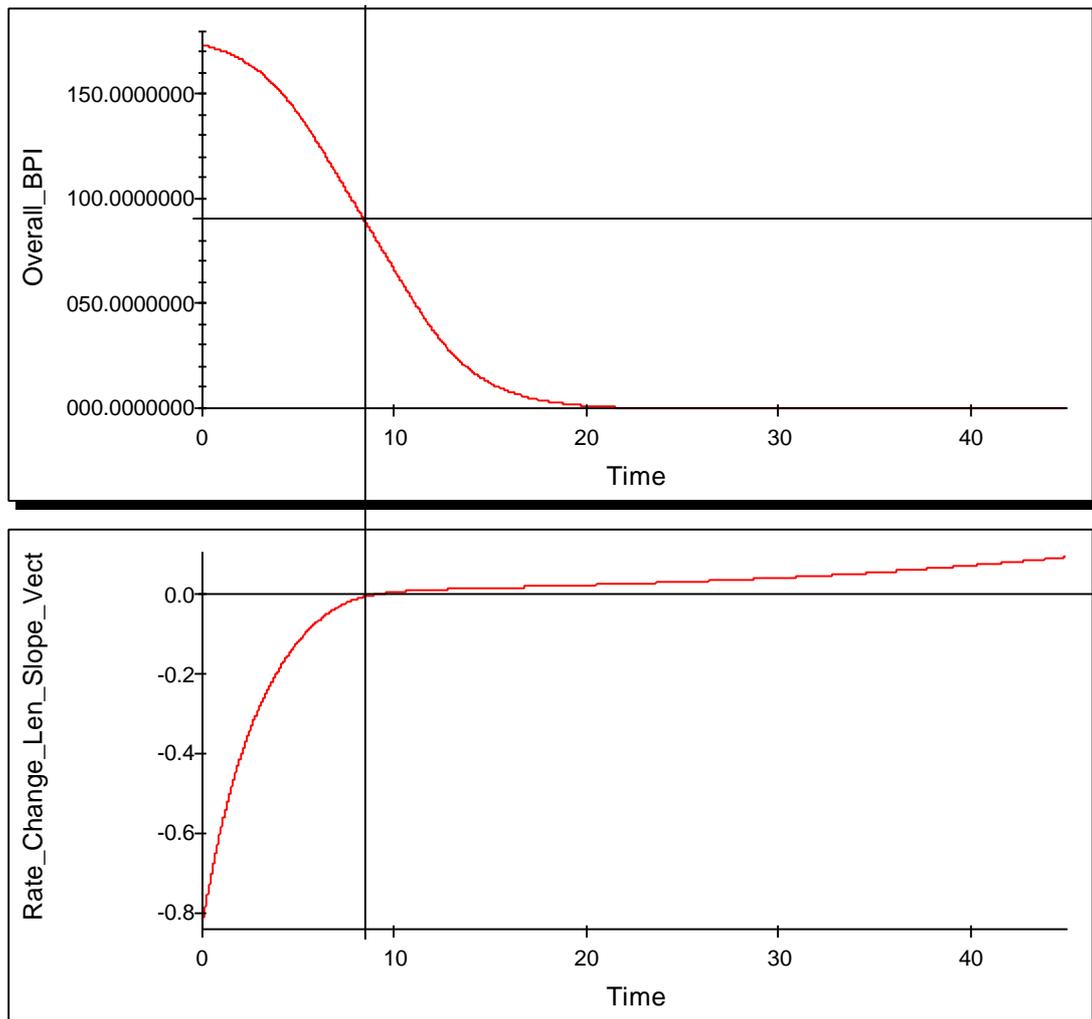
Now, considering the other state variable, Level\_2, we observe that, in the first phase, the slope,  $s$ , is negative and increasing yet not reaching 0, while the curvature,  $c$ , is positive, and diminishing, while approaching 0. The implication is that that  $BPI_2 = c/s$  approaches 0. Thereafter, the slope decreases (becomes more negative), and the curvature also decreases (becomes more negative). Between these two modes of behavior, a transition takes place at time 10.93, whereby  $BPI_2$  changes, also in this case, from a negative  $(-\epsilon)$  to a positive value  $(+\epsilon)$ .

For *the model as a whole*, the overall BPI (angle) starts out with a value around  $180^0$  and slowly approaches  $0^0$ . At the point of transition for the model as a whole, the overall BPI takes on the value  $90^0$ . This happens at time 8.5, i.e. between the points of transition for the individual state variables of the model; or, more specifically, -- midway between the two state transitions (i.e.  $8.5=6.07 + (10.93-6.07)/2$ ). Thus we can define a period of transition of behavior mode dominance for the model as a whole, initiated by the time the first state variable transition takes place and ending at the time of the last state variable transition. For the model as a whole, therefore, we define the transition of mode dominance to take place when the overall  $BPI = 90^0$ .

**Note one very important fact: The model described above is a linear model and yet it exhibits shifts between modes of behavior while in a transient development.**

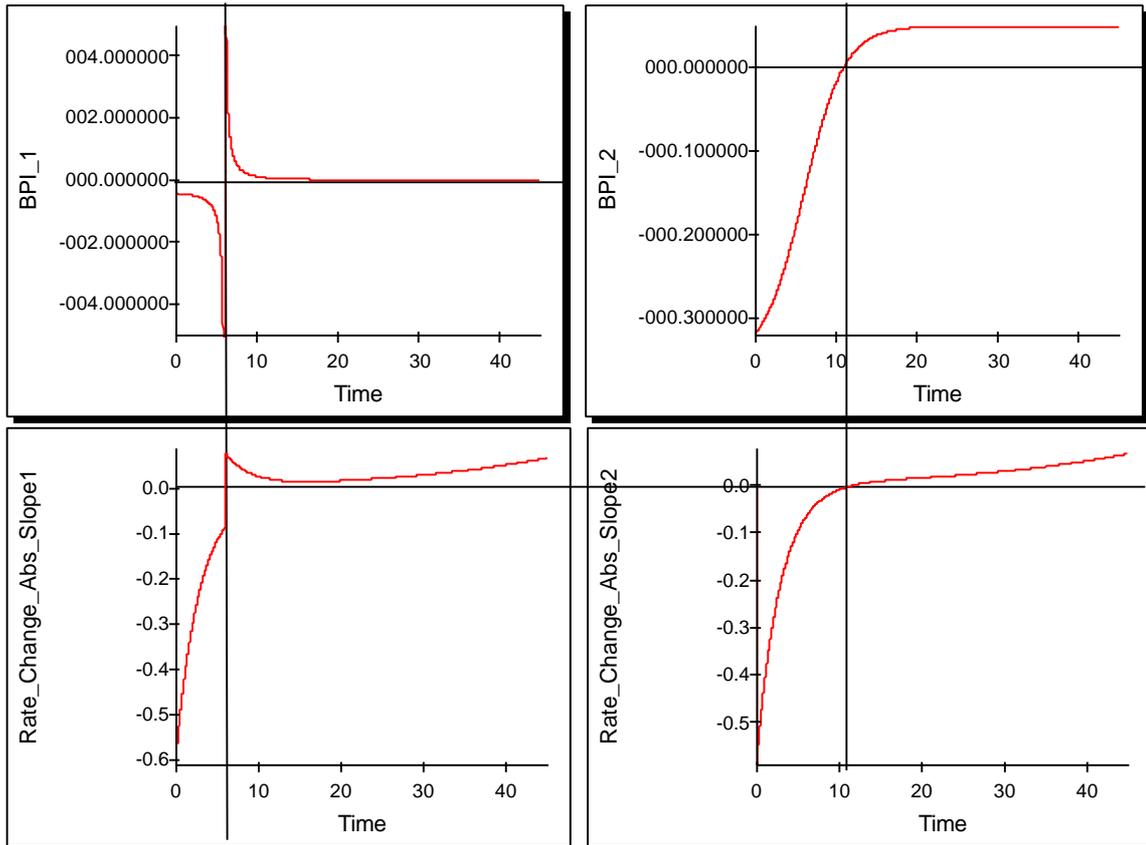
In figure 17, we portray the behavior of the model as a whole, expressed in terms of the overall BPI, and we relate that to the rate of change of the length of the slope vector. In the first phase, the overall BPI is larger than  $90^0$ , and the rate of change in

the length of the slope vector is, consequently, negative. This implies a contraction of the slope vector and indicates a convergent behavior for the model as a whole. The contraction is gradually fading and reaches 0 at time = 8.5. At that time, the overall BPI takes the value  $90^\circ$ , and there is neither a contraction, nor an expansion in the slope vector. Subsequently, in the second phase, the overall BPI is smaller than  $90^\circ$ , and the rate of change of the length of the slope vector is, consequently, positive. This implies an expansion of the slope vector and indicates a divergent behavior for the model as a whole.



**Fig. 17: Example 1, the relationship between the overall BPI and the rate of change of the length of the slope vector.**

Note from figure 18 that  $BPI_x$  and  $d|s_x|/dt$  will always have the same sign, so that they qualitatively provide the same information about the modes of behavior of the state variables of the model.



**Fig. 18: Example 1, the relationships between the BPIx and the rates of change of the absolute values of the slopes for the state variables.**

To summarize our analysis of this example (example 1) so far, observe that the transition of the mode of behavior *of the model as a whole* takes place in the midst of a series of single mode behavior transitions, each associated with a particular state variable. At each such individual change in mode of behavior, a particular component of the slope vector changes from a contraction to an expansion in length (or, as in other cases, the reverse). The total impact of such individual component transitions on the length of the slope vector, determines the mode of behavior of the model as a whole. In this case, there is a transition from contraction to expansion in the first slope vector component, i.e. Slope\_1 at time 6.07. Thereafter, however, the second component, i.e. Slope\_2, is still contracting and that contraction is sufficiently significant to continue the contraction in the length of *the slope vector as a whole*. Over the next time period, until time = 8.5, the contraction continues, in spite of the expansion of the first slope vector component. Eventually, however, at time = 8.5, this expansion begins to dominate the mode of behavior of *the model as a whole*, in the sense that the contraction in the second vector component, i.e. Slope\_2, is no longer

sufficient to compensate for the expansion in the first component, i.e. Slope\_1. As a consequence, we experience a transition in the mode of behavior for the model as a whole from convergence to divergence, i.e. the contraction of the slope vector halts and an expansion sets in. Over the next time period, until time = 10.93, the second component of the vector, i.e. Slope\_2, still contracts. But now the impact of this contraction on the length of *the vector as a whole* is less significant than the impact of the reinforced expansion, taking place in the first component, i.e. Slope\_1. Thus the divergent behavior mode of *the model as a whole* becomes gradually more apparent. At time = 10.93, we see a transition from contraction to expansion also in the second slope vector component, i.e. Slope\_2. Thereafter, both of the components of the slope vector are expanding and, thus, so also the vector itself.

The transition of the mode of behavior of a model, state variable by state variable, which we have described above, - one that eventually leads to a transition of the model of behavior of *the model as a whole*, can be generalized to a model of  $n$  state variables. Typically, we may observe a transition in the mode of behavior for single state variables long before there is a change in the mode of behavior of the model as a whole. Moreover, other single state variable mode transitions are taking place after there has been a transition in the mode of behavior of the model as a whole. Note that such transitions typically constitute significant “events” in the development of dynamic systems, events that we want to recognize, predict, prepare for, promote or postpone, cause or avoid. Our analysis can be generalized to any  $n^{\text{th}}$  order model, and indicates that we may establish an “early warning system” for such events, based on “leading indicators”, so as to be able to take appropriate actions in time.

### **3 The impact of the eigenvalues on the rate of change of the length of the slope vector and its components**

#### **3.1 The gain matrix as a foundation for the relationship between the curvature and slope vectors**

The properties of the structure of the model that we will be considering are the gains of the links that constitute the structure of the model. In any model, linear as well as non-linear, we can identify the gain matrix (equivalent to the Jacobian used in linear analysis):

$$\mathbf{G} = \begin{bmatrix} \frac{\partial \dot{x}_1}{\partial x_1} & \frac{\partial \dot{x}_1}{\partial x_2} & \cdot & \frac{\partial \dot{x}_1}{\partial x_n} \\ \frac{\partial \dot{x}_2}{\partial x_1} & & & \cdot \\ \cdot & & & \cdot \\ \frac{\partial \dot{x}_n}{\partial x_1} & \cdot & \cdot & \frac{\partial \dot{x}_n}{\partial x_n} \end{bmatrix}$$

Each element  $(\frac{\partial \dot{x}_i}{\partial x_j})$  in the above matrix constitutes a gain; i.e. the change in the net rate (slope) of each state variable in response to a change in the level (value) of any state variable in the model. Note that for any non-linear model, the value of any element in the gain matrix can be calculated numerically (using finite-difference approximations) for any state of the model (i.e. values of the state variables at any point in time).

Note that as

$$\dot{x}_i = F(x_1, x_2, \dots, x_n)$$

Then, by the chain rule:

$$\ddot{x}_i = \frac{\partial \dot{x}_i}{\partial x_1} * \dot{x}_1 + \frac{\partial \dot{x}_i}{\partial x_2} * \dot{x}_2 + \dots + \frac{\partial \dot{x}_i}{\partial x_n} * \dot{x}_n$$

We can now utilize the chain rule for all state variables with the following result:

$$\underline{\ddot{x}} = \mathbf{G} \underline{\dot{x}}$$

Thus

$$\underline{c} = \mathbf{G} \underline{s}$$

so that the gain matrix,  $G$ , relates the slope vector characterizing the behavior of the model at any point in time to the curvature vector, also characterizing that behavior. At any point in time, therefore,  $G$  transforms the  $\underline{s}$  vector into the  $\underline{c}$  vector in an  $n$ -dimensional standard space. In this space, the solution to the system of differential equations,

$$\underline{c} = G \underline{s},$$

provides us with the time trajectory of the curvature and the slope of the model. In general, however, this system of differential equations can be solved only in exceptionally simple cases because of the fact that each curvature  $\ddot{x}$  in principle is a linear combination of all the slopes  $\dot{x}$ . The implication is that the curvature  $\ddot{x}$  associated with a single coordinate (axis) in this space is determined by the slopes associated with all the coordinates ( $\dot{\underline{x}}$ ) (components mixing).

To be able to solve for the time trajectories of the curvature and the slope of the model, we consequently, change coordinate system using eigenvalue analysis. (The reader who is interested in a thorough overview of matrix eigensystem theorems may consult appendix A).

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

$$G \underline{r}_i = \lambda_i \underline{r}_i$$

If  $G$  is an  $n \times n$  matrix, we will have  $n$  eigenvalues,  $\lambda_i$ , each associated with a right eigenvector  $\underline{r}_i$ . In the normal case, we will have  $n$  distinct eigenvalues and the right eigenvectors will be linearly independent (see appendix A, Corollary A.3), and span an  $n$ -dimensional space,  $R^n$ ; i.e. the right eigenvectors will form a new coordinate system in  $R^n$  that we, in this paper, will call the “*eigen-coordinate system*”. 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 this space along which the dynamics of the model unfold as explained below.

In this space, therefore, the slopes vector,  $\underline{s}$ , can, at any time be expressed as a linear combination of the right eigenvectors:

$$\underline{s} = \alpha_1 \underline{r}_1 + \alpha_2 \underline{r}_2 + \dots + \alpha_n \underline{r}_n$$

In this new (eigen-) coordinate system the alphas ( $\alpha_i$ ) will be the *new* components of the slope vector.

By differentiating the previous equation over time we obtain

$$\underline{c} = \dot{\alpha}_1 \underline{r}_1 + \dot{\alpha}_2 \underline{r}_2 + \dots + \dot{\alpha}_n \underline{r}_n$$

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

Substituting  $\underline{s}$  into the equation,  $\underline{c} = \mathbf{G} \underline{s}$ , we obtain;

$$\underline{c} = \mathbf{G} [\alpha_1 \underline{r}_1 + \alpha_2 \underline{r}_2 + \dots + \alpha_n \underline{r}_n]$$

by rearranging, we obtain;

$$\underline{c} = \alpha_1 \mathbf{G} \underline{r}_1 + \alpha_2 \mathbf{G} \underline{r}_2 + \dots + \alpha_n \mathbf{G} \underline{r}_n$$

and utilizing that  $\mathbf{G} \underline{r}_i = \lambda_i \underline{r}_i$ , we obtain;

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

Recalling that

$$\underline{c} = \dot{\alpha}_1 \underline{r}_1 + \dot{\alpha}_2 \underline{r}_2 + \dots + \dot{\alpha}_n \underline{r}_n$$

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

$$\dot{\alpha}_i = \lambda_i \alpha_i$$

Hence we obtain the solutions

$$\alpha_i = \alpha_i^0 e^{\lambda_i(t-\tau)},$$

where  $\tau$  is the initial time.  $\alpha_i^0$  are the initial values of  $\alpha_i$  (at time  $\tau$ ).

It is clear that the only factor determining the dynamics along a particular coordinate (i.e. right eigenvectors) is the eigenvalue associated with that coordinate itself.

Substituting these solutions into the equation for the slope vector yields the time trajectory of the slope:

$$\underline{\mathbf{s}} = \alpha_1^0 e^{\lambda_1(t-\tau)} \underline{\mathbf{r}}_1 + \alpha_2^0 e^{\lambda_2(t-\tau)} \underline{\mathbf{r}}_2 + \dots + \alpha_n^0 e^{\lambda_n(t-\tau)} \underline{\mathbf{r}}_n$$

Now, in a linear model, the relationship

$$\underline{\mathbf{c}} = \mathbf{G} \underline{\mathbf{s}}$$

holds over time and so does the expression for the time trajectory of the slope vector,  $\underline{\mathbf{s}}$ . For a non-linear model this relationship is instantaneous. We will presuppose that the relationship,  $\mathbf{G}$ , holds for a small period of time, corresponding to the original simulation interval. Thus the expression for the time trajectory of  $\underline{\mathbf{s}}$  will also hold for this small period of time. For the purpose of our numerical analysis, we will now choose a new simulation interval that constitutes a fraction of the original interval. Partitioning the time horizon, will allow us to apply the analysis described above to non-linear models over each of the resulting time intervals.

We have now described the relationship between the eigenvalues  $\lambda_i$  and the behavior of the model, characterized by the slopes and the curvatures that are exhibited by the state variables of the model. More specifically, we have demonstrated that the dynamics along each of the coordinates (i.e. the right eigenvectors) is determined by a single eigenvalue.

Note that the eigenvalues originate from the *loops* of the structure of the model (as explained in section 4.1). Thus, by studying their relative impact on the rate of change of the length of the slope vector, we will be able to identify the impact of the structural components, i.e. loops, of the model on the model behavior. Moreover, by studying the impact of the eigenvalues on the rate of change of the length of a particular slope vector component that is associated with a single state variable, we will be able to identify the impact of the structural components in the model, on the behavior of that state variable.

In the rest of this section, we outline our method of investigating the relative impact of a particular eigenvalue on the model behavior as a whole. We also apply this method to our main example, example 1.

### 3.2 A method for ranking eigenvalues according to their relative impact on the dynamics of the slope vector or its components

In this method, we first start out with a base run, whereby we allow for all of the eigenvalues to simultaneously impact the behavior of the model as described by the equation for the time trajectory of  $\underline{s}$ . Subsequently, we eliminate, one at a time, the dynamics of the model along one particular eigenvector,  $\underline{r}_i$ , caused by the associated eigenvalue,  $\lambda_i$ . I.e. in the equations for  $\underline{s}$ , we multiply a particular exponent  $\lambda_i(t-\tau)$  by 0, e.g. for  $i=1$ ;

$$\underline{s} = \alpha_1^0 e^{\lambda_1(t-\tau)*0} \underline{r}_1 + \alpha_2^0 e^{\lambda_2(t-\tau)} \underline{r}_2 + \dots + \alpha_n^0 e^{\lambda_n(t-\tau)} \underline{r}_n$$

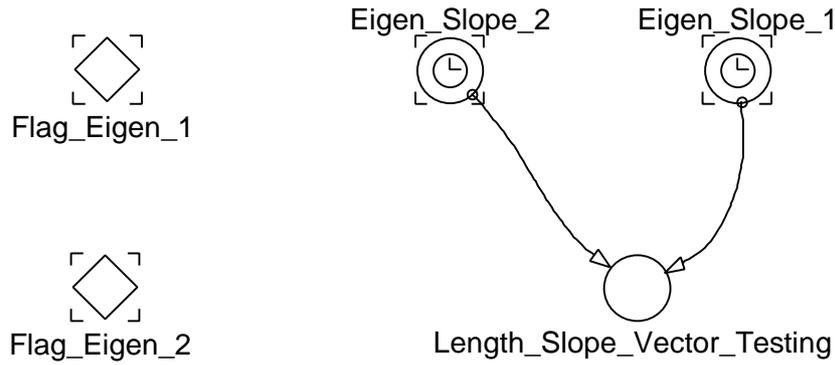
$$\text{i.e. } \underline{s} = \alpha_1^0 \underline{r}_1 + \alpha_2^0 e^{\lambda_2(t-\tau)} \underline{r}_2 + \dots + \alpha_n^0 e^{\lambda_n(t-\tau)} \underline{r}_n$$

The result is that there is no more dynamics along  $\underline{r}_1$ . That is, the value taken along this axis (the contribution to the total behavior) is constant,  $\alpha_1^0$ , i.e. the initial value taken by  $\alpha_1$  at the start of the current interval.

By repeating this process for each  $i = 1 \dots n$ , we can rank the eigenvalues according to their relative impact on the rate of change of the length of the slope vector at any point (interval) in time. Since each of these eigenvalues characterizes the dynamics along a specific eigenvector, we implicitly rank the impact of each such dynamic component on the total behavior of the model.

We now return to Example 1 (fig. 1) to illustrate the relationship between the eigenvalues and the rate of change of the length of the slope vector. We first describe an additional section in the model (fig. 19 and table 5); one that describes the slopes, Slope\_1 and \_2 in terms of the eigenvalues, thus linking the eigenvalues to the behavior of the model.

Note that the flags introduced in the equations for Slope\_1 and \_2 are used to (de-) activate the impact of the eigenvalues on the behavior of the model.



**Fig. 19: Example 1, stock and flow diagram, linking behavior to eigenvalues.**

$$\begin{aligned} \text{Eigen\_Slope\_1} = & \\ & (\text{InitAlpha1} * r1(1) * \text{EXP}(\text{Eigenvalue1} * (\text{TIME} - \text{STARTTIME}) * \text{Flag\_Eigen\_1})) \\ & + (\text{InitAlpha2} * r2(1) * \text{EXP}(\text{Eigenvalue2} * (\text{TIME} - \text{STARTTIME}) * \text{Flag\_Eigen\_2})) \end{aligned}$$

$$\begin{aligned} \text{Eigen\_Slope\_2} = & \\ & (\text{InitAlpha1} * r1(2) * \text{EXP}(\text{Eigenvalue1} * (\text{TIME} - \text{STARTTIME}) * \text{Flag\_Eigen\_1})) \\ & + (\text{InitAlpha2} * r2(2) * \text{EXP}(\text{Eigenvalue2} * (\text{TIME} - \text{STARTTIME}) * \text{Flag\_Eigen\_2})) \end{aligned}$$

$$\text{Length\_Slope\_Vector\_Testing} = \text{SQRT}(\text{Eigen\_Slope\_1}^2 + \text{Eigen\_Slope\_2}^2)$$

$$\begin{aligned} \text{Flag\_Eigen\_1} &= 1 && (\text{set to 1 or 0}) \\ \text{Flag\_Eigen\_2} &= 1 && (\text{set to 1 or 0}) \end{aligned}$$

Where:

*Eigenvalue1 is the first eigenvalue.*

*Eigenvalue2 is the second eigenvalue.*

*r1 is first right eigenvector associated with Eigenvalue1; r1(i) is the i<sup>th</sup> element in the vector. r2 is second right eigenvector associated with Eigenvalue2; r2(i) is the i<sup>th</sup> element in the vector.*

*Note that the eigenvalues and the eigenvectors are calculated from the gain matrix G.*

*InitAlpha1, and InitAlpha2 are the initial values of the as. Their values are computed from the following simultaneous equations:*

$$\text{Initial value of Slope\_1} = \text{InitAlpha1} * r1(1) + \text{InitAlpha2} * r2(1)$$

$$\text{Initial value of Slope\_2} = \text{InitAlpha1} * r1(2) + \text{InitAlpha2} * r2(2)$$

*Note that those simultaneous equations result from substituting TIME by STARTTIME in the equations of the slopes.*

**Simulation Setup Parameters:**

**Start Time=0; Stop Time =45; Simulation Time Step = 0.005; Analysis Time Step = 0.1 (original simulation time step).**

**Table. 5: Example 1, equations, linking behavior to eigenvalues.**

We will now perform two experiments, one in the convergent phase using the analysis time step (0,0.1); and the other experiment in the divergent phase using the analysis time step (20,20.1). Each experiment will take as a point of departure its unique initial values for the  $\alpha$ s, which are computed from the eigenvectors (that are computed from

the gain matrix,  $\mathbf{G}$ ) and the initial values of the slopes at the start of the analysis time step. In the first experiment, the initial values of slopes are the values of the slopes in the original model at time = 0. In the second experiment, the initial values of the slopes are the values of the slopes in the original model at time = 20.

- First Experiment:

The parameters governing the simulation are:

Start Time=0; Stop Time =0.1; Simulation Time Step = 0.005; Analysis Time Step = 0.1; Time for Setting Flags= Start Time=0.

The Gain Matrix is computed from table 1 as follows:

$$\mathbf{G} = \begin{bmatrix} -0.15 & -0.2 \\ -0.2 & -0.15 \end{bmatrix}$$

From  $\mathbf{G}$ , one can use a “matrix eigensystem” algorithm to compute the eigenvalues and the eigenvectors. The interested reader may consult the “EISPACK Guide” (Goos & Hartmanis, 1976).

$$\text{Eigenvalue1}=0.05; \quad \text{Eigenvalue2}=-0.35$$

$$r1 = \begin{bmatrix} -1/\sqrt{2} \\ 1/\sqrt{2} \end{bmatrix}; \quad r2 = \begin{bmatrix} 1/\sqrt{2} \\ 1/\sqrt{2} \end{bmatrix}$$

The Initial Values of the slopes (at time = 0) are:

$$\text{Slope}_1 = -1.55; \quad \text{Slope}_2 = -1.85$$

Hence, the Values of InitAlpha1 and InitAlpha2 are:

$$\text{InitAlpha1} = -0.212; \quad \text{InitAlpha2} = -2.404$$

Now, we are carry out three simulation runs, each with a different setting of the flags, and we report on the dynamics of the length of the slope vector for each run in the following table.

$$\text{Base Run: Flag\_Eigen\_1} = 1, \text{Flag\_Eigen\_2} = 1$$

$$\text{1st Case: Flag\_Eigen\_1} = 0, \text{Flag\_Eigen\_2} = 1 \\ (\text{Stop the dynamics associated with the first eigenvalue})$$

2nd Case: Flag\_Eigen\_1 = 1, Flag\_Eigen\_2 = 0  
 (stop the dynamics associated with the second eigenvalue)

Time	$ \underline{s} $ in Base Run	$ \underline{s} $ in 1 <sup>st</sup> case	$ \underline{s} $ in 2 <sup>nd</sup> Case
0.0	2.41	2.41	2.41
0.1	2.33	2.33	2.41

**Table. 6: Example 1, results of the first experiment, the impact of the eigenvalues on the Dynamics of the length of the slope vector.**

It is clear from these results that eliminating the effect of the *first* eigenvalue (in this time interval) does not cause a change in the rate of change of the length of the slope vector. On the other hand, eliminating the effect of the *second* eigenvalue (in this time interval) does cause a significant change in the rate of change of the length of the slope vector. In fact, the rate of change of the length of the slope vector becomes 0. From this we can conclude that that the *second* eigenvalue *dominates* the model behavior in this time interval.

- Second Experiment:

The parameters governing the simulation are:  
 Start Time=20; Stop Time =20.1; Simulation Time Step = 0.005; Analysis Time Step = 0.1; Time for Setting Flags= Start Time=20

The Gain Matrix is:

$$\mathbf{G} = \begin{bmatrix} -0.15 & -0.2 \\ -0.2 & -0.15 \end{bmatrix}$$

Note that the gain matrix in the second experiment is the same as the gain matrix in the first experiment. Thus we have the same values for the eigenvalues and the eigenvectors. Generally, in non-linear models, the gain matrix will change with time, yet this will have no effect on our method of analysis (as we mentioned before).

$$\text{Eigenvalue1}=0.05; \quad \text{Eigenvalue2}=-0.35$$

$$\mathbf{r1} = \begin{bmatrix} -1/\sqrt{2} \\ 1/\sqrt{2} \end{bmatrix}; \quad \mathbf{r2} = \begin{bmatrix} 1/\sqrt{2} \\ 1/\sqrt{2} \end{bmatrix}$$

The Initial Values of the rates (at time = 20) are:

$$\text{Slope}_1 = 0.406; \quad \text{Slope}_2 = -0.409$$

Hence, the Values of InitAlpha1 and InitAlpha2 are:

$$\text{InitAlpha1} = -0.577; \text{InitAlpha2} = -0.00218$$

Once again, we carry out three simulation runs, each with a different setting of the flags, and we report on the dynamics of the length of the slope vector for each run in the following table.

Base Run: Flag\_Eigen\_1 = 1, Flag\_Eigen\_2 = 1

1st Case: Flag\_Eigen\_1 = 0, Flag\_Eigen\_2 = 1  
(Stop the dynamics associated with the first eigenvalue)

2nd Case: Flag\_Eigen\_1 = 1, Flag\_Eigen\_2 = 0  
(stop the dynamics associated with the second eigenvalue)

Time	$ \underline{s} $ in Base Run	$ \underline{s} $ in 1 <sup>st</sup> case	$ \underline{s} $ in 2 <sup>nd</sup> Case
20.0	0.577	0.577	0.577
20.1	0.579	0.577	0.579

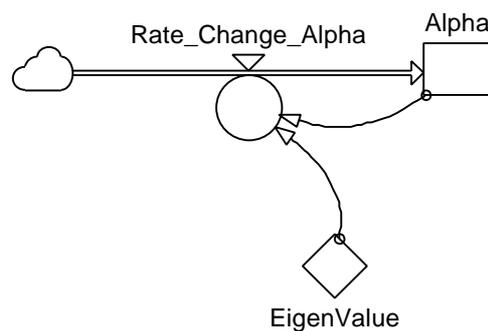
**Table. 7: Example 1, results of the second experiment, the impact of the eigenvalues on the Dynamics of the length of the slope vector.**

It is clear from these results that eliminating the effect of the *second* eigenvalue (in this time interval) does not cause a change in the rate of change of the length of the slope vector. On the other hand, eliminating the effect of the *first* eigenvalue (in this time interval) does cause a significant change in the rate of change of the length of the slope vector. In fact, the rate of change of the length of the slope vector becomes 0. From this we can conclude that that the *first* eigenvalue *dominates* the model behavior in this time interval.

### 3.3 Remarks on the method of ranking eigenvalues

#### 3.3.1 The eigen-loop

We want to comment on an important observation that is revealed by the experiments reported in the previous section. Note that the gain matrix,  $\mathbf{G}$ , was constant over time (as this is a linear model) and hence the eigenvalues and eigenvectors were also constants; yet eigenvalue2 was dominating at the analysis time step (0,0.1), while eigenvalue1 was dominating at the analysis time step (20,20.1). So, what is the reason behind this shift in dominance for the eigenvalues in this linear model? From the above experiments, one can see that the only attributes that changed from the analysis time step (0,0.1) to the analysis time step (20,20.1) were the initial values of the  $\alpha$ s (InitAlpha1 & InitAlpha2). Hence one can conclude that the changes in the values of the initial values of the  $\alpha$ s, caused this shift in eigenvalue dominance. So, what do the initial values of the  $\alpha$ s represent after all? In the eigen-coordinate system, the initial values of the  $\alpha$ s represent the components of the slope vector (along each of the right-eigenvectors) at the start time of the analysis period. So what make the values of  $\alpha$ s evolve (change) with time? Recall that the dynamics of each  $\alpha$  is governed by the first order differential equation:  $\dot{\alpha}_i = \lambda_i \alpha_i$ . In SD terminology, this differential equation constitutes a first order loop like the one shown in the figure below.



**Fig. 20: The Eigen-Loop**

Thus, from the eigen-coordinate system perspective, a linear model -- or a non-linear one, whose gain matrix is constant over a short period of time (by numerical approximations) -- can be visualized as a collection of “ $n$ ” disjoint simple feedback

loops (such as the one in the figure above). Each loop has its unique dynamic characteristic that is solely determined by its eigenvalue.

### 3.3.2 Complex eigenvalues

Finally, in this section we turn our attention to the fact that an eigenvalue can be a complex number; yet as we are going to explain below, the equation of the time trajectory of the slope vector (or any of its components) will contain only real values.

In fig. 9, we portray an example of a model (example 3) that has complex eigenvalues. Recall that this model has a sustained oscillation behavior.

From table 3, we can calculate the gain matrix.

$$\mathbf{G} = \begin{bmatrix} 0 & 0.1 \\ -0.1 & 0 \end{bmatrix}$$

From the gain matrix, we can compute the eigenvalues.

$$l_1 = 0 + 0.1i; l_2 = 0 - 0.1i$$

Notice that  $l_2$  is the complex conjugate of  $l_1$ . In general if  $l_i$  is a complex number, then  $l_{i+1}$  is its complex conjugate. As we are going to see below, the superposition of a complex conjugate pair of eigenvalues in the time equation of the slope vector (or any of its components) yields a real sinusoidal wave (with no complex values).

In our method to rank eigenvalues (according to their impact on the model behavior) we will treat complex eigenvalues slightly different than real ones. Recall that the method to determine the impact of an eigenvalue on the model behavior is to eliminate the dynamics associated with this eigenvalue. In the case where we have a complex conjugate pair of eigenvalues, we will eliminate the dynamics associated with these two complex conjugate eigenvalues at the same time to determine their compound effect on the behavior.<sup>3</sup>

---

<sup>3</sup> For future research, one may consider eliminating separately the dynamics associated with the real part of the eigenvalues, and the dynamics associated with the imaginary part of the eigenvalues.

We will demonstrate that superposition of  $l_1$  and  $l_2$  in the time equation of Slope\_1 (like the one given in table 5) yields a real sinusoidal wave.

We will start by computing the right eigenvectors and the initial values of  $\alpha s$  (InitAlpha1 and Init Alpha2) at time =0.

The right eigenvectors are computed directly from the gain matrix, G.

$$r1 = \begin{bmatrix} 1/\sqrt{2} \\ 0 + 1/\sqrt{2}i \end{bmatrix} \quad \& \quad r2 = \begin{bmatrix} 1/\sqrt{2} \\ 0 - 1/\sqrt{2}i \end{bmatrix}$$

Notice that each element in r1 has its complex conjugate counterpart in r2.

InitAlpha1 and Init Alpha2 are computed from the initial values of rates and the right eigenvectors (as we mentioned before in table 5).

$$\begin{aligned} &\text{At time} = 0 \\ \text{Slope}_1 &= 0.1 \quad \& \quad \text{Slope}_2 = -0.1 \end{aligned}$$

Thus

$$\text{Init Alpha1} = 0.1 * \left( \frac{1}{\sqrt{2}} + \frac{1}{\sqrt{2}}i \right) \quad \& \quad \text{Init Alpha2} = 0.1 * \left( \frac{1}{\sqrt{2}} - \frac{1}{\sqrt{2}}i \right)$$

The time equation of Slope\_1 (in table 5) is as follows:

$$\text{Slope}_1 = (\text{InitAlpha1} * r1(1) * \text{EXP}(l_1 * \text{TIME})) + (\text{InitAlpha2} * r2(1) * \text{EXP}(l_2 * \text{TIME}))$$

Substituting the values of InitAlpha1, InitAlpha2, r1(1), r2(1), Eigenvalue1 and Eigenvalue2 yields:

$$\text{Slope}_1 = 0.1 * \left[ \left( \frac{1}{\sqrt{2}} + \frac{1}{\sqrt{2}}i \right) * \frac{1}{\sqrt{2}} * \text{Exp}(0.1i * t) + \left( \frac{1}{\sqrt{2}} - \frac{1}{\sqrt{2}}i \right) * \frac{1}{\sqrt{2}} * \text{Exp}(-0.1i * t) \right]$$

$$\text{Slope}_1 = 0.1 * \left[ \frac{1}{2} \{ \text{EXP}(0.1i * t) + \text{Exp}(-0.1i * t) \} + \frac{1}{2}i \{ \text{EXP}(0.1i * t) - \text{Exp}(-0.1i * t) \} \right]$$

$$\text{Slope}_1 = 0.1 * \cos(0.1 * t) - 0.1 * \sin(0.1 * t)$$

$$\text{Slope}_1 = 0.1 * \sqrt{2} * \sin(0.1 * t + 135^\circ)$$

Thus the time equation of Slope\_1 is a sinusoidal wave, i.e. oscillatory behavior.

Note that if the eigenvalues had a real part, i.e.  $\lambda_1 = a + b i$ ;  $\lambda_2 = a - b i$ , then the time equation of Slope\_1 will take this general form:

$$\text{Slope}_1 = C * \text{EXP}(a * \text{TIME}) * \sin (b * \text{TIME} + \psi)$$

Where C is a real number.

## **4 The impact of the feedback loops on the eigenvalues**

### **4.1 The structural foundation of the eigenvalues**

In this section we will describe the relationship between the gains and the eigenvalues. This section is mainly based on the works by Nathan Forrester (Forrester, 1982 & 1983) and Christian Kampmann (Kampmann, 1996). Traditional eigenvalue analysis is based on a compact model representation, called a “condensed form” by N. Forrester and a “reduced form” by C. Kampmann, which focus only on state variables and the associated *net* rates. Such a compact representation constitutes a homomorphic mapping, i.e. an abstraction, of the original model structure and thus constitutes a loss of information. The problem is, therefore, that the results of the analysis performed on a model in its compact form do not easily lend themselves to an interpretation with respect to the model expressed in its original form.

In order to retain the mathematical rigor and convenience offered by the compact form of representing a model and yet not lose the association with the original model, we choose, in this paper, a mini model form (representation) suggested by Scott Guthrie (Guthrie, 1999), as a tool for teaching dynamic systems thinking. In this mini-model form, we retain the individual inflows and outflows and the corresponding rates associated with each of the state variables in the original model. The result is a semi-compact (mini-) model which does not contain any of the auxiliaries of the original model, yet can be recognized as a semi-compact representation of the system portrayed by the original model. The mini-model serves a very important purpose: On the one hand, it is close enough to the compact model to allow us to perform the mathematical and computational analysis facilitated by the compact form. On the other hand, the mini- model is close enough to the original

model so as to allow us to interpret the loops of the mini- model in the context of the system, represented by the original model. Thus we are able to interpret the significance of its feedback loops to the behavior of that model. By building this bridge between a relatively abstract, compact model and the original model, the mini-model constitutes an *interpretable representation* of the *relationship* between *structure* and *behavior*, and thus serves an important *pedagogical* purpose.

To establish the relationship between the eigenvalues and the underlying feedback loops that constitute the structure of the model, we first begin by defining the units of analysis of structure as the gain of a link and the "characteristic gain" (hereafter called the "gain") of a loop. All the links in the mini-model form, relate rate variables to the state variables of the model. The gain of a link, say from state A to rate B is mathematically defined as:

$$g_{AB} = \frac{\partial B}{\partial A}$$

By definition, the gain of a loop is the product of the gains of the links that constitutes that loop. Note: in the process of computing the gain of a loop (in the mini-model), we multiply the gains of links associated with outflow rates by  $-1$ .

The eigenvalues of a model are determined from the loop gains of that model as described by C. Kampmann (Kampmann, 1996):

*"The eigenvalues  $\lambda$  are determined as the roots of the characteristic polynomial  $P(\lambda)$ , and it turns out that the coefficients of this polynomial can be expressed in terms of the gains of the loops in the system."*

Moreover, in some system dynamics models one may be able to identify a subset of the structure of the model that exclusively determine the values of certain eigenvalues. We first begin by introducing some "graph theory" definitions that characterize the structure of a model in its mini-model form. Any pair of state variables, say  $x$  and  $y$ , is said to be strongly connected, if there is a path of directed links from  $x$  to  $y$  and a corresponding directed path from  $y$  to  $x$ <sup>4</sup>. A strongly connected structure component -- in short, a strong structure component -- is a subset of the structure of the model in which any pair of state variables in this subset is

---

<sup>4</sup> Not necessarily a feedback loop

strongly connected. If there is a model consisting of a set of disjoint strong structure components, where no pairs of state variables from two separate components are strongly connected (yet they maybe unidirectional connected), then the strong components of the model *independently determine corresponding subsets of the eigenvalues*. The implication is that individual eigenvalues in such cases are rooted in distinct subcomponents of a model. A documentation of this fact can be found in (Kampmann, 1996).

## **4.2 A measure for the significance of a gain of a unit structure on a certain eigenvalue**

### **4.2.1 The elasticity measure**

We now want to develop a measure for the significance of a gain of a unit structure (link or loop) on a certain eigenvalue. N. Forrester (Forrester, 1982 & 1983) suggests the eigenvalue elasticity as a measure for the significance of a gain of a unit structure, whether link or loop, to a certain eigenvalue. The eigenvalue elasticity is a dimensionless ratio defined as the relative change in an eigenvalue resulting from a relative change of the gain. The eigenvalue elasticities are defined as the fractional response in the eigenvalues to a fractional change in the gains and thus resemble the price elasticities of goods used in economics. The larger the magnitude of an eigenvalue elasticity associated with the gain of a certain unit of structure, the more significant is that structural unit to that particular eigenvalue. Note that we have already suggested a way to rank the eigenvalues with respect to their significance to the model behavior. We can now weigh the significance of each structure unit to a particular eigenvalue, with the significance of that eigenvalue to the model behavior. In this way, we obtain a direct measure of the significance of a structure unit on the model behavior. This will be illustrated in the “yeast cells generation” model described in section 5.

We have established that the eigenvalue elasticity is a key concept in identifying the significance of a structure unit on model behavior, and we now provide a formal definition of this concept.

### 4.2.2 Compact link elasticity

We are going to calculate the elasticity,  $e$ , of an eigenvalue,  $l$ , with respect to the gain,  $g$ , associated with a link in a *compact model*:

$$e = (d l / l) / (d g / g) = (d l / d g) * (g / l) = s * (g / l)$$

where  $s (d l / d g)$  is the sensitivity of the eigenvalue to the gain.

Fortunately, there is a mathematical formula that expresses this sensitivity as a function of the corresponding elements in the associated eigenvectors. The interested reader may consult appendix A.

### 4.2.3 Mini-model link elasticity

The next step is to relate the elasticities of links in the *mini-model form* to the elasticities of links in the compact form. To do so, we will first investigate the relationship between the gains of links in the mini-model form and the gains of links in the compact form. After that, we will investigate the relationship between the eigenvalue sensitivities of links in the mini-model form and the sensitivities of links in the compact form. For this purpose we will use the following simple example as an illustration.

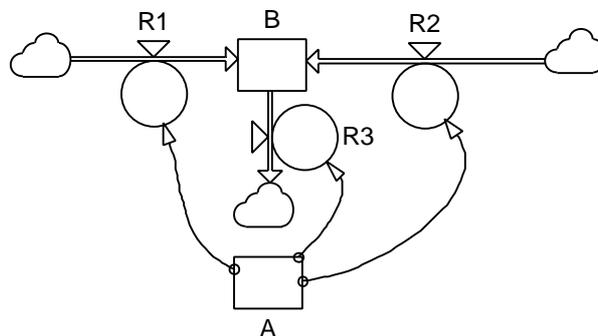


Fig. 21: Mini-model form of the relationship between state variables A and B

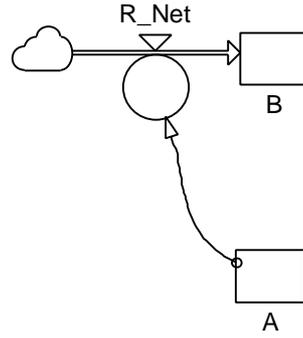


Fig. 22: Compact form of the relationship between state variables A and B

As seen in the figure 21, in the mini-model form, there are two state variables, A and B, where B has two inflow rates R1, R2, and one outflow rate R3. Hence the net rate of B, R\_Net, equals R1+R2-R3. There are three links from state A to the three rates respectively. We will denote the gains of those links g1, g2, g3 respectively.

On the other hand, in the compact form (fig. 22), there is only a single link from state A to the net rate (R\_Net) of state B. We will denote the gain associated with this single link g\_net.

Suppose we introduce a small change in the value of state A, i.e.  $\delta A$ , then the values of the rates will change from  $R1_{original}$ ,  $R2_{original}$ ,  $R3_{original}$  to  $R1_{new}$ ,  $R2_{new}$ ,  $R3_{new}$  respectively.

Now, the mathematical formula for g\_net is:

$$g_{net} = \delta R_{Net} / \delta A = (1 / \delta A) * [ (R1_{new} + R2_{new} - R3_{new}) - (R1_{original} + R2_{original} - R3_{original}) ]$$

$$g_{net} = (1 / \delta A) * [ (R1_{new} - R1_{original}) + (R2_{new} - R2_{original}) - (R3_{new} - R3_{original}) ]$$

$$g_{net} = (\delta R1 / \delta A) + (\delta R2 / \delta A) - (\delta R3 / \delta A)$$

$$g_{net} = g1 + g2 - g3$$

Having identified the relationship between the gains of links in the mini-model form and the gains of links in the compact form, it remains to relate the eigenvalue sensitivities in the mini-model form to the eigenvalue sensitivities in the compact form.

Suppose we introduce a small change in the value of g1, i.e.  $\delta g$ . As

$$g_{\text{net}} = g_1 + g_2 - g_3,$$

is a linear relationship, then a change in  $g_1$  with the magnitude of  $\delta g$ , leads to a change in the value of  $g_{\text{net}}$  with exactly the same magnitude  $\delta g$ , thus inevitably causing a change in the eigenvalue with the magnitude of

$$s * \delta g,$$

where

$$s = (\delta \lambda / \delta g_{\text{net}})$$

is the sensitivity of the eigenvalue to  $g_{\text{net}}$ . As a change in  $g_1$  with magnitude  $\delta g$  causes a corresponding change in the eigenvalue of magnitude  $s * \delta g$ , then the sensitivity of the eigenvalue to  $g_1$  is

$$s * \delta g / \delta g,$$

i.e.  $s$  itself.

Similarly  $g_2$  will also have its eigenvalue sensitivity equal to  $s$ .

As  $g_3$  is associated with an outflow rate, i.e.  $R_3$ , its case will be slightly different: Suppose we introduce a small change in the value of  $g_3$ , i.e.  $\delta g$ . As

$$g_{\text{net}} = g_1 + g_2 - g_3,$$

then a change in  $g_3$  with a magnitude of  $\delta g$ , causes a corresponding change in the value of  $g_{\text{net}}$  with the magnitude of  $-\delta g$ , thus leading inevitably to a change in the eigenvalue with the magnitude of

$$-s * \delta g.$$

Hence the sensitivity of the eigenvalue to  $g_3$  is  $-s$ .

Now, we are finally ready to relate the eigenvalue elasticities in the mini-model form to the eigenvalue elasticities in the compact form.

$$e_{\text{net}} = s * (g_{\text{net}} / \lambda) = (s / \lambda) * (g_1 + g_2 - g_3)$$

$$e_{\text{net}} = (s * g_1 / \lambda) + (s * g_2 / \lambda) + (-s * g_3 / \lambda)$$

i.e.

$$e_{\text{net}} = e_1 + e_2 + e_3$$

where  $e_{\text{net}}$  is the elasticity of  $g_{\text{net}}$ ,  $e_1$  is the elasticity of  $g_1$ ,  $e_2$  is the elasticity of  $g_2$ , and  $e_3$  is the elasticity of  $g_3$ .

In general, the elasticity of a compact link equals the sum of elasticities of those links in the mini-model that constitute that compact link.

Having established the relationship between link elasticities in a compact and a mini-model form, we now turn to the elasticities of loops in the mini-model.

#### 4.2.4 Loop elasticity

The concept of loop elasticities originates from N. Forrester (Forrester, 1982, p. 225). He identified a property of eigenvalue link elasticities that demonstrates the existence of eigenvalues loop elasticities as well:

*"Note that the sum of the eigenvalue elasticities of all links coming into a variable equals the sum of the eigenvalue elasticities of all links leaving the variable... This property implies that a certain numerical value for the eigenvalue elasticity passes from one link to another around each feedback loop. The elasticity of any link is the sum of the elasticities of all the loops that pass through the link. A link could lie in two different feedback loops with opposite effects on an eigenvalue. The elasticities of the two loops would cancel, leaving the link with a small elasticity. Both loops might be considered unimportant, because they contained a link with a small elasticity. The problem of cancellation can be overcome by solving a system of simultaneous linear equations for the eigenvalue elasticity associated with each loop. To make the calculation, first identify all feedback loops in the model and all the links they pass through. Then set up and solve a set of simultaneous equations, where each equation sets the eigenvalue elasticity of a structural link equal to the sum of eigenvalue elasticities of all the loops that contain the link."*

C. Kampmann (Kampmann, 1996) uses the metaphor of a “current” to describe the loop elasticity:

*"In network theory, a set of values that fulfill the condition that the sum of ingoing and outgoing values are equal is called a current in the network."*

In Appendix A, we investigate further the relationship between an eigenvalue and the feedback loops to demonstrate the key role that the loop elasticities play. The interested reader may see section A.11, Appendix A.

In his paper, based on graph theory, (Kampmann, 1996), C. Kampmann addresses a problem that arises from N. Forrester's work: N. Forrester suggested the method of solving simultaneous equations to identify the loop elasticities of a model. The problem is that, in many models, the number of loops will be far larger than the number of links, i.e. the number of unknowns (loop elasticities) is larger than the number of equations (the system of equations is under-determined). Hence, in general, there will not be a solution to those simultaneous equations. Moreover, even if a solution exists, the total number of feedback loops can reach astronomical numbers even for medium sized models. Consequently, the computational task of identifying that solution may be prohibitively large.

To solve this problem, C. Kampmann focuses on an independent set of loops that, in large models, typically contains a significantly smaller number of loops than the total number of feedback loops in the model. This set of independent loops is constructed so that each of the loops constitutes a unique composition of links. This implies that a matrix of linearly independent columns can be used to represent the membership of links in the set of independent loops:

$$\begin{bmatrix} k_1 \\ k_2 \\ \cdot \\ k_n \end{bmatrix} = \begin{bmatrix} 0 & 1 & \cdot & 0 \\ \cdot & a_{ij} & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ 1 & \cdot & \cdot & 0 \end{bmatrix} * \begin{bmatrix} l_1 \\ l_2 \\ \cdot \\ l_m \end{bmatrix}$$

where  $k_i$  is a link elasticity and  $l_j$  is the loop elasticity and where  $a_{ij} = 1$  if the link  $i$  is a component in loop  $j$ , 0 otherwise. The number of loops in this independent set of loops is lower than the number of links. Hence the system of equations, as suggested by N. Forrester, will always be over-determined, yet it will be consistent and thus will have a solution. As Kampmann puts (Kampmann, 1996) it:

*"the most significant contribution [of his paper] is the notion of an independent loop set, which gives grounds for optimism about using the method to large-scale models, even though these will contain millions of feedback loops"*

Note that this set of independent loops is a pertinent description of the model as a whole, which by no means disregard some important structural information in the

model; it only eliminates the redundant structural information associated with the superfluous dependent loops. The purpose of this selection of an independent set of loops is to identify the loop elasticities. Now, there may be several sets of independent loops and, from a practical point of view, the choice is not arbitrary: As C. Kampmann points out, our goal should be to identify an independent set of loops that contains a large subset of relatively comprehensive (i.e. containing relatively many links), insignificant loops and a small subset of relatively specific (i.e. containing relatively few links), significant loops. This is important since we can identify a subset of specific loops that are responsible for the main dynamics of the model, we may legitimately simplify our understanding of the model, and we may utilize this small number of high leverage loops for management purposes.

### **4.3 Non-linear models**

In a non-linear model, the loop gains are dependent on the current state of the model (i.e. the current values of the state variables). Thus the loop gains may change over time. The implication is that the contribution of the loops to each of the eigenvalues will change. This will modify the eigenvalues. Such a change in an eigenvalue will in turn affect the behavior of the model. Therefore, the relationship between the model structure and behavior, which is characterized by the intermediate eigenvalues link, *is a dynamic one that needs to be assessed iteratively over the simulation time period.*

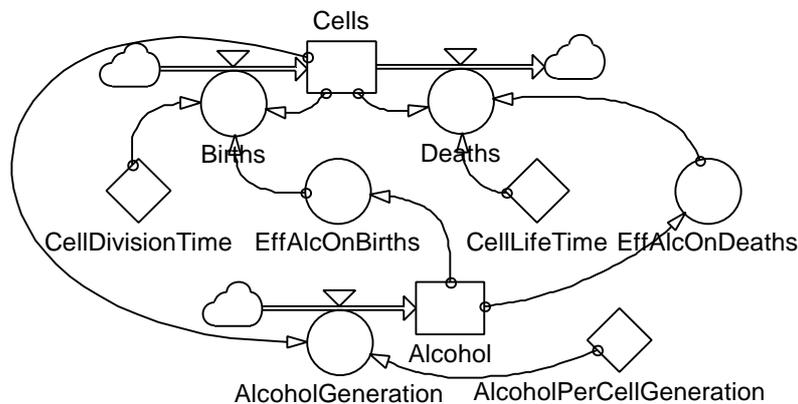
In non-linear models that have been analyzed over a sequence of elementary (analysis) time intervals, we experience a *continuous* change in the characteristic gains of the loops, and thus a *continuous* change in the contribution of the various feedback loops to the individual eigenvalues. Consequently, there is a *continuous* change in the values of the eigenvalues and hence a *continuous* change in the contribution of each of these eigenvalues to the model behavior. By aggregating the results from the sequence of analyses associated with the elementary (analysis) time intervals, we can obtain a characterization of the contribution of each feedback loop to the eigenvalues that govern (dominate) the model behavior over a longer time horizon. In summary, the method outlined allows us to investigate the transient

behavior of complex, non-linear models in view of the underlying structure of these models. To exemplify this, our next example will be a non-linear model.

## 5 Closing Example: The yeast cells generation

### 5.1 Model introduction

We will conclude this paper with a model of yeast cell growth <sup>5</sup>, portrayed in the form of a stock-and-flow diagram in figure 23. Yeast cells produce alcohol that, in high concentrations is toxic to the cells and thus reduces the birth rate and increases the death rate of the cells so that the net growth rate of the cells eventually falls below 0, so as to cause a depletion of the cells. We will apply our method of analysis, which is a formal rigorous mathematical analysis, to understand what are the various feedback loops in the model that generate the observed cell's pattern of behavior.



**Fig. 23: The yeast cells generation example, stock and flow diagram of the model**

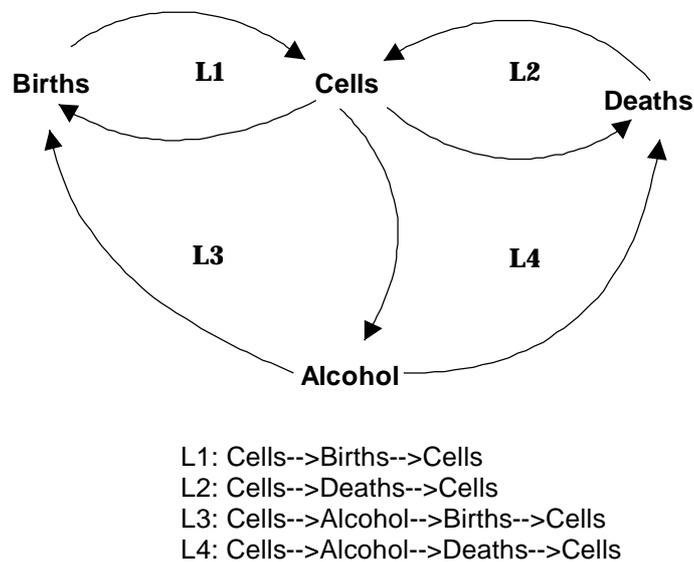
Considering the corresponding causal loop diagram, portrayed in figure 24, the model consists of four feedback loops, L1 – L4. L1 is the cells-births-cells loop. L2 is the cells-deaths-cells loop. L3 is the cells-alcohol (through the AlcoholGeneration rate) - births-cells loop. L4 is the cells-alcohol-deaths-cells loop.

<sup>5</sup> This is a demo model that is not based on real data. Magne Myrtviet is the one who originally suggested this model as an intriguing model to study.

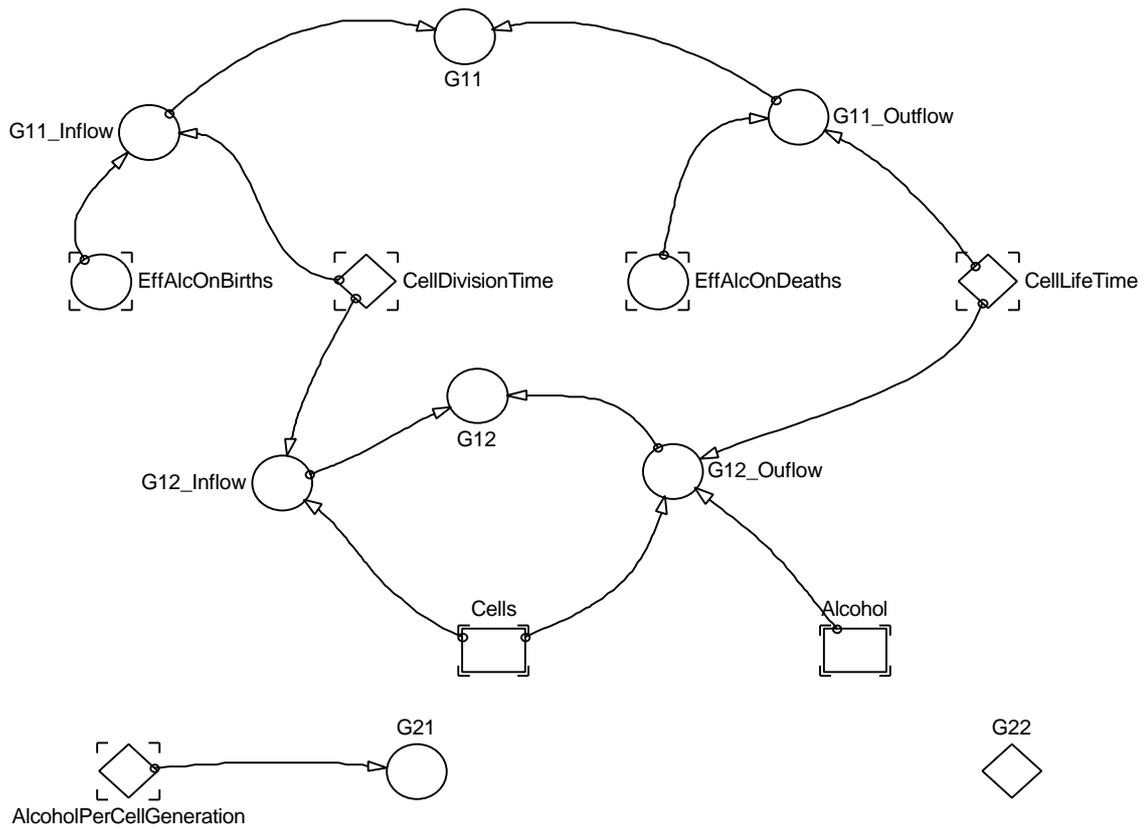
Moreover, in this example, the gains of the links are calculated (analytically) and portrayed in figure 25. In the mini model form, the gains are  $G_{11\_inflow}$  (effect of cells on births),  $G_{11\_outflow}$  (effect of cells on deaths),  $G_{12\_inflow}$  (effect of alcohol on births), and  $G_{12\_outflow}$  (effect of alcohol on deaths),  $G_{21}$  (effect of cells on alcohol generation), and  $G_{22}$  (effect of alcohol on alcohol generation) which are both constant. The gain of L1 (the first loop),  $g(L1)$ , is equal to  $G_{11\_inflow}$ . The gain of L2,  $g(L2)$ , is equal to  $(-1 * G_{11\_outflow})$ . The gain of L3,  $g(L3)$ , is equal to  $(G_{12\_inflow} * G_{21})$ . The gain of L4,  $g(L4)$ , is equal to  $(-1 * G_{12\_outflow} * G_{21})$ .

In the compact form, the gains are  $G_{11}$  (effect of cells on the net rate of the cells), and  $G_{12}$  (effect of alcohol on the net rate of cells); in addition to  $G_{21}$  (effect Cells on alcohol generation), and  $G_{22}$  (effect of alcohol on alcohol generation).

Note: in this case we can analytically calculate the gains; in most non-linear models we must use finite-difference approximations to calculate the gains.



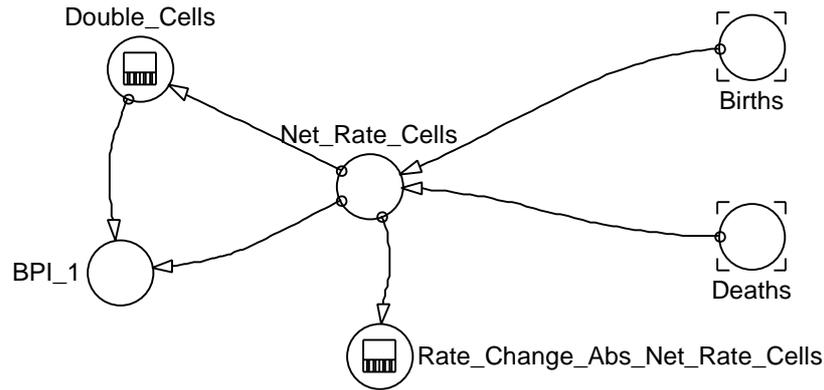
**Fig. 24: The yeast cells generation example, - causal loop diagram of the model**



**Fig. 25: The yeast cells generation example, stock and flow diagram of the calculations of the gains of links (mini-model & compact forms)**

In addition, we calculate the BPI<sub>1</sub>, based on the first and second derivatives of the cell state variable, and the rate of change of the absolute value of the net rate (slope) of the cell state variable (see figure 26).

Table 8 contains the equations of the model organized according to the various stock-and-flow diagrams presented above.



**Fig. 26: The yeast cells generation example, - stock and flow diagram of the calculations of the BPI\_1 & the rate of change of the absolute value of the net rate of cells state**

- *First Part, The Model:*

**init Cells = 1**  
**flow Cells = Births -Deaths**

**init Alcohol = 0**  
**flow Alcohol = AlcoholGeneration**

**Births = (Cells/CellDivisionTime)\*EffAlcOnBirths**  
**Deaths = (Cells/CellLifeTime)\*EffAlcOnDeaths**  
**AlcoholGeneration = Cells\*AlcoholPerCellGeneration**

**EffAlcOnBirths = (-0.1\*Alcohol)+1.1**  
**EffAlcOnDeaths = EXP(Alcohol-11)**

**CellLifeTime = 30**  
**CellDivisionTime = 15**  
**AlcoholPerCellGeneration = 0.01**

- *Second Part, the calculations of the gains of links (mini-model & compact forms):*

*Mini-model:*

**G11\_Inflow = (EffAlcOnBirths/CellDivisionTime)**  
**G11\_Outflow = (EffAlcOnDeaths/CellLifeTime)**  
**G12\_Inflow = (-0.1\* (Cells/CellDivisionTime) )**  
**G12\_Outflow = ( (Cells/CellLifeTime) \* EXP(Alcohol-11))**  
**G21 = AlcoholPerCellGeneration**  
**G22 = 0**

*Compact-model:*

**G11 = G11\_Inflow-G11\_Outflow**  
**G12 = G12\_Inflow-G12\_Outflow**  
**G21 = AlcoholPerCellGeneration**  
**G22 = 0**

- *Third Part, the calculations of the BPI\_1 and the rate of change of the absolute value of the net rate of cells state:*

$$\text{Net\_Rate\_Cells} = \text{Births-Deaths}$$

$$\text{Double\_Cells} = \text{DERIVN}(\text{Net\_Rate\_Cells})$$

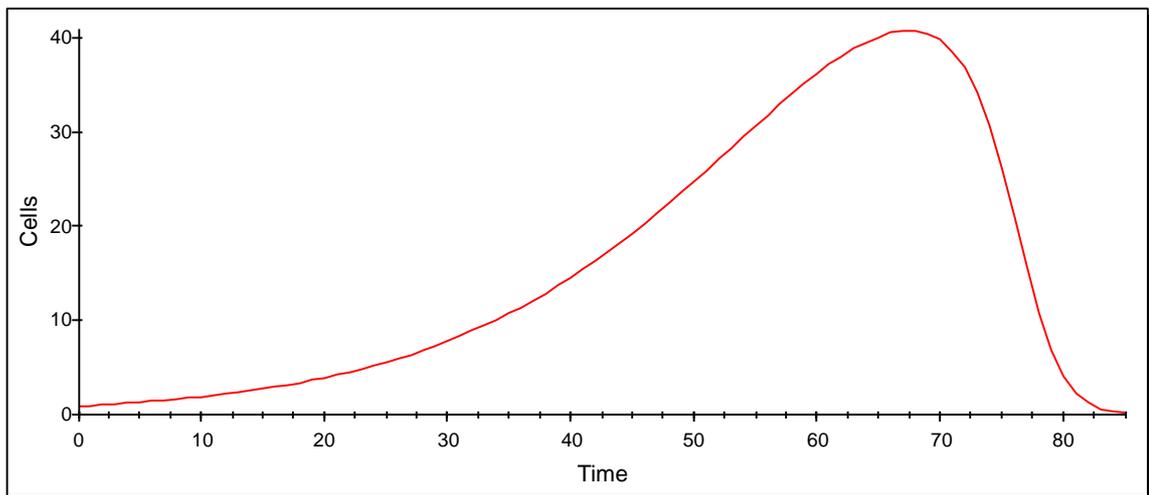
$$\text{BPI\_1} = \text{Double\_Cells} / \text{Net\_Rate\_Cells}$$

$$\text{Rate\_Change\_Abs\_Net\_Rate\_Cells} = \text{DERIVN}(\text{ABS}(\text{Net\_Rate\_Cells}))$$

*Simulation Setup Parameters:*  
 Start Time=0; Stop Time =85; Simulation Time Step = 1

**Table 8: The yeast cells generation example, - equations**

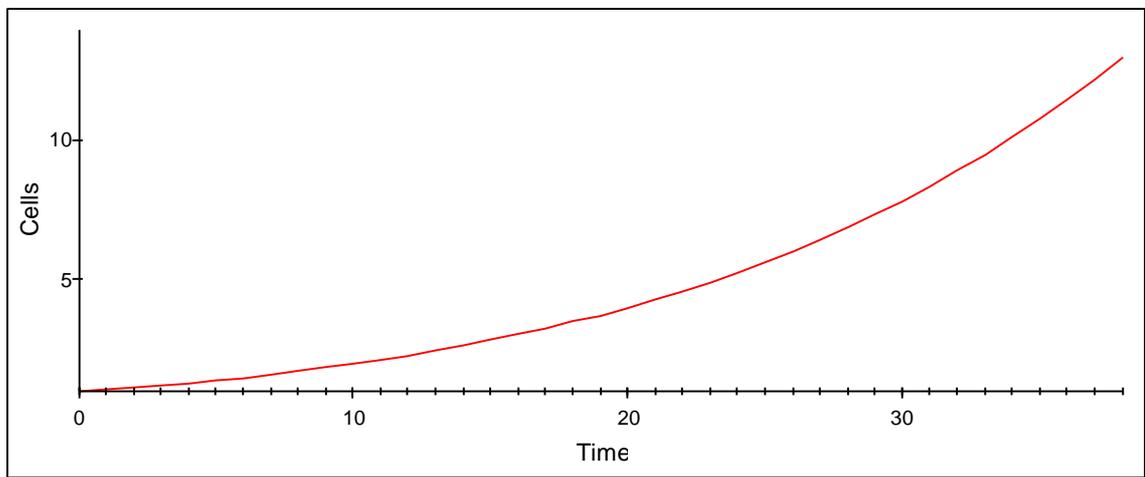
The behavior, exhibited by the cells state, is portrayed in figure 27. At first glance, this behavior can be characterized as “exponential” growth, an overshoot and a collapse. In fact, as we shall see from our mathematical treatment, the behavior of this relatively simple model is quite complex. As we will see below, our mathematical analysis reveals that a number of modes of behavior are hidden under this apparently simple behavior. We will see that a divergent behavior yields to an oscillatory behavior, which subsequently yields to a convergent behavior. The transitions from one mode of behavior to the next are characterized by qualitative changes (or, as we will explain later, bifurcations) in the eigenvalues that characterize the behavior. We will investigate each of these modes of behavior in more detail.



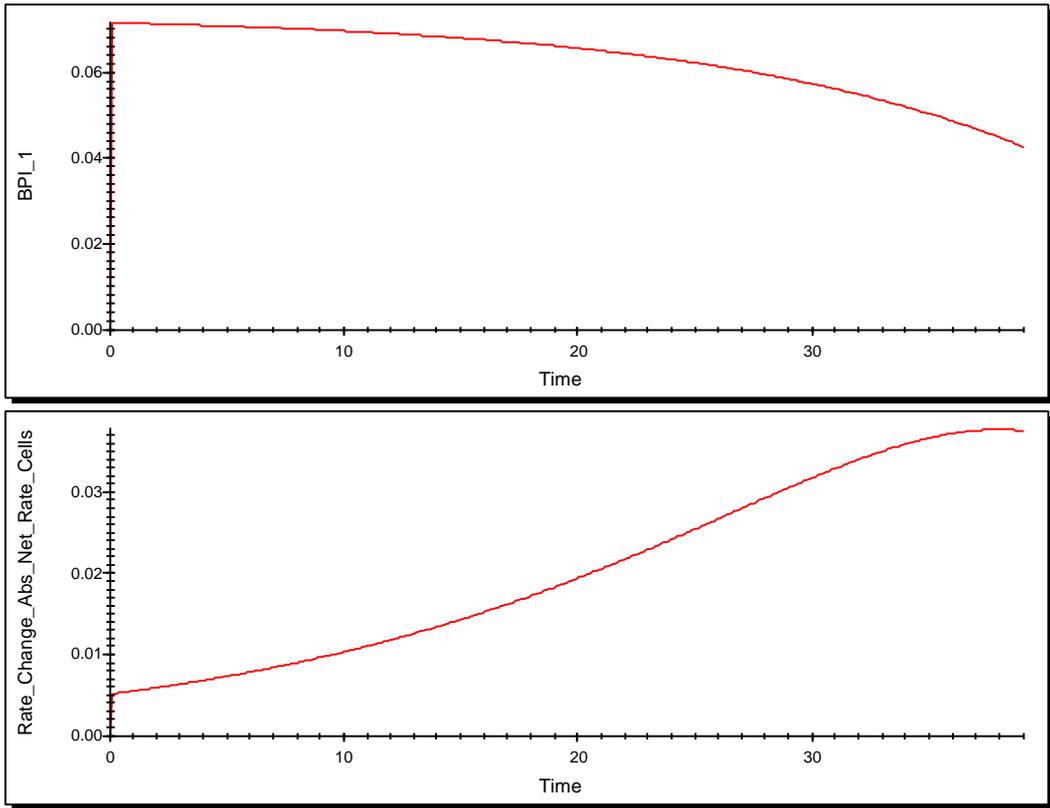
**Fig. 27: The yeast cells generation example, cells state behavior**

## 5.2 First time phase: time interval 0-39

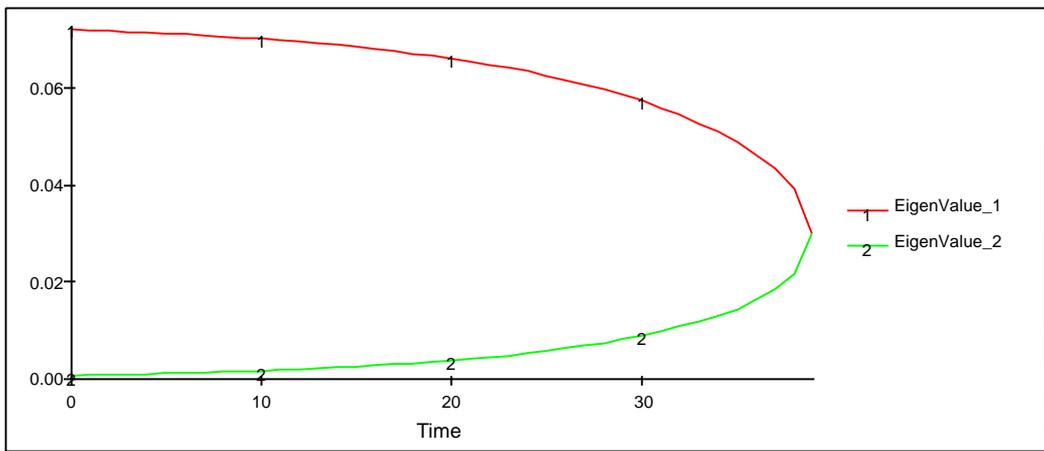
The first time phase (0-39) is characterized by a divergent mode of behavior. This is indicated by the graph of cells state (fig. 28); and by the positive values for BPI\_1 and for the rate of change of the absolute value of the net rate of cells (fig. 29). In this time phase, there are two positive eigenvalues (fig. 30), which also indicates a divergent behavior. Towards the end of this period, the eigenvalues approach each other. Thereafter they bifurcate into a complex eigenvalue pair (as we will discuss in the next phase), consisting of a complex number and its complex conjugate.



**Fig. 28: Cells state behavior in the first phase**



**Fig. 29: BPI\_1 and the rate of change of the absolute value of the net rate of cells state in the first time phase.**



**Fig. 30: Eigenvalues in the first time phase**

In order to identify the feedback loop(s) that is mainly responsible for the divergent behavior in this time phase, we will pick an arbitrary analysis time step during this phase and compute the gains and the eigenvalues at this time step. Then we will identify the eigenvalue(s) that dominates the behavior (the one with the largest impact on behavior) at this analysis time step. Then, for this dominant eigenvalue, we will compute and rank the elasticities of all loops in the model, so as to identify the dominant loop(s). In this phase we will focus on the analysis time step 7.0-8.0 (arbitrarily chosen).

### **Gains, Eigenvalues, Elasticities in the Analysis time step 7.0-8.0**

#### **Mini-model gains:**

$$\begin{aligned} G_{11\_inflow} &= 0.073 \\ G_{11\_outflow} &\approx 0^6 \\ G_{12\_inflow} &= -0.011 \\ G_{12\_outflow} &\approx 0 \end{aligned}$$

#### **Compact\_form gains:**

$$\begin{aligned} G_{11} &= 0.073 \\ G_{12} &= -0.011 \end{aligned}$$

#### **Compact gain matrix:**

$$\mathbf{G} = \begin{bmatrix} 0.073 & -0.011 \\ 0.01 & 0 \end{bmatrix}$$

*Note that always  $G_{21}=0.01$  &  $G_{22}=0$ .*

#### **Eigenvalues**

$$\lambda_1 = 0.071 \text{ \& } \lambda_2 = 0.0015$$

Now, we will rank the eigenvalues according to their impact on the cells state behavior. As we did before (recall section 3.2), we will set the analysis time step equal to the original simulation time step, i.e. 1, and then reduce the simulation time step to 0.1. We start out with a base run whereby we allow for all of the eigenvalues to simultaneously impact the behavior of the state of the cells. Subsequently, we eliminate the dynamics caused by the each eigenvalue, one at a time. Recall that to do

---

<sup>6</sup> To the accuracy of 4 decimal points

this we must calculate the right eigenvectors (at time =7) from the G matrix, and also the initial values of  $\alpha$ s (at time =7) from the right eigenvectors and the values of net rates. Here we will only report the final results, i.e. the impact of stopping the dynamics associated with each eigenvalue on the absolute value of the net rate of state of the cells.

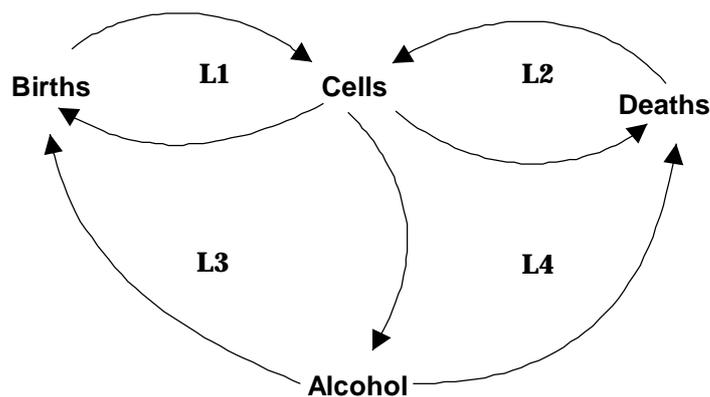
1<sup>st</sup> Case: stop the dynamics associated with the first eigenvalue ( $\lambda_1= 0.071$ ).

2<sup>nd</sup> Case: stop the dynamics associated with the second eigenvalue ( $\lambda_2 = 0.0015$ )

Time	net rate cells  in Base Run	net rate cells  in 1 <sup>st</sup> case	net rate cells  in 2 <sup>nd</sup> Case
7.0	0.119	0.119	0.119
8	0.128	0.119	0.128

**Table. 9: First phase experiment, the impact of stopping the dynamics associated with each eigenvalue on the absolute value of the net rate of cells state.**

From table 9, it is obvious that the *first eigenvalue* ( $\lambda_1= 0.071$ ) is *dominant* at that time. The next step is to draw the “elasticity map” associated with this dominant eigenvalue. The elasticity map is a causal-loop diagram of the model that shows the elasticities (e) of all loops in the model. We will also show the gains (g) of the loops in the model.



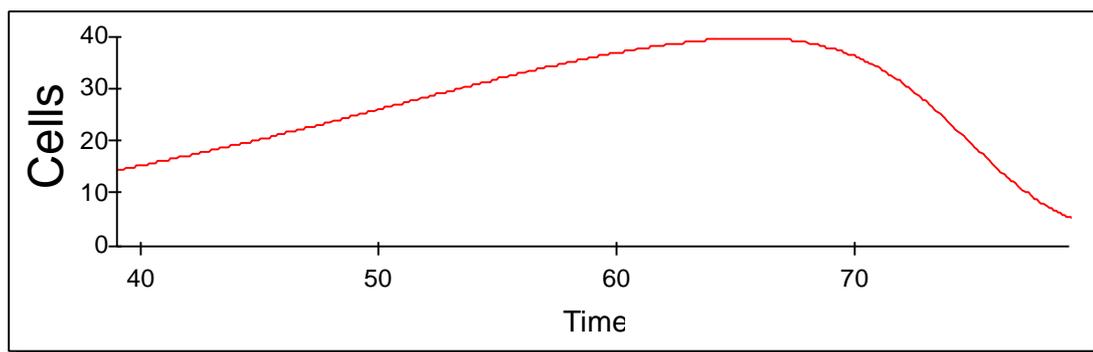
$g(L1) = 0.073$	$e(L1) = 1.044$
$g(L2) = 0$	$e(L2) = 0$
$g(L3) = -0.0001$	$e(L3) = -0.022$
$g(L4) = 0$	$e(L4) = 0$

**Fig. 31: The elasticity map of the dominant eigenvalue ( $\lambda_1= 0.071$ ) in the analysis time step 7-8.**

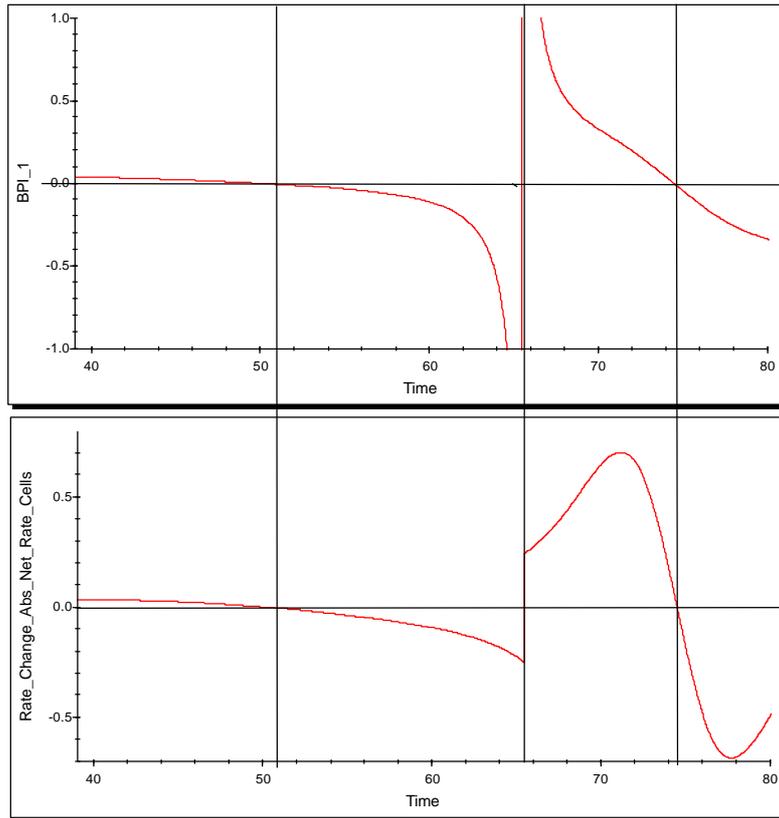
From figure 31, it is obvious that first loop ( $L1$ ), i.e. the cells-births-cells loop, has the largest elasticity; hence one can directly conclude that this is the *dominant* loop at that time;  $L1$  is the loop that mainly contributes to the generation of the divergent behavior observed in the first time phase.

### 5.3 Second time phase: time interval 39-78

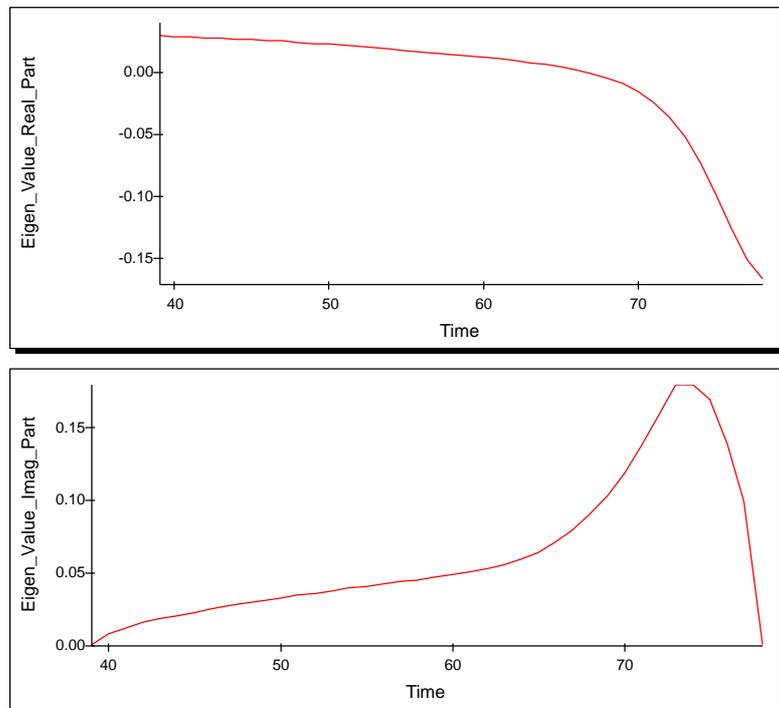
This time phase (39-78) is characterized by an oscillatory behavior. This is indicated by the graph of cells state (fig. 32) and by the fluctuating signs (+/-) for  $BPI_1$  and for the rate of change of the absolute value of the net rate of cells state (fig. 33). At this time there are two complex conjugate pairs of eigenvalues (their real and negative parts are plotted in fig. 34), - an additional indicator of oscillatory behavior (recall section 3.3.2).



**Fig. 32: Cells state behavior in the second phase**



**Fig. 33: BPI\_1 and the rate of change of the absolute value of the net rate of cells state in the second phase**



**Fig. 34: The real and imaginary parts of the complex eigenvalues pair in the second phase**

In the graph above, observe that the imaginary part of the complex eigenvalues pair is equal to zero at the start and the end of the second phase. Also observe that the value of the real part of the complex eigenvalues pair at the beginning of the second phase is equal to the values of both real eigenvalues at the end of the first phase (recall fig. 30), and that the value of the real part of the complex eigenvalues pair at the end of the second phase is equal to the values of both real eigenvalues at the beginning of the third phase (as will be shown in fig. 39). In fact, the points of transitions between different time phases are points of bifurcations in the eigenvalues.

In this time phase, we will repeat the same procedure that we followed in the previous phase. Yet, in this case, we will study two analysis time steps (50-51 and 70-71), rather than a single one as we did previously. The reason for this is that we observed in our various experimentations with the model, that there is shift in loop dominance occurring during this second time phase; thus, by identifying the different dominant loops at these two time steps, we are able to track this shift in loop dominance.

### **Gains, Eigenvalues, Elasticities in the Analysis time step 50.0-51.0**

#### **Mini-model gains:**

$$\begin{aligned} G11\_inflow &= 0.047 \\ G11\_outflow &\approx 0 \\ G12\_inflow &= -0.166 \\ G12\_outflow &= 0.0007 \approx 0 \end{aligned}$$

#### **Compact form gains:**

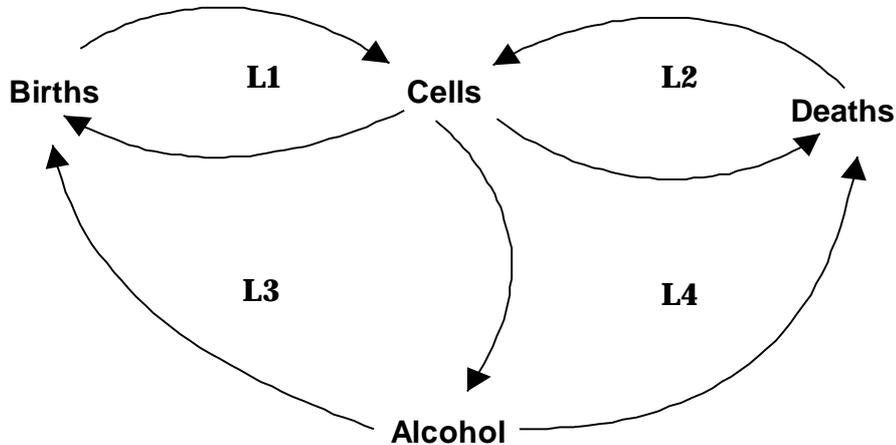
$$\begin{aligned} G11 &= 0.047 \\ G12 &= -0.166 \end{aligned}$$

#### **Compact gain matrix:**

$$\mathbf{G} = \begin{bmatrix} 0.047 & -0.166 \\ 0.01 & 0 \end{bmatrix}$$

$$\lambda_1 = 0.024 + 0.033 i; \lambda_2 = 0.024 - 0.033 i$$

In this case, there is only one compound eigenvalue (complex conjugate pair of eigenvalues). The figure below shows the “elasticity map” associated with this compound eigenvalue.



$g(L1) = 0.047$	$e(L1) = 0 - 0.7 i$	; i.e. $ e(L1)  = 0.7$
$e(L2) = 0$	$e(L2) = 0$	
$e(L3) = -0.0017$	$e(L3) = 0.5 + 0.35 i$	; i.e. $ e(L3)  = 0.61$
$e(L4) = 0$	$e(L4) = 0$	

**Fig. 35: The elasticity map of the compound eigenvalue (the complex conjugate pair) in the analysis time step 50-51.**

From figure 35, it is obvious that loops L1 and L3 are the loops whose elasticities (magnitudes) are the largest; hence both loops *L1* and *L3* are the *dominant* loops at that time.

### Gains, Eigenvalues, Elasticities in the Analysis time step 70.0-71.0

#### Mini-model gains:

$$G_{11\_inflow} = 0.0006 \approx 0$$

$$G_{11\_outflow} = 0.03$$

$$G_{12\_inflow} = -0.266$$

$$G_{12\_outflow} = 1.2$$

#### Compact form gains:

$$G_{11} = -0.03$$

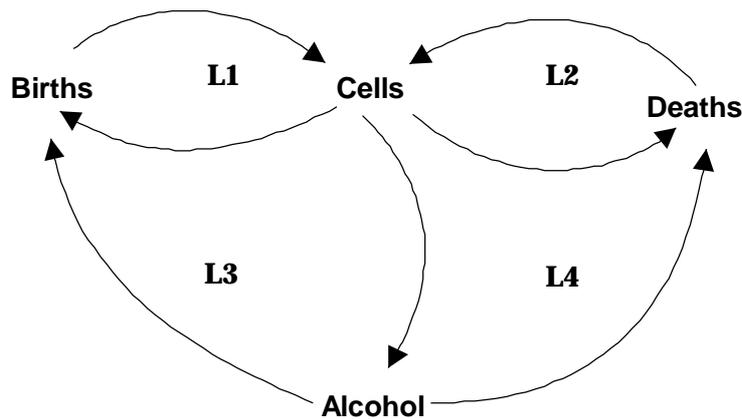
$$G_{12} = -1.466$$

#### Compact gain matrix:

$$\mathbf{G} = \begin{bmatrix} -0.03 & -1.466 \\ 0.01 & 0 \end{bmatrix}$$

$$\lambda_1 = -0.015 + 0.12i; \lambda_2 = -0.015 - 0.12i$$

Also, at that time there is only one compound eigenvalue (complex conjugate pair of eigenvalues). The figure below shows the “elasticity map” associated with this compound eigenvalue.



$g(L1) = 0$	$e(L1) = 0$	
$g(L2) = -0.03$	$e(L2) = 0 + 0.12i$	; i.e. $ e(L1)  = 0.12$
$g(L3) = -0.0027$	$e(L3) = 0.09 - 0.01i$	; i.e. $ e(L3)  = 0.091$
$g(L4) = -0.012$	$e(L4) = 0.41 - 0.05i$	; i.e. $ e(L4)  = 0.41$

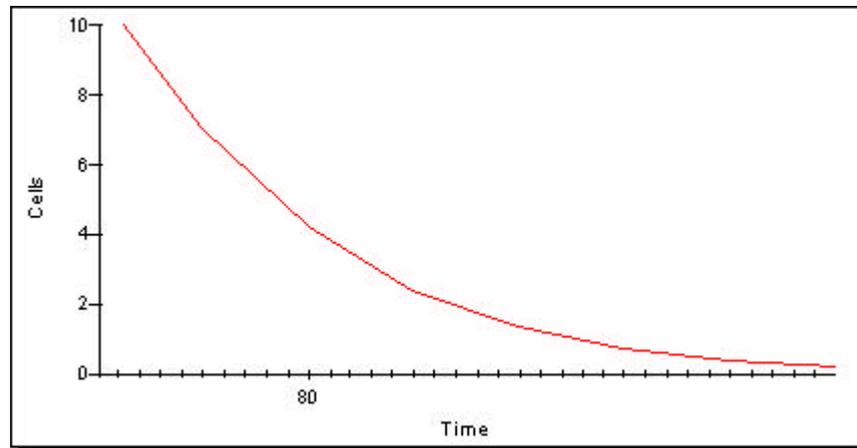
**Fig. 36: The elasticity map of the compound eigenvalue in the analysis time step 70-71**

From figure 36, it is obvious that loop  $L4$  is the *dominant* loop at the time step 70-71, while loops  $L1$  and  $L2$  play a less significant role. Recall that, at the time step 50-51, loops  $L1$  and  $L3$  were the dominant loops; hence there has been a gradual shift in loop dominance from time 50 till time 70. This shift in loop dominance did not qualitatively change the eigenvalues (they remain a complex pair of eigenvalues). Yet what did change is the sign of the real part of the complex eigenvalues pair (recall that the real part was equal to 0.024 at time 50, and it was equal to  $-0.015$  at time 70). A positive real part is an indication of a diverging (oscillatory) behavior, while a negative one is an indication of a converging (oscillatory) behavior. This shift, from a diverging oscillation at the start of the second time phase, to a converging one in the end of the phase, is consistent with the fact that this oscillatory behavior takes over

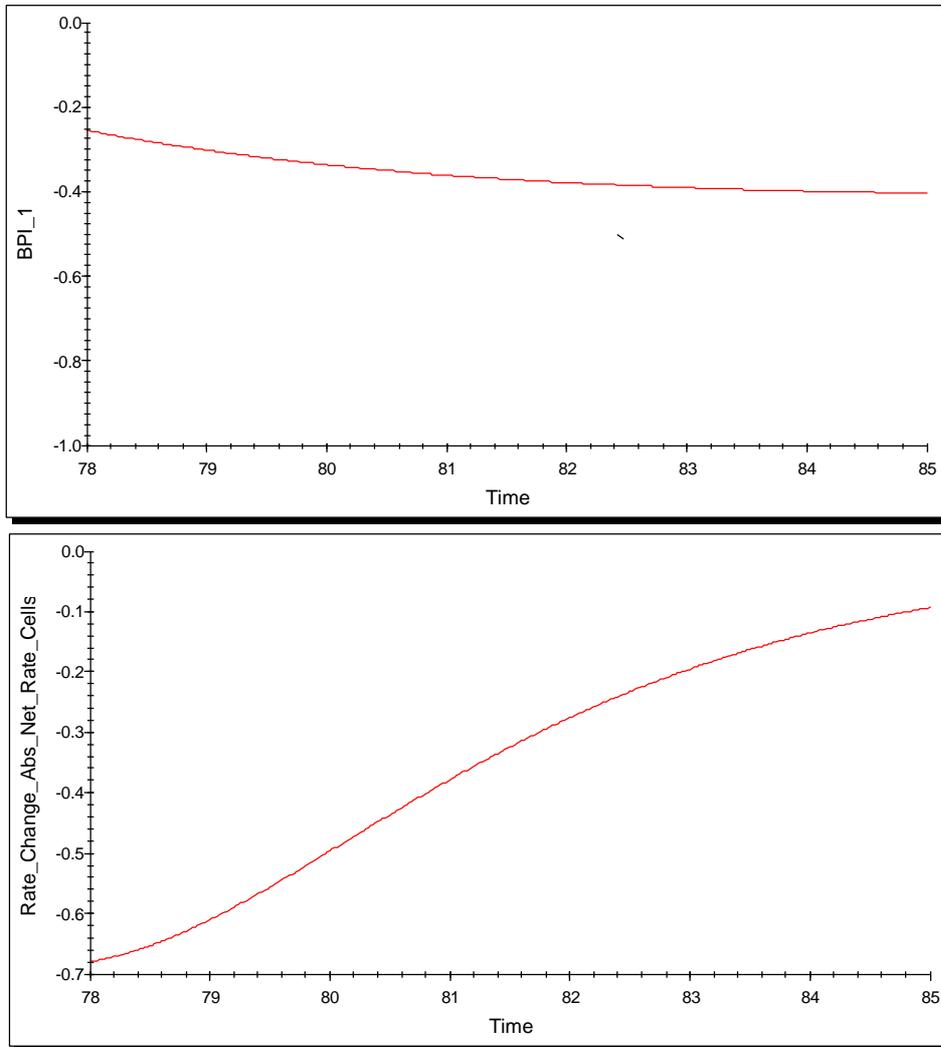
from a divergent behavior in the first phase, and then yields to a convergent behavior in the last phase (as we will see below).

### 5.4 Third time phase: time interval 78-85

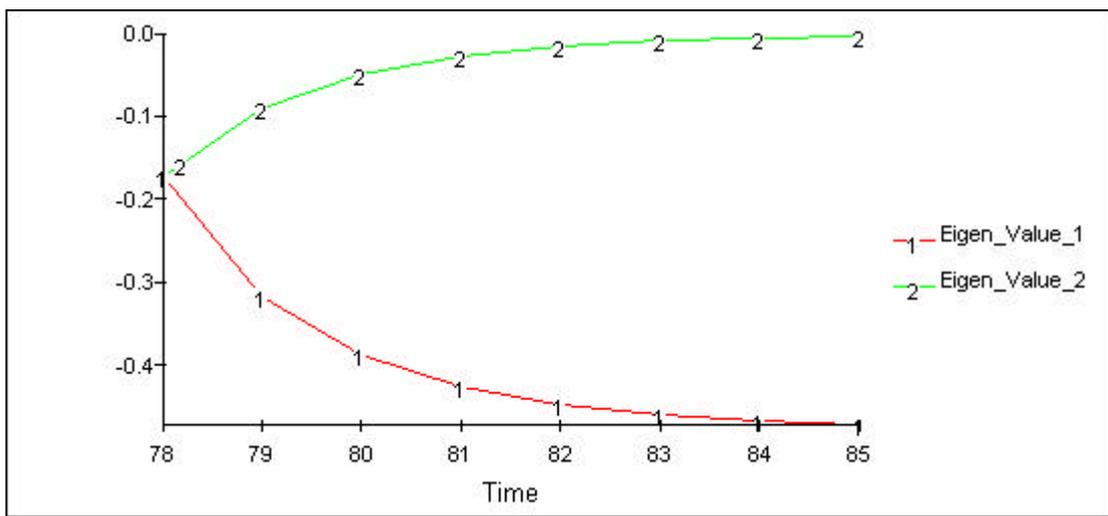
This time phase (78-85) is characterized by a convergent behavior, indicated by the graph of state of the cells (fig. 37) and by the negative values for BPI\_1 and for the rate of change of the absolute value of the net rate of the state of the cells (fig. 38). In this phase, there are two negative eigenvalues (fig. 39), - another indicator for convergent behavior.



**Fig. 37: Cells state behavior in the third phase**



**Fig. 38: BPI\_1 and the rate of change of the absolute value of the net rate of cells state in the third phase**



**Fig. 39: Eigenvalues in the third phase.**

In this phase, we chose (arbitrarily picked) time step 80-81 to study

## Gains, Eigenvalues, Elasticities in the Analysis time step 80.0-81.0

### Mini-model gains:

$$\begin{aligned} G11\_inflow &= -0.017 \\ G11\_outflow &= 0.413 \\ G12\_inflow &= -0.028 \\ G12\_outflow &= 1.736 \end{aligned}$$

### Compact\_form gains:

$$\begin{aligned} G11 &= -0.43 \\ G12 &= -1.764 \end{aligned}$$

### Compact gain matrix:

$$\mathbf{G} = \begin{bmatrix} -0.43 & -1.764 \\ 0.01 & 0 \end{bmatrix}$$

$$\lambda_1 = -0.384 \text{ \& } \lambda_2 = -0.046$$

As we did at the analysis time step 7-8, we are going to rank the eigenvalues according to their impact on the behavior. Here are the results.

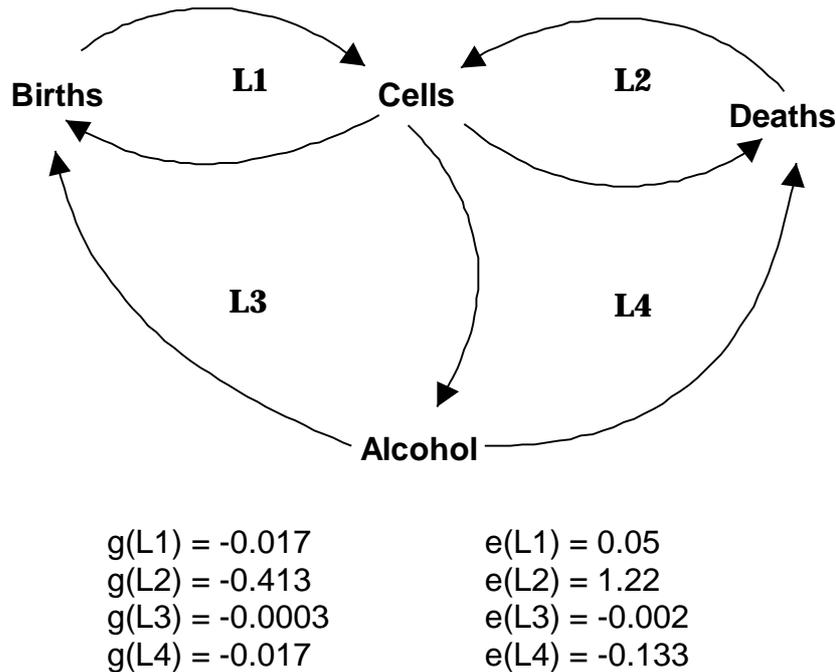
1<sup>st</sup> Case: stop the dynamics associated with the first eigenvalue ( $\lambda_1 = -0.384$ ).

2<sup>nd</sup> Case: stop the dynamics associated with the second eigenvalue ( $\lambda_2 = -0.046$ )

Time	net rate cells  in Base Run	net rate cells  in 1 <sup>st</sup> case	net rate cells  in 2 <sup>nd</sup> Case
<b>80.0</b>	<b>0.119</b>	<b>0.119</b>	<b>0.119</b>
<b>81.0</b>	<b>0.128</b>	<b>0.119</b>	<b>0.128</b>

**Table. 10: Third phase experiment, the impact of stopping the dynamics associated with each eigenvalue on the absolute value of the net rate of cells state.**

From table 10, it is obvious that the *first eigenvalue* ( $\lambda_1 = -0.384$ ) is *dominant* in this case. Below is “elasticity map” associated with this dominant eigenvalue.



**Fig. 40: The elasticity map of the dominant eigenvalue ( $\lambda_1 = -0.384$ ) in the analysis time step 80-81.**

From figure 40, it is obvious that second loop ( $L2$ ), i.e. the cells-deaths-cells loop, is the *dominant* loop at that time;  $L2$  is the loop that is mainly generating the convergent behavior observed in this phase.

## 6 Conclusion

Using the system dynamics method, we should be able to understand the nature of the complexity governing the real world – i.e. how structure drives behavior and how the resulting behavior causes shifts in structural dominance in complex non-linear, dynamics systems. This would then serve as a foundation for problem identification, problem analysis, for problem solving in the form of policy design and strategy development and, thus for the management of complex, dynamic systems. Yet, until now, the explanatory power of system dynamics has not been sufficient. Most system dynamicists (Ford, 1999; Richardson, 1984; Sterman, 2000, etc.) agree that a rigorous, scientific method for identifying dominant feedback loops is *required* in order to boost the explanatory power of system dynamics field. In this paper we document such a method.

The method suggested consists of the following steps: First, we have provided a characterization of model behavior in the form of the slope, the curvature, the individual  $BPI_{x,s}$ , and the overall BPI. We have also documented the significance of the rate of change of the length of the slope vector and its components to the behavior of a model. Moreover, we have demonstrated that the overall BPI and the  $BPI_{x,s}$  are qualitative indicators that serve as proxies for the rate of change of the length of the slope vector and its components, respectively. Second, we described the relationship between the eigenvalues and the rate of change of the length of the slope vector and its components. Third, we described the relationship between the gains of loops in the model, i.e. the structure of the model and the eigenvalues. Hence we demonstrated that the eigenvalues serve as a link between the structure and the behavior in dynamic models.

In a nutshell, our method of identifying dominant feedback loops is a two-stage filtration process. In the first stage, we rank eigenvalues according to their impact on the behavior. Then we select the dominant eigenvalue(s). In the second stage we rank loops according to their significance to the dominant eigenvalue(s). Then we select the dominant loops. As the dominant eigenvalues and the dominant loops may change with time, we iterate this two-stage filtration process over the simulation period. The main goal of this filtration process is to identify the “*core structure*” of the model, - where the core structure is defined as the smallest number of loops that is responsible for generating the behavior of the model over time. The core structure is a presentable distillation of the structure of the model (an archetype). Another important goal is identifying the time phases in which different parts of the core structure are active (i.e. dominating the behavior). By focusing on the active part of the core structure in a certain time phase, we may legitimately simplify our understanding of the model. And we may utilize the small number of high leverage loops, which are present in the active part, for management purposes.

So what is the *Next step*? To continue research in this direction, it is *essential* to develop a *user-friendly software package* that can automate this method of identifying dominant feedback loops for any SD model. Steps have already been taken in this direction. Most of the algorithms that are required to compute the gains, the

eigenvalues and elasticities have been implemented. In future research, we may investigate the rare case of a defective gain matrix (see Appendix A, Corollary A.3). Also steps have been taken to design a friendly user-interface that presents the steps in and the results from this sophisticated mathematical method in a simple and intuitive way. For more details, the reader can refer to (Myrtveit & Saleh 2000). The software package to be developed shows promise as an important tool that may contribute significantly to our understanding of large complex non-linear SD models. Moreover, this package we enable us to further test our method on a variety of large, complex models. Thus we can further validate our method and improve on it.

## References

- Barlas, Y. (1989): Multiple Tests for Validation of System Dynamics Type of Simulation Models. *European Journal of Operational Research*. Vol. 42 no. 1. pp. 59-87
- Davidson, 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.
- Goos, G.; Hartmanis, J. (1976): *Matrix Eigensystem Routines \_ EISPACK Guide*. Second Edition. *Lecture Notes in Computer Science*. Springer-Verlag.
- Guthrie, S. (1999): *Mini-Model Presentations \_ A Tool for Teaching Dynamic Systems Thinking*. *Proceedings of the 1999 International System Dynamics Conference*. Willington.
- 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.

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

Watkins, D. (1991): *Fundamentals of Matrix Computations*. John Wiley & Sons.

Wilkinson, J. (1965): *The Algebraic Eigenvalue Problem*. Oxford Univ. Press.

## Appendix A: Matrix eigensystem theorems

### A.1 Definitions of eigenvalues, eigenvectors, and the characteristic polynomial

A number  $\lambda$  is an eigenvalue of a  $n \times n$  real matrix  $\mathbf{A}$ , if there is a nonzero  $n$  vector  $\mathbf{r}$  such that:

$$\mathbf{A} * \mathbf{r} = \lambda * \mathbf{r}$$

The corresponding vector  $\mathbf{r}$  is called the right-eigenvector of matrix  $\mathbf{A}$ . It is called “right-eigenvector” in the sense that it appears on the right-hand side of matrix  $\mathbf{A}$  in the previous equation. Vector  $\mathbf{r}$  has  $n$  elements, where at least one element is a nonzero element.

In the meanwhile, the left-eigenvector  $\mathbf{l}$  is defined by the following equation:

$$\mathbf{l}^T * \mathbf{A} = \lambda * \mathbf{l}^T$$

Where  $\mathbf{l}^T$  is the transpose of  $\mathbf{l}$

It is called “left-eigenvector” in the sense that it appears on the left-hand side of matrix  $\mathbf{A}$  in the previous equation. Vector  $\mathbf{l}$  has  $n$  elements, where at least one element is a nonzero element.

Note: It is clear (from the previous equations) that eigenvectors (right and left) are defined only to within a scalar multiple. If  $\mathbf{v}$  is an eigenvector, then so is  $\alpha\mathbf{v}$  for any nonzero scalar  $\alpha$ .

The equation that defines the eigenvalue  $\lambda$  (i.e.  $\mathbf{A} * \mathbf{r} = \lambda * \mathbf{r}$ ) can be written in the following form:

$$[\mathbf{A} - \lambda\mathbf{I}] * \mathbf{r} = 0$$

Where  $\mathbf{I}$  is  $n \times n$  identity matrix.

The previous equation has a nonzero solution (i.e. a vector  $\underline{\mathbf{r}}$  whose elements are not all zero) if and only if  $\text{Det}[\mathbf{A} - \lambda\mathbf{I}] = 0$  (i.e. the determinant equals zero).

Recall that if the  $\text{Det}[\mathbf{A} - \lambda\mathbf{I}]$  has a nonzero value, then  $[\mathbf{A} - \lambda\mathbf{I}]^{-1}$  (the inverse) does exist, and the previous equation could be multiplied by it yielding:

$$[\mathbf{A} - \lambda\mathbf{I}]^{-1} * [\mathbf{A} - \lambda\mathbf{I}] * \mathbf{r} = \mathbf{0}$$

$$\underline{\mathbf{r}} = \mathbf{0}$$

Which is a contradiction.

The equation:  $\text{Det}[\mathbf{A} - \lambda\mathbf{I}]$  is a polynomial of the n degree in the variable  $\lambda$ .

$$\text{Det}[\mathbf{A} - \lambda\mathbf{I}] = \lambda^n + c_1 \lambda^{n-1} + \dots + c_{n-1} \lambda + c_n$$

It is called the characteristic polynomial of matrix  $\mathbf{A}$ ; where  $c_i$  ( $i = 1 \dots n$ ) are the coefficients of the characteristic polynomial.

The “n” roots of the characteristic polynomial are the eigenvalues of matrix  $\mathbf{A}$ . The roots of the characteristic polynomial can be either real numbers or complex ones; yet if a complex number is a root for the characteristic polynomial, then its complex conjugate must also be a root.

Each element in an eigenvector (right or left), which is associated with a real eigenvalue, is a real number.

Each element in an eigenvector (right or left), which is associated with a complex eigenvalue, is, in general, a complex number that has its complex conjugate counterpart element in the eigenvector that is associated with the complex conjugate eigenvalue.

In the following theorems, we will use the term “*distinct eigenvalues*”. Distinct eigenvalues implies distinct (non-repeated) roots to the characteristic polynomial. Equal eigenvalues coming from multiple roots are called “*degenerate*”.

**Theorem A.2:** For any real square matrix  $\mathbf{A}$ , the right-eigenvectors associated with the distinct eigenvalues are linearly independent.

**Proof**

Say that we have  $q$  distinct eigenvalues. Now, we will assume that the  $q$  right-eigenvectors (associated with the  $q$  distinct eigenvalues) are linearly dependent; and then we will falsify this hypothesis.

In mathematical term our hypothesis (that we want to falsify), can be stated as:

$$\sum_{i=1}^p \alpha_i * \underline{\mathbf{r}}_i = 0 \dots(1)$$

Where  $p$  is the “*smallest*” number of linearly dependent right-eigenvectors

The value of  $p$  satisfies the following inequality:

$$1 < p \leq q$$

$\underline{\mathbf{r}}_i$  is the right-eigenvector associated with the eigenvalue  $\lambda_i$

And  $\alpha_i$  is a nonzero scalar value.

Multiplying equation (1) by  $\mathbf{A}$  yields:

$$\sum_{i=1}^p \alpha_i * \mathbf{A} * \underline{\mathbf{r}}_i = \sum_{i=1}^p \alpha_i * \lambda_i * \underline{\mathbf{r}}_i = 0 \dots(2)$$

Multiplying equation (1) by  $\lambda_p$  yields:

$$\sum_{i=1}^p \alpha_i * \lambda_p * \underline{\mathbf{r}}_i = 0 \dots(3)$$

In equations (2) and (3) the summand  $\alpha_p * \lambda_p * \underline{\mathbf{r}}_p$  is identical, then subtracting equation (2) from (3) yields:

$$\sum_{i=1}^{p-1} \alpha_i * (\lambda_p - \lambda_i) * \underline{\mathbf{r}}_i = 0$$

Since the eigenvalues are distinct, then none of the coefficients in the previous equation is equal to zero, thus we have (p-1) linearly dependent right-eigenvectors. This contradicts our hypothesis that p is the “*smallest*” number of linearly dependent right-eigenvectors. Therefore, we can directly conclude that the q right-eigenvectors are linearly independent.

**Corollary A.3:** For any real square matrix A (nxn) with n distinct eigenvalues  $\{\lambda_1 \dots \lambda_n\}$ ; its n right-eigenvectors span the whole n-dimensional space. That is every real vector  $\underline{\mathbf{x}}$  in the n-dimensional space has a unique representation as:

$$\mathbf{x} = \sum_{i=1}^n \alpha_i * \underline{\mathbf{r}}_i$$

Where  $\underline{\mathbf{r}}_i$  is the right-eigenvector associated with the eigenvalue  $\lambda_i$

And  $\alpha_i$  is a scalar value (can be zero).

In matrix form equation  $\mathbf{x} = \sum_{i=1}^n \mathbf{a}_i * r_i$  can be written as:

$$\underline{\mathbf{x}} = \mathbf{R} * \underline{\mathbf{a}} \quad \dots(1)$$

Where  $\mathbf{R} = [\underline{\mathbf{r}}_1 \quad \underline{\mathbf{r}}_2 \quad \dots \quad \underline{\mathbf{r}}_n]$ ;  $\underline{\mathbf{r}}_i$  is the right-eigenvector associated with the eigenvalue  $\lambda_i$

$\underline{\mathbf{a}}$  is a n vector.

The first step is to illustrate that  $\mathbf{R}^{-1}$  (the inverse of  $\mathbf{R}$ ) always exist.

The inverse  $\mathbf{R}^{-1}$  exists if, and only if,  $\text{rank}(\mathbf{R})=n$ .

The rank of a matrix equals the maximum number of linearly independent column vectors in this matrix.

In the previous theorem, we demonstrated that the right-eigenvectors of the distinct eigenvalues are linearly independent. As we have  $n$  distinct eigenvalues, we can conclude that  $\text{rank}(\mathbf{R})=n$  and that  $\mathbf{R}^{-1}$  always exist.

Now equation (1) can be written as:

$$\underline{\mathbf{a}} = \mathbf{R}^{-1} * \underline{\mathbf{x}}$$

Since, there can be only unique inverse for a matrix, then for every real vector  $\underline{\mathbf{x}}$  there is a corresponding unique  $\underline{\mathbf{a}}$  vector.

Note that the term “*complete right-eigenvectors*” is used to describe the fact that the right-eigenvectors span the whole  $n$ -dimensional space. Incomplete right-eigenvectors can only occur where there are degenerate eigenvalues. But even in the case of degenerate eigenvalues, the right-eigenvectors will usually (but not always) be complete. In the rare case of incomplete right-eigenvectors, matrix  $\mathbf{A}$  is called a “*defective matrix*”.

**Theorem A.4:** For any real square matrix  $\mathbf{A}$  ( $n \times n$ ) with  $n$  distinct eigenvalues  $\{\lambda_1 \dots \lambda_n\}$ ; its two sets of right-eigenvectors and left-eigenvectors form a bi-orthogonal system. That is, the left-eigenvector of one eigenvalue is orthogonal to the right-eigenvector of the other, while the left and right eigenvectors of the same eigenvalue are not orthogonal to each other.

Note: The two vectors  $\underline{\mathbf{x}}$  and  $\underline{\mathbf{y}}$  (each having  $n$  elements) are called orthogonal if the angle ( $\theta$ ) between them is  $90^\circ$ . The formula of the angle  $\theta$  is:

$$\text{Cos } \theta = \frac{\underline{\mathbf{x}} \cdot \underline{\mathbf{y}}}{|\underline{\mathbf{x}}| * |\underline{\mathbf{y}}|}$$

It is clear that the two vectors  $\underline{\mathbf{x}}$  and  $\underline{\mathbf{y}}$  are orthogonal, only if their inner product (i.e.  $\underline{\mathbf{x}} \cdot \underline{\mathbf{y}}$ ) is equal to zero. Another way to express the inner product is  $\underline{\mathbf{x}}^T * \underline{\mathbf{y}}$ . Thus, mathematically the above theorem can be stated as:

$$\underline{\mathbf{l}}_i^T * \underline{\mathbf{r}}_j = 0 \text{ if } i \neq j$$

$$\mathbf{l}_j^T * \mathbf{r}_i \neq 0 \text{ if } i = j$$

$$\text{for } j = 1 \dots n \text{ \& } i = 1 \dots n$$

**Proof**

First, we are going to prove that:

$$\mathbf{l}_j^T * \mathbf{r}_i = 0 \text{ if } i \neq j$$

Starting from the definition of the eigenvalue:

$$\mathbf{A} * \mathbf{r}_i = \lambda_i * \mathbf{r}_i$$

We are going to use the fact that the transpose of a product (of several matrices and vectors) equals the product of the transposed factors, taken in reverse order.

Thus, transposing both sides of the above equation yields:

$$\mathbf{r}_i^T * \mathbf{A}^T = \lambda_i * \mathbf{r}_i^T$$

Post-Multiplying both sides by  $\mathbf{l}_j$  yields:

$$\mathbf{r}_i^T * \mathbf{A}^T * \mathbf{l}_j = \lambda_i * \mathbf{r}_i^T * \mathbf{l}_j \dots (1)$$

Now, we will return back to the definition of the left eigenvector:

$$\mathbf{l}_j^T * \mathbf{A} = \lambda_j * \mathbf{l}_j^T$$

Transposing both sides of the equation yields:

$$\mathbf{A}^T * \mathbf{l}_j = \lambda_j * \mathbf{l}_j$$

Pre-multiplying both sides by  $\mathbf{r}_i^T$  yields:

$$\mathbf{r}_i^T * \mathbf{A}^T * \mathbf{l}_j = \lambda_j * \mathbf{r}_i^T * \mathbf{l}_j \dots (2)$$

Subtracting (1) from (2) yields:

$$0 = (\lambda_j - \lambda_i) * \mathbf{r}_i^T * \mathbf{l}_j$$

Since the eigenvalues are distinct, then:

$$\mathbf{r}_i^T * \mathbf{l}_j = 0$$

The second step is to prove that:

$$\mathbf{l}_j^T * \mathbf{r}_i \neq 0 \text{ if } i = j$$

This proof is simpler than the previous one. In this proof, we are going to assume that  $\mathbf{l}_i$  is orthogonal to  $\mathbf{r}_i$ , then we will falsify this hypothesis.

If  $\mathbf{l}_i$  was orthogonal to  $\mathbf{r}_i$ , then it would be orthogonal to  $\mathbf{r}_1, \mathbf{r}_2, \dots, \mathbf{r}_n$ . Since  $\mathbf{r}_1, \mathbf{r}_2, \dots, \mathbf{r}_n$  are linearly independent, then this would make  $\mathbf{l}_i$  orthogonal to the whole n-space; which is something impossible, as the only n vector that is orthogonal to the whole n-space is the null-vector.

**Corollary A.5:** For any real square matrix  $\mathbf{A}$  (nxn) with n distinct eigenvalues  $\{\lambda_1, \dots, \lambda_n\}$  the transpose of the matrix of left-eigenvectors is the inverse of the matrix of right-eigenvectors; i.e.  $\mathbf{L}^T = \mathbf{R}^{-1}$

**Where**

$$\mathbf{L} = [\mathbf{l}_1 \quad \mathbf{l}_2 \quad \dots \quad \mathbf{l}_n]$$

$\mathbf{l}_i$  is the left-eigenvector associated with the eigenvalue  $\lambda_i$

$$\mathbf{R} = [\mathbf{r}_1 \quad \mathbf{r}_2 \quad \dots \quad \mathbf{r}_n]$$

$\mathbf{r}_i$  is the right-eigenvector associated with the eigenvalue  $\lambda_i$

Comment: One can consider the matrix of left-eigenvectors as a “*mirror image*” of the matrix of right-eigenvectors.

We will denote the matrix  $[\mathbf{L}^T * \mathbf{R}]$  by  $\mathbf{X}$ . From the bi-orthogonal relationship between left-eigenvectors and right-eigenvectors (the above theorem), one can directly infer that  $\mathbf{X}$  will be in the form:

$$\mathbf{X} = \begin{bmatrix} \mathbf{x}_{11} & 0 & 0 & 0 \\ 0 & \mathbf{x}_{22} & 0 & 0 \\ 0 & 0 & . & 0 \\ 0 & 0 & 0 & \mathbf{x}_{nn} \end{bmatrix}$$

As we stated before (section A.1), it is a general property that eigenvectors are defined only to within a scalar multiple. If  $\underline{\mathbf{v}}$  is an eigenvector, then so is  $\alpha \underline{\mathbf{v}}$  for any nonzero scalar  $\alpha$ . As  $x_{ii}$  equals  $\underline{\mathbf{l}}_i^T * \underline{\mathbf{r}}_i$ , then we can normalize  $x_{ii}$  to equal unity. This normalization process will not affect the other elements in matrix  $\mathbf{X}$ .

The common normalization for the eigenvectors is to make the length of  $\underline{\mathbf{r}}_i$  equals unity, by multiplying  $\underline{\mathbf{r}}_i$  with an appropriate scalar value. Then to make the inner product of  $\underline{\mathbf{r}}_i$  and  $\underline{\mathbf{l}}_i$  (i.e.  $\underline{\mathbf{l}}_i^T * \underline{\mathbf{r}}_i$ ) equals unity, by multiplying  $\underline{\mathbf{l}}_i$  with an appropriate scalar value.

After normalizing all the diagonal elements in matrix  $\mathbf{X}$  to unity, matrix  $\mathbf{X}$  will be equal to:

$$\mathbf{X} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \mathbf{I}$$

$$\text{As } \mathbf{L}^T * \mathbf{R} = \mathbf{X} = \mathbf{I}$$

$$\text{Thus, } \mathbf{L}^T = \mathbf{R}^{-1}$$

**Corollary A.6:** Any real square matrix  $\mathbf{A}$  ( $n \times n$ ) with  $n$  distinct eigenvalues  $\{\lambda_1 \dots \lambda_n\}$  can be put in the following diagonal form:

$$\mathbf{L} = \mathbf{L}^T * \mathbf{A} * \mathbf{R}$$

Where  $\Lambda = \begin{bmatrix} \lambda_1 & 0 & 0 & 0 \\ 0 & \lambda_2 & 0 & 0 \\ 0 & 0 & \cdot & 0 \\ 0 & 0 & 0 & \lambda_n \end{bmatrix}$

Consider the matrix  $[\mathbf{A} * \mathbf{R}]$ . It is a nxn matrix.

The  $i^{\text{th}}$  column of this matrix is  $\mathbf{A} * \mathbf{r}_i$

That is, the  $i^{\text{th}}$  column of this matrix is  $\lambda_i * \mathbf{r}_i$

Now, consider this matrix  $[\mathbf{R} * \mathbf{L}]$ . It is a nxn matrix.

The  $i^{\text{th}}$  column of this matrix is  $\lambda_i * \mathbf{r}_i$

Thus we conclude that:

$$\mathbf{A} * \mathbf{R} = \mathbf{R} * \mathbf{L}$$

By arranging this equation we get:

$$\mathbf{L} = \mathbf{R}^{-1} * \mathbf{A} * \mathbf{R}$$

Substituting  $\mathbf{L}^T$  for  $\mathbf{R}^{-1}$  we get:

$$\mathbf{L} = \mathbf{L}^T * \mathbf{A} * \mathbf{R}$$

**Theorem A.7:** For any real square matrix  $\mathbf{A}$  (nxn) having a distinct eigenvalue  $\lambda_i$  (regardless of whether or not the rest of eigenvalues are distinct) the sensitivity matrix,  $\mathbf{S}_i$ , associated with this eigenvalue is equal to the product of the left-eigenvector (associated with this eigenvalue), and the transpose of the right-eigenvector (associated with this eigenvalue).

$$\text{That is: } \mathbf{S}_i = \mathbf{l}_i * \mathbf{r}_i^T$$

$$\text{Where: } \mathbf{S}_i = \begin{bmatrix} \frac{\partial \lambda_i}{\partial \mathbf{A}(1,1)} & \cdot & \cdot & \frac{\partial \lambda_i}{\partial \mathbf{A}(1,n)} \\ \cdot & \cdot & \cdot & \cdot \\ \frac{\partial \lambda_i}{\partial \mathbf{A}(n,1)} & \cdot & \cdot & \frac{\partial \lambda_i}{\partial \mathbf{A}(n,n)} \end{bmatrix}$$

### Proof

The matrix equation:  $\mathbf{S}_i = \mathbf{l}_i * \mathbf{r}_i^T$  implies that for any element  $\mathbf{S}_i(x,y)$ :

$$\mathbf{S}_i(x,y) = \mathbf{l}_i(x) * \mathbf{r}_i(y)$$

To simplify the proof, we will assume that  $\mathbf{A}$  is a 2x2 matrix (generalizing to any nxn matrix after that, is not a problem), and we will be focusing on single element in the sensitivity matrix, say  $\mathbf{S}_i(1,1)$  (generalizing to any element in the matrix after that, is straightforward).

Then, our goal will be to proof that:

$$\mathbf{S}_i(1,1) = \mathbf{l}_i(1) * \mathbf{r}_i(1).$$

From corollary A.6, we know that:

$$\mathbf{L} = \mathbf{L}^T * \mathbf{A} * \mathbf{R}$$

By expanding this matrix equation to an ordinary algebraic equation for only one eigenvalue (i.e. one diagonal element in matrix  $\mathbf{L}$ ), we get:

$$\lambda_i = \mathbf{l}_i(1) * \mathbf{r}_i(1) * \mathbf{A}(1,1) + \mathbf{l}_i(1) * \mathbf{r}_i(2) * \mathbf{A}(1,2) + \mathbf{l}_i(2) * \mathbf{r}_i(1) * \mathbf{A}(2,1) + \mathbf{l}_i(2) * \mathbf{r}_i(2) * \mathbf{A}(2,2)$$

By differentiating the above equation with respect to  $\mathbf{A}(1,1)$  we get:

$$\begin{aligned} \frac{\partial \lambda_i}{\partial \mathbf{A}(1,1)} = & \mathbf{l}_i(1) * \mathbf{r}_i(1) + \frac{\partial \mathbf{l}_i(1)}{\partial \mathbf{A}(1,1)} * \mathbf{r}_i(1) * \mathbf{A}(1,1) + \frac{\partial \mathbf{r}_i(1)}{\partial \mathbf{A}(1,1)} * \mathbf{l}_i(1) * \mathbf{A}(1,1) + \\ & \frac{\partial \mathbf{l}_i(1)}{\partial \mathbf{A}(1,1)} * \mathbf{r}_i(2) * \mathbf{A}(1,2) + \frac{\partial \mathbf{r}_i(2)}{\partial \mathbf{A}(1,1)} * \mathbf{l}_i(1) * \mathbf{A}(1,2) + \frac{\partial \mathbf{l}_i(2)}{\partial \mathbf{A}(1,1)} * \mathbf{r}_i(1) * \mathbf{A}(2,1) + \\ & \frac{\partial \mathbf{r}_i(1)}{\partial \mathbf{A}(1,1)} * \mathbf{l}_i(2) * \mathbf{A}(2,1) + \frac{\partial \mathbf{l}_i(2)}{\partial \mathbf{A}(1,1)} * \mathbf{r}_i(2) * \mathbf{A}(2,2) + \frac{\partial \mathbf{r}_i(2)}{\partial \mathbf{A}(1,1)} * \mathbf{l}_i(2) * \mathbf{A}(2,2) \end{aligned}$$

Our aim is to proof that:

$$\frac{\partial \lambda_i}{\partial \mathbf{A}(1,1)} = \mathbf{l}_i(1) * \mathbf{r}_i(1)$$

Thus, our focus will be to proof that:

$$\begin{aligned} 0 = & \frac{\partial \mathbf{l}_i(1)}{\partial \mathbf{A}(1,1)} * \mathbf{r}_i(1) * \mathbf{A}(1,1) + \frac{\partial \mathbf{r}_i(1)}{\partial \mathbf{A}(1,1)} * \mathbf{l}_i(1) * \mathbf{A}(1,1) + \frac{\partial \mathbf{l}_i(1)}{\partial \mathbf{A}(1,1)} * \mathbf{r}_i(2) * \mathbf{A}(1,2) + \\ & \frac{\partial \mathbf{r}_i(2)}{\partial \mathbf{A}(1,1)} * \mathbf{l}_i(1) * \mathbf{A}(1,2) + \frac{\partial \mathbf{l}_i(2)}{\partial \mathbf{A}(1,1)} * \mathbf{r}_i(1) * \mathbf{A}(2,1) + \frac{\partial \mathbf{r}_i(1)}{\partial \mathbf{A}(1,1)} * \mathbf{l}_i(2) * \mathbf{A}(2,1) + \\ & \frac{\partial \mathbf{l}_i(2)}{\partial \mathbf{A}(1,1)} * \mathbf{r}_i(2) * \mathbf{A}(2,2) + \frac{\partial \mathbf{r}_i(2)}{\partial \mathbf{A}(1,1)} * \mathbf{l}_i(2) * \mathbf{A}(2,2) \end{aligned}$$

Recall that the left-eigenvector (that is associated with eigenvalue  $\lambda_i$ ) is:

$$\mathbf{l}_i = [\mathbf{l}_i(1), \mathbf{l}_i(2)]$$

And that right-eigenvector (that is associated with eigenvalue  $\lambda_i$ ) is:

$$\mathbf{r}_i = [\mathbf{r}_i(1), \mathbf{r}_i(2)]$$

From Corollary A.5, we know that the inner product of the left and right eigenvectors is always equal to unity:

$$\{ \mathbf{l}_i(1) * \mathbf{r}_i(1) \} + \{ \mathbf{l}_i(2) * \mathbf{r}_i(2) \} = 1$$

Now, to compute the sensitivity of  $\lambda_i$  to  $\mathbf{A}(1,1)$ , we are going to introduce a small perturbation to the value of  $\mathbf{A}(1,1)$ :

$$\mathbf{A}(1,1)|_{\text{new}} = \mathbf{A}(1,1) + \Delta\mathbf{A}(1,1)$$

The new left-eigenvector will be:

$$\underline{\mathbf{l}}_i |_{\text{new}} = [\underline{\mathbf{l}}_i(1) + \Delta\underline{\mathbf{l}}_i(1), \underline{\mathbf{l}}_i(2) + \Delta\underline{\mathbf{l}}_i(2)]$$

The new right-eigenvector will be:

$$\underline{\mathbf{r}}_i |_{\text{new}} = [\underline{\mathbf{r}}_i(1) + \Delta\underline{\mathbf{r}}_i(1), \underline{\mathbf{r}}_i(2) + \Delta\underline{\mathbf{r}}_i(2)]$$

Also the inner product of the new left and right eigenvectors is equal to unity:

$$\{ \{ \underline{\mathbf{l}}_i(1) + \Delta\underline{\mathbf{l}}_i(1) \} * \{ \underline{\mathbf{r}}_i(1) + \Delta\underline{\mathbf{r}}_i(1) \} \} + \{ \{ \underline{\mathbf{l}}_i(2) + \Delta\underline{\mathbf{l}}_i(2) \} * \{ \underline{\mathbf{r}}_i(2) + \Delta\underline{\mathbf{r}}_i(2) \} \} = 1$$

Hence the inner product of the new left and right eigenvectors is equal to the inner product of the original left and right eigenvectors:

$$\{ \{ \underline{\mathbf{l}}_i(1) + \Delta\underline{\mathbf{l}}_i(1) \} * \{ \underline{\mathbf{r}}_i(1) + \Delta\underline{\mathbf{r}}_i(1) \} \} + \{ \{ \underline{\mathbf{l}}_i(2) + \Delta\underline{\mathbf{l}}_i(2) \} * \{ \underline{\mathbf{r}}_i(2) + \Delta\underline{\mathbf{r}}_i(2) \} \} = \{ \underline{\mathbf{l}}_i(1) * \underline{\mathbf{r}}_i(1) \} + \{ \underline{\mathbf{l}}_i(2) * \underline{\mathbf{r}}_i(2) \}$$

By Simplifying this equation we get:

$$0 = \{ \Delta\underline{\mathbf{l}}_i(1) * \underline{\mathbf{r}}_i(1) \} + \{ \Delta\underline{\mathbf{r}}_i(1) * \underline{\mathbf{l}}_i(1) \} + \{ \Delta\underline{\mathbf{l}}_i(1) * \Delta\underline{\mathbf{r}}_i(1) \} + \{ \Delta\underline{\mathbf{l}}_i(2) * \underline{\mathbf{r}}_i(2) \} + \{ \Delta\underline{\mathbf{r}}_i(2) * \underline{\mathbf{l}}_i(2) \} + \{ \Delta\underline{\mathbf{l}}_i(2) * \Delta\underline{\mathbf{r}}_i(2) \}$$

By dividing both sides by  $\Delta\mathbf{A}(1,1)$ , and by taking the limit as  $\Delta\mathbf{A}(1,1) \rightarrow 0$  we get:

$$\frac{\partial \underline{\mathbf{l}}_i(1)}{\partial \mathbf{A}(1,1)} * \underline{\mathbf{r}}_i(1) + \frac{\partial \underline{\mathbf{l}}_i(2)}{\partial \mathbf{A}(1,1)} \underline{\mathbf{r}}_i(2) + \frac{\partial \underline{\mathbf{r}}_i(1)}{\partial \mathbf{A}(1,1)} \underline{\mathbf{l}}_i(1) + \frac{\partial \underline{\mathbf{r}}_i(2)}{\partial \mathbf{A}(1,1)} \underline{\mathbf{l}}_i(2) = 0$$

Multiplying both sides by  $\lambda_i$  yields:

$$\frac{\partial \underline{\mathbf{l}}(1)}{\partial \mathbf{A}(1,1)} * \lambda_i * \underline{\mathbf{r}}(1) + \frac{\partial \underline{\mathbf{l}}(2)}{\partial \mathbf{A}(1,1)} * \lambda_i * \underline{\mathbf{r}}(2) + \frac{\partial \underline{\mathbf{r}}(1)}{\partial \mathbf{A}(1,1)} * \lambda_i * \underline{\mathbf{l}}(1) + \frac{\partial \underline{\mathbf{r}}(2)}{\partial \mathbf{A}(1,1)} * \lambda_i * \underline{\mathbf{l}}(2) = 0$$

Substituting

$$\begin{aligned} \{ \lambda_i * \underline{\mathbf{r}}(1) \} & \text{ by } \{ \underline{\mathbf{r}}(1) * \mathbf{A}(1,1) + \underline{\mathbf{r}}(2) * \mathbf{A}(1,2) \} \\ \{ \lambda_i * \underline{\mathbf{r}}(2) \} & \text{ by } \{ \underline{\mathbf{r}}(1) * \mathbf{A}(2,1) + \underline{\mathbf{r}}(2) * \mathbf{A}(2,2) \} \\ \{ \lambda_i * \underline{\mathbf{l}}(1) \} & \text{ by: } \{ \underline{\mathbf{l}}(1) * \mathbf{A}(1,1) + \underline{\mathbf{l}}(2) * \mathbf{A}(2,1) \} \\ \{ \lambda_i * \underline{\mathbf{l}}(2) \} & \text{ by: } \{ \underline{\mathbf{l}}(1) * \mathbf{A}(1,2) + \underline{\mathbf{l}}(2) * \mathbf{A}(2,2) \} \end{aligned}$$

And then arranging the equation we get:

$$\begin{aligned} 0 = & \frac{\partial \underline{\mathbf{l}}(1)}{\partial \mathbf{A}(1,1)} * \underline{\mathbf{r}}(1) * \mathbf{A}(1,1) + \frac{\partial \underline{\mathbf{r}}(1)}{\partial \mathbf{A}(1,1)} * \underline{\mathbf{l}}(1) * \mathbf{A}(1,1) + \frac{\partial \underline{\mathbf{l}}(1)}{\partial \mathbf{A}(1,1)} * \underline{\mathbf{r}}(2) * \mathbf{A}(1,2) + \\ & \frac{\partial \underline{\mathbf{r}}(2)}{\partial \mathbf{A}(1,1)} * \underline{\mathbf{l}}(1) * \mathbf{A}(1,2) + \frac{\partial \underline{\mathbf{l}}(2)}{\partial \mathbf{A}(1,1)} * \underline{\mathbf{r}}(1) * \mathbf{A}(2,1) + \frac{\partial \underline{\mathbf{r}}(1)}{\partial \mathbf{A}(1,1)} * \underline{\mathbf{l}}(2) * \mathbf{A}(2,1) + \\ & \frac{\partial \underline{\mathbf{l}}(2)}{\partial \mathbf{A}(1,1)} * \underline{\mathbf{r}}(2) * \mathbf{A}(2,2) + \frac{\partial \underline{\mathbf{r}}(2)}{\partial \mathbf{A}(1,1)} * \underline{\mathbf{l}}(2) * \mathbf{A}(2,2) \end{aligned}$$

**Corollary A.8:** For any real square matrix  $\mathbf{A}$  ( $n \times n$ ) having a distinct eigenvalue  $\lambda_i$  (regardless of whether or not the rest of eigenvalues are distinct):

$$\lambda_i = \sum_{x=1}^n \sum_{y=1}^n \mathbf{S}_i(\mathbf{x}, \mathbf{y}) * \mathbf{A}(\mathbf{x}, \mathbf{y})$$

From corollary A.6, we know that:

$$\mathbf{L} = \mathbf{L}^T * \mathbf{A} * \mathbf{R}$$

By, focusing only on one eigenvalue (i.e. one diagonal element in matrix  $\Lambda$ ), we get:

$$\underline{\mathbf{l}}_i^T * \mathbf{A} * \underline{\mathbf{r}}_i = \lambda_i$$

$$\because \mathbf{S}_i(\mathbf{x}, \mathbf{y}) = \underline{\mathbf{l}}_i(\mathbf{x}) * \underline{\mathbf{r}}_i(\mathbf{y})$$

$$\therefore \mathbf{l}_i^T * \mathbf{A} * \mathbf{r}_i = \sum_{x=1}^n \sum_{y=1}^n \mathbf{S}_i(\mathbf{x}, \mathbf{y}) * \mathbf{A}(\mathbf{x}, \mathbf{y})$$

$$\therefore \lambda_i = \sum_{x=1}^n \sum_{y=1}^n \mathbf{S}_i(\mathbf{x}, \mathbf{y}) * \mathbf{A}(\mathbf{x}, \mathbf{y})$$

**Corollary A.9:** For any real square matrix A (nxn) having a distinct eigenvalue  $\lambda_i$  (regardless of whether or not the rest of eigenvalues are distinct):

$$1 = \sum_{x=1}^n \sum_{y=1}^n \mathbf{E}_i(\mathbf{x}, \mathbf{y})$$

Where  $\mathbf{E}_i(\mathbf{x}, \mathbf{y})$  is the  $\lambda_i$  elasticity to element  $\mathbf{A}(\mathbf{x}, \mathbf{y})$ .

$\mathbf{E}_i(\mathbf{x}, \mathbf{y})$  is defined as

$$\mathbf{E}_i(\mathbf{x}, \mathbf{y}) = \frac{\mathbf{S}_i(\mathbf{x}, \mathbf{y}) * \mathbf{A}(\mathbf{x}, \mathbf{y})}{\lambda_i}$$

$$\therefore \sum_{x=1}^n \sum_{y=1}^n \mathbf{E}_i(\mathbf{x}, \mathbf{y}) = \sum_{x=1}^n \sum_{y=1}^n \frac{\mathbf{S}_i(\mathbf{x}, \mathbf{y}) * \mathbf{A}(\mathbf{x}, \mathbf{y})}{\lambda_i} = \frac{1}{\lambda_i} * \sum_{x=1}^n \sum_{y=1}^n \mathbf{S}_i(\mathbf{x}, \mathbf{y}) * \mathbf{A}(\mathbf{x}, \mathbf{y}) = \frac{\lambda_i}{\lambda_i} = 1$$

**Theorem A.10:** For any real square matrix A (nxn) having a distinct eigenvalue  $\lambda_i$  (regardless of whether or not the rest of eigenvalues are distinct):

$$\sum_{y=1}^n \mathbf{E}_i(\mathbf{q}, \mathbf{y}) = \sum_{y=1}^n \mathbf{E}_i(\mathbf{y}, \mathbf{q}) \quad (\mathbf{q} = 1 \dots n)$$

This theorem is a mathematical description of the system dynamics phenomenon: “the sum of the eigenvalue elasticities of all links coming into a variable equals the sum of the eigenvalue elasticities of all links leaving the variable”. This phenomenon was originally observed by N. Forrester.

### Proof

Starting from the definition of the left-eigenvector:

$$\underline{\mathbf{l}}_i^T * \mathbf{A} = \lambda_i * \underline{\mathbf{l}}_i^T$$

Pre-multiplying both sides by  $\underline{\mathbf{r}}_i$  yields:

$$\underline{\mathbf{r}}_i * \underline{\mathbf{l}}_i^T * \mathbf{A} = \lambda_i * \underline{\mathbf{r}}_i * \underline{\mathbf{l}}_i^T$$

Recalling that the transpose of a product equals the product of the transposed factors, taken in reverse order; then transposing both sides of the equation yields:

$$\mathbf{A}^T * \underline{\mathbf{l}}_i * \underline{\mathbf{r}}_i^T = \lambda_i * \underline{\mathbf{l}}_i * \underline{\mathbf{r}}_i^T$$

Substituting  $\underline{\mathbf{l}}_i * \underline{\mathbf{r}}_i^T$  by  $\mathbf{S}_i$  yields:

$$\mathbf{A}^T * \mathbf{S}_i = \lambda_i * \mathbf{S}_i \dots(1)$$

Now, from the definition of the right-eigenvector:

$$\mathbf{A} * \underline{\mathbf{r}}_i = \lambda_i * \underline{\mathbf{r}}_i$$

Getting the transpose of both sides yields:

$$\underline{\mathbf{r}}_i^T * \mathbf{A}^T = \lambda_i * \underline{\mathbf{r}}_i^T$$

Pre-multiplying both sides by  $\underline{\mathbf{l}}_i$  yields:

$$\underline{\mathbf{l}}_i * \underline{\mathbf{r}}_i^T * \mathbf{A}^T = \lambda_i * \underline{\mathbf{l}}_i * \underline{\mathbf{r}}_i^T$$

Substituting  $\underline{\mathbf{l}}_i * \underline{\mathbf{r}}_i^T$  by  $\mathbf{S}_i$  yields:

$$\mathbf{S}_i * \mathbf{A}^T = \lambda_i * \mathbf{S}_i \dots(2)$$

From equations (1) and (2) we can conclude that:

$$\mathbf{A}^T * \mathbf{S}_i = \mathbf{S}_i * \mathbf{A}^T$$

$$\therefore \sum_{y=1}^n \mathbf{S}_i(\mathbf{q}, \mathbf{y}) * \mathbf{A}(\mathbf{q}, \mathbf{y}) = \sum_{y=1}^n \mathbf{S}_i(\mathbf{y}, \mathbf{q}) * \mathbf{A}(\mathbf{y}, \mathbf{q}) \quad (\mathbf{q}=1\dots n)$$

Dividing both sides by  $\lambda_i$  yields:

$$\sum_{y=1}^n \mathbf{E}_i(\mathbf{q}, \mathbf{y}) = \sum_{y=1}^n \mathbf{E}_i(\mathbf{y}, \mathbf{q}) \quad (\mathbf{q} = 1\dots n)$$

### **A.11 Developing a mathematical function that relates the eigenvalue to the gains of feedback loops.**

The idea is to develop a function that has the eigenvalue as the dependent parameter, and the gains of loops as the independent parameters. Such a function can aid in illustrating the key role that the loop elasticities play in determining the significance of a loop to an eigenvalue.

We are going to develop this function step by step with the help of the yeast cells generation example (section 5). We will start by recapping the gain information of the model.

In the mini model form, the link gains are G11\_inflow (effect of cells on births), G11\_outflow (effect of cells on deaths), G12\_inflow (effect of alcohol on births), and G12\_outflow (effect of alcohol on deaths); in addition to G21 (effect of cells on alcohol generation). Recall that G22 (effect of alcohol on alcohol generation) is always zero, thus we can easily ignore it. The mini-model link elasticities associated with those gains are E11\_inflow, E11\_outflow, E12\_inflow, E12\_outflow, and E21 respectively. We are going to use the shortcut notations  $e_1$ ,  $e_2$ ,  $e_3$ ,  $e_4$ , and  $e_5$  for E11\_inflow, E11\_outflow, E12\_inflow, E12\_outflow, and E21 respectively; and the

shortcut notations  $g_1, g_2, g_3, g_4,$  and  $g_5,$  for  $G11\_inflow, G11\_outflow, G12\_inflow, G12\_outflow,$  and  $G21$  respectively.

This model has four loops (mini model form): L1 – L4. L1 is the cells-births-cells loop. L2 is the cells-deaths-cells loop. L3 is the cells-alcohol (through the AlcoholGeneration rate) – births-cells loop. L4 is the cells-alcohol-deaths-cells loop. The gain of L1,  $g(L1),$  is equal to  $g_1.$  The gain of L2,  $g(L2),$  is equal to  $(-1 * g_2).$  The gain of L3,  $g(L3),$  is equal to  $(g_3 * g_5).$  The gain of L4,  $g(L4),$  is equal to  $(-1 * g_4 * g_5).$

Now, we will start developing our desired function; our point of departure is the fact that sum of all compact link elasticities equals unity (recall Corollary A.9). As the elasticity of a compact link equals the sum of elasticities of mini-model links that constitute that compact link, then one can directly infer that the sum of all mini-model link elasticities must also equal unity.

Thus:

$$1 = \sum_{x=1}^5 \mathbf{e}_x$$

Multiplying both sides by the eigenvalue  $\lambda_i$  yields:

$$\lambda_i = \sum_{x=1}^5 \frac{\partial \lambda_i}{\partial \mathbf{g}_x} \mathbf{g}_x \quad (\text{Recall the definition of elasticity})$$

The above equation is a homogenous linear partial differential equation of the first order.

Its solution is:

$$\lambda_i = \mathbf{c} \prod_{x=1}^5 \mathbf{g}_x^{\mathbf{e}_x} = \mathbf{c} \cdot \mathbf{g}_1^{\mathbf{e}_1} \cdot \mathbf{g}_2^{\mathbf{e}_2} \cdot \mathbf{g}_3^{\mathbf{e}_3} \cdot \mathbf{g}_4^{\mathbf{e}_4} \cdot \mathbf{g}_5^{\mathbf{e}_5}$$

Where  $\mathbf{c}$  is a constant.

This solution is only valid in the infinitesimal circular neighborhood of the operating point, as the elasticities are treated as constants in the solution.

Now, recall that the elasticity of a link equals to the sum of elasticities of all the loops that contain the link. Then from the causal-loop diagram of the model (fig. 24), one can easily conclude that:

$$\begin{aligned} e_1 &= e(L1) \\ e_2 &= e(L2) \\ e_3 &= e(L3) \\ e_4 &= e(L4) \\ e_5 &= e(L3) + e(L4) \end{aligned}$$

Substituting the link elasticities by the loop elasticities yields:

$$\lambda_i = \mathbf{c} \cdot \mathbf{g}_1^{e(L1)} \cdot \mathbf{g}_2^{e(L2)} \cdot \mathbf{g}_3^{e(L3)} \cdot \mathbf{g}_4^{e(L4)} \cdot \mathbf{g}_5^{\{e(L3)+e(L4)\}}$$

By arranging this equation we get:

$$\lambda_i = \mathbf{c} \cdot \mathbf{g}_1^{e(L1)} \cdot \mathbf{g}_2^{e(L2)} \cdot (\mathbf{g}_3 \cdot \mathbf{g}_5)^{e(L3)} \cdot (\mathbf{g}_4 \cdot \mathbf{g}_5)^{e(L4)}$$

$$\lambda_i = \mathbf{c}^* \cdot \mathbf{g}_1^{e(L1)} \cdot (-\mathbf{g}_2)^{e(L2)} \cdot (\mathbf{g}_3 \cdot \mathbf{g}_5)^{e(L3)} \cdot (-\mathbf{g}_4 \cdot \mathbf{g}_5)^{e(L4)}$$

$$\mathbf{c}^* = \frac{\mathbf{c}}{(-1)^{e(L2)+e(L4)}}$$

Recall that  $g(L1)$  is equal to  $g_1$ ,  $g(L2)$  is equal to  $(-1 * g_2)$ ,  $g(L3)$  is equal to  $(g_3 * g_5)$ , and  $g(L4)$  is equal to  $(-1 * g_4 * g_5)$ ; then:

$$\lambda_i = \mathbf{c}^* \prod_{x=1}^5 \mathbf{g}(Lx)^{e(Lx)}$$

Where  $g(Lx)$  is the gain of loop  $x$ ; and  $e(Lx)$  is the elasticity of loop  $x$ .

From the above formula, one can directly infer that the only parameter that determines the significance of a loop to an eigenvalue, is the elasticity associated with that loop.

Note: the above formula can be generalized to any model; i.e. it represents a “*universal relationship*” between any eigenvalue and the feedback loops in a model.