Appendix A.

A.1. Mathematical foundation of the eigenvalue elasticity analysis

The causal structure of a linear (or linearized) model can be represented as a gain matrix. Let **G** be the gain matrix of an nth order system dynamics model. Then **G** will be an n×n square matrix. The entries of the gain matrix are the net gains of compact links between the state variables (stocks) of the model to be analyzed. Forrester (1982) proposed using the eigenvalues of the gain matrix as a bridge between model structure and behavior. Then the model can be represented in matrix form:

$$\dot{\mathbf{x}} = \mathbf{G}\mathbf{x} \qquad \text{(A.1)}$$

where $\dot{\mathbf{x}}$ is the vector of the net rate (flow) of state variables and $\mathbf{x}$ is the state variables vector.

While the eigenvalues of the gain matrix **G** stand for the elemental behavior modes of the system the eigenvalue elasticities reflect the influence of different parts of model structure on these behavior modes. Saleh (2002) showed how overall model behavior could be partitioned into its components on the eigenspace. The departure point in this approach is differentiating both sides of Eq. A.1, which leads to the observation that the behavior pattern of any state variable is determined by its time derivative (slope) and its second time derivative (curvature).

$$\mathbf{c} = \mathbf{G}\mathbf{s} \qquad \text{(A.2)}$$

where **c** is the curvature (the second derivative) of **x** and **s** is the net rate (slope) (the first derivative) of **x**.

The $\dot{\mathbf{x}}$ in Eq. A.1 constitutes the slope vector of the state variables. In the eigenspace, the slope vector can be expressed as a linear combination of the right eigenvectors of the gain matrix, **G**.

$$\mathbf{s} = \sum_{i=1}^{n} \alpha_i \mathbf{r}_i \qquad \text{(A.3)}$$

In the eigenspace, α_i are the new components of the slope vector, **s**.

Differentiating Eq. A.3 gives:

$$\mathbf{c} = \sum_{i=1}^{n} \dot{\alpha}_i \mathbf{r}_i \qquad \text{(A.4)}$$

Now, in the eigenspace the α_i are the new components of the curvature vector.

Substituting Eq. A.3 into Eq. A.4 and utilizing the fact that $\mathbf{G}\mathbf{r}_i = \lambda_i \mathbf{r}_i$ results in

$$\mathbf{c} = \sum_{i=1}^{n} \alpha_i \lambda_i \mathbf{r}_i \qquad \text{(A.5)}$$

Then along a particular coordinate (spanned by a right eigenvector) the dynamics that unfolds can be described by,

$$\dot{\alpha}_i = \lambda_i \alpha_i \qquad i = 1 \ldots n \qquad \text{(A.6)}$$

The solution of Eq. A.6 is then,

$$\alpha_i = \alpha_i^0 e^{\lambda_i (t-t_o)} \qquad i = 1\ldots\text{n} \qquad \text{(A.7)}$$

where t_o is the initial time, α_i^0 is the initial value of α_i at t_o.

Eqs. A.2 through A.7 shows that the only factor determining the dynamics along a particular coordinate on the eigenspace is the eigenvalue associated with that coordinate.

Finally, substituting Eq. A.7 into Eq. A.3 yields the time trajectory of the slope:

$$\mathbf{s} = \sum_{i=1}^{\text{n}} \alpha_i^0 e^{\lambda_i (t-t_o)} \mathbf{r}_i \qquad \text{(A.8)}$$

Thus, the slope trajectory is decomposed into several behavior modes, each expressed by an eigenvalue and its associated right eigenvector.

The elasticity of an eigenvalue with respect to a variable measures the percentage change in the eigenvalue for a given percentage change in that variable. The sensitivity $S_{pq,i}$ of an eigenvalue λ_i with respect to a variable pq is given by the partial derivative of that eigenvalue with respect to the variable (Eq. A.9). The eigenvalue elasticity is then defined as the sensitivity of the eigenvalue with respect to the variable normalized for the size of the variable and the size of the eigenvalue (Eq. A.10). As shown in the equation, eigenvalue elasticity can be computed using left and right eigenvectors and the partial derivative of the linear system matrix, **G**, with respect to gain g_{pq}. The partial derivative of **G** with respect to a variable can be found by calculating the matrix before and after a small change in the parameter. Another way is to derive the expressions for the partial derivative of **G** and using them to compute elasticity values.

$$S_{pq,i} = \frac{\partial \lambda_i}{\partial g_{pq}} \qquad \text{(A.9)}$$

$$e_{pq,i} = S_{pq,i} \frac{g_{pq}}{\lambda_i} = \mathbf{l}_i^{'} \frac{\partial \mathbf{G}}{\partial g_{pq}} \mathbf{r}_i * \frac{g_{pq}}{\lambda_i} \qquad \text{(A.10)}$$

where
$e_{pq,i}$ ≡ the elasticity of eigenvalue i, λ_i to compact gain link pq, g_{pq}
λ_i ≡ eigenvalue i
g_{pq} ≡ gain of the link from p to q
$\mathbf{l}_i^{'}$ ≡ transpose of the i^{th} left eigenvector (1×n vector)
G ≡ linear(ized) system matrix (n×n)
$\mathbf{r}_i$ ≡ i^{th} right eigenvector (n×1).

A more useful formulation of eigenvalue elasticity is, however, possible without having to calculate the partial derivative of the gain matrix, **G** (Eq. A.11).

$$e_{pq,i} = \mathbf{l}_i(p) * \mathbf{r}_i(q) * \frac{g_{pq}}{\lambda_i} \qquad \text{(A.11)}$$

where
$\mathbf{l}_i(p)$ ≡ the p^{th} element of the i^{th} left eigenvector (1×n vector)
$\mathbf{r}_i(q)$ ≡ the q^{th} element of the i^{th} right eigenvector (1×n vector)

Eq. A.11 comes from the relation that the sensitivity matrix $\mathbf{S}_i$ of the eigenvalue λ_i is equal to the product of the i^{th} left eigenvector and the i^{th} right eigenvector of the gain matrix, **G** (Eq. A.12). The theorem behind this relation and its proof is given in Saleh (2002).

$$\mathbf{S}_i = \mathbf{l}_i \cdot \mathbf{r}_i' \qquad \text{(A.12)}$$

The compact link elasticities are not much of use by themselves in revealing the dominant structure. They are, by definition, compact giving no hint of the detailed model structure. One needs to make use of the elasticities to individual causal links in the model to bring in the structural detail into the analysis. It turns out that it is possible to obtain causal link elasticities using pathway gains and compact link elasticities. The general expression for causal link elasticities is

$$e_{l_j,i} = \sum_{p=1}^{\mathrm{P}} \sum_{q=1}^{\mathrm{Q}} e_{pq,i} \frac{\sum_{s=1}^{\mathrm{S}} g_{pq,s}}{\sum_{r=1}^{\mathrm{R}} g_{pq,r}} \qquad \text{(A.13)}$$

where $e_{l_j,i}$ ≡ the elasticity of eigenvalue i, λ_i to causal link l_j

$e_{pq,i}$ ≡ the elasticity of eigenvalue i, λ_i to compact gain link pq, g_{pq}

$g_{pq,r(s)}$ ≡ gain of pathway $r(s)$ from state variable p to q

S ≡ the number of pathways causal link l_j is a part of

R ≡ the total number of pathways between state variables p and q

A.2. Directed Cycle Matrix

A directed cycle matrix, in system dynamics context, reflects the membership of causal links (except the ones that involve constants and flow-to-stock links) to the feedback loops of a model (Kampmann 1996). If only the loops in an "independent loop set" as in Kampmann or the "shortest independent loop set" as in Oliva (2004) is used then the columns of the directed cycle matrix are linearly independent. Thus one can show the relation of links with the loops as in Eq. A.14.

$$\begin{bmatrix} l_1 \\ . \\ l_j \\ . \\ l_{\mathrm{J}} \end{bmatrix} = \mathbf{D} \begin{bmatrix} lp_1 \\ . \\ lp_k \\ . \\ lp_{\mathrm{K}} \end{bmatrix} \qquad \text{(A.14)}$$

where l_j represents a causal link, lp_k a loop of the model under study and **D** is the directed cycle matrix:

$$\mathbf{D} = \begin{bmatrix} d_{11} & . & . & d_{1\mathrm{K}} \\ . & d_{jk} & . & . \\ . & . & . & . \\ d_{\mathrm{J}1} & . & . & d_{\mathrm{JK}} \end{bmatrix} \qquad \text{(A.15)}$$

where d_{jk} equals 1 if link j is an element of loop k and 0 otherwise.

Finally, the set of equations that relates the link elasticities to loop elasticities is

$$\begin{bmatrix} e_{l_1} \\ \cdot \\ e_{l_j} \\ \cdot \\ e_{l_J} \end{bmatrix} = \mathbf{D} \begin{bmatrix} e_{lp_1} \\ \cdot \\ e_{lp_k} \\ \cdot \\ e_{lp_K} \end{bmatrix} \tag{A.16}$$

where e_{l_i} is the elasticity of link l_j, e_{lp_k} is the elasticity of loop lp_k and **D** is the directed cycle matrix as shown in Eq. A.15.

The number of links is almost always larger than the number of loops in system dynamics models. This means that the directed cycle matrix will be overdetermined. In fact, it is shown by Kampmann that the number of independent loops is equal to (total number of links – total number of nodes + 1) and in a typical system dynamics model the number of links far exceeds the number of nodes. Refer to Kampmann (1996) for a proof of this relationship. The linear independence of columns in a directed cycle matrix, however, allows one to obtain a unique solution for loop elasticities even when the matrix is overdetermined.

Appendix B.
Pseudo-codes for eigenvalue elasticity analysis implementation
B.1. Pseudo-code for the main function

The main function serves as an intermediary between Vensim and MATLAB. The pseudo-code of the last MATLAB function called in the main function, tofile.m is not given below. It simply writes the results of the analysis to files.

```
ep ← engOpen(NULL)                                                  start MATLAB connection
dt ← TIMESTEP                                                       read timestep into MATLAB
ord ← ORDER                                                         read order of model into MATLAB
for time = 0: FINAL_TIME*(1/TIMESTEP)                               for each timestep throughout simulation run
   timein ← time * TIMESTEP                                            actual time in simulation
   for i = 1: total number of compact gains and pathway gains          for each compact and pathway gain
     vensim_get_sens_at_time("data.vdf", "gain(i)", "Time", timein, gain(i), 1)   read simulated gain data from Vensim;
                                                                            populate gain matrix
   end                                                                 end
   for i = 1: total number of state variables (stocks)                 for each stock
     vensim_get_sens_at_time("data.vdf", "slope(i)", "Time", timein, slope(i), 1)   read simulated slope data from Vensim;
                                                                            populate slope matrix
   end                                                                 end
   t ← timein                                                          read current time into MATLAB
   g ← gain                                                            read gain matrix into MATLAB
   slp ← slope                                                         read slope matrix into MATLAB
   call Evcont.m                                                       call MATLAB function  Evcont.m
   call Elast.m                                                          ″      ″       ″    Elast.m
   call LoopElast.m                                                      ″      ″       ″    LoopElast.m
   call tofile.m                                                         ″      ″       ″    tofile.m
end                                                                 end time loop
engClose(ep)                                                        close MATLAB connection
```

B.2. Pseudo-code of the function Evcont.m

The function calculates the eigenvalues, eigenvectors, initial alphas and the contribution of each eigenvalue on the behavior of the selected state variable. Its pseudo-code tailored for a third-order system is given below.

```
function Evcont.m

[v,d] ← eig(G)                                        calculate eigenvalues and (right) eigenvectors
vinv ← inv(v)                                         take inverse of eigenvector matrix
alp ← vinv * slp                                      calculate initial alphas
voi ← {1, 2, or 3}                                    select variable of interest
if (imag(d) == 0)                                     if all eigenvalues are real
   s1(1) ← alp(1) * v(voi,1) * exp(d(1,1) * (0-0))       calculate initial value of slope component along
                                                         eigenvector associated with first eigenvalue
   s2(1) ← alp(2) * v(voi,2) * exp(d(2,2) * (0-0))       calculate initial value of slope component along
                                                         eigenvector associated with second eigenvalue
   s3(1) ← alp(3) * v(voi,3) * exp(d(3,3) * (0-0))       calculate initial value of slope component along
                                                         eigenvector associated with third eigenvalue
   s1(2) ← alp(1) * v(voi,1) * exp(d(1,1) * (dt-0))      calculate final value of slope component along
                                                         eigenvector associated with first eigenvalue
   s2(2) ← alp(2) * v(voi,2) * exp(d(2,2) * (dt-0))      calculate final value of slope component along
                                                         eigenvector associated with second eigenvalue
   s3(2) ← alp(3) * v(voi,3) * exp(d(3,3) * (dt-0))      calculate final value of slope component along
                                                         eigenvector associated with third eigenvalue
   diff_s(1) ← s1(2) - s1(1)                             calculate change along   first   slope component
   diff_s(2) ← s2(2) - s2(1)                                 ″       ″      ″    second     ″       ″
   diff_s(3) ← s3(2) - s3(1)                                 ″       ″      ″    third      ″       ″
   absdiff ← abs(diff_s(1)) + abs(diff_s(2)) + abs(diff_s(3))   calculate sum of magnitudes of changes
   for i = 1: 3                                          for each eigenvalue
```

```
      cont(i) ← diff_s(i) / absdiff                                  calculate its contribution
    end                                                              end
else                                                             if there is complex (eigenvalue) pair
    beta1 ← real(alp(1))*real(v(voi,2))-abs(imag(alp(1)))*abs(imag(v(voi,2)))      calculate the coefficient of cosine component
    gamma1 ← real(alp(1))*abs(imag(v(voi,2)))+ abs(imag(alp(1)))*real(v(voi,2))     ″   ″   ″   ″ sine   ″
    sr(1) ← real(alp(3))*real(v(voi,3))*exp(real(d(3,3))*(0-0))      calculate initial value of slope component along
                                                                     eigenvector associated with real eigenvalue
    sc(1) ← 2*exp(real(d(1,1))*(0-0))*(beta1*cos(abs(imag(d(1,1)))*(0-0)*((2*pi)/360))   calculate initial value of slope component along
              -gamma1*sin(abs(imag(d(1,1)))*(0-0)*((2*pi)/360)))      eigenvector associated with complex pair
    sr(2) ← real(alp(3))*real(v(voi,3))*exp(real(d(3,3))*(dt-0))     calculate final value of slope component along
                                                                     eigenvector associated with real eigenvalue
    sc(2) ← 2*exp(real(d(1,1))*(dt-0))*(beta1*cos(abs(imag(d(1,1)))*(dt-0)*((2*pi)/360))   calculate final value of slope component along
              -gamma1*sin(abs(imag(d(1,1)))*(dt-0)*((2*pi)/360)))     eigenvector associated with complex pair
    diff_real ← sr(2) - sr(1)                                        calculate change due to real eigenvalue
    diff_cmp ← sc(2) - sc(1)                                           ″   ″   ″   ″ complex pair
    cont_real ← diff_sreal / (abs(diff_real) + abs(diff_cmp))        calculate contribution of real eigenvalue
    cont_cmp ← diff_scmp / (abs(diff_real) + abs(diff_cmp))            ″   ″   ″ complex pair
end                                                              end
```

B.3. Pseudo-code of the function Elast.m

The function calculates the compact link elasticities of each eigenvalue regardless of the order of the model under study.

function Elast.m

```
l = vinv.'                                              get left eigenvectors
for ev = 1: ord                                         for each eigenvalue
    for i=1: ord                                            for each row
      for j=1: ord                                              for each column
        E(i,j,ev) ← l(i,ev) * vinv(j,ev) * G(i,j) / d(ev,ev)        calculate the compact link elasticity values
      end                                                       end
    end                                                     end
end                                                     end
```

B.4. Pseudo-code of the function LoopElast.m

The function calculates the elasticities of each eigenvalue to the links and then to the loops in the Shortest Independent Loop Set (SILS). It is modified for the analysis of the simple long wave model.

function LoopElast.m

The for loop below calculates, for each eigenvalue, the elasticities to causal links (except the ones that involve constants and flow-to-stock links) based on the compact link elasticities

```
for i = 1: 3
    el(1,i) ← E(1,3,i) * (g13p1 / G(1,3)) + E(2,3,i) * (g23p1 / G(2,3))
    el(2,i) ← E(3,3,i) * ((g33p1 + g33p3 + g33p4) / G(3,3)) + E(2,3,i) * ((g23p2 + g23p3 + g23p4) / G(2,3)) + E(1,3,i) * (g13p2 / G(1,3))
    el(3,i) ← E(2,3,i) * (g23p5 / G(2,3)) + E(3,3,i) * (g33p2 / G(3,3))
    el(4,i) ← E(1,1,i) * (g11p2 / G(1,1)) + E(2,1,i) * ((g21p2 + g21p8) / G(2,1)) + E(3,1,i) * ((g31p2 + g31p8) / G(3,1))
    el(5,i) ← E(1,1,i) * (g11p1 / G(1,1)) + E(2,1,i) * ((g21p1 + g21p7) / G(2,1)) + E(3,1,i) * ((g31p1 + g31p7) / G(3,1))
    el(6,i) ← E(1,1,i) * (g11p2 / G(1,1)) + E(1,3,i) * (g13p2 / G(1,3)) + E(2,1,i) * ((g21p2 + g21p8) / G(2,1))
      + E(2,3,i) * ((g23p2 + g23p3) / G(2,3)) + E(3,1,i) * ((g31p2 + g31p8) / G(3,1)) + E(3,3,i) * ((g33p1 + g33p3) / G(3,3))
    el(7,i) ← E(1,1,i) * ((g11p1 + g11p2) / G(1,1)) + E(2,1,i) * ((g21p1 + g21p2 + g21p7 + g21p8) / G(2,1))
      + E(3,1,i) * ((g31p1 + g31p2 + g31p7 + g31p8) / G(3,1))
    el(8,i) ← E(2,1,i) * (g21p4 / G(2,1)) + E(3,1,i) * (g31p4 / G(3,1))
    el(9,i) ← E(1,1,i) * (g11p3 / G(1,1)) + E(2,1,i) * ((g21p3 + g21p5 + g21p6 + g21p9) / G(2,1))
      + E(3,1,i) * ((g31p3 + g31p5 + g31p6 + g31p9) / G(3,1))
    el(10,i) ← E(2,1,i) * (g21p4 / G(2,1)) + E(2,3,i) * (g23p4 / G(2,3)) + E(3,1,i) * (g31p4 /G(3,1)) + E(3,3,i) * (g33p4 / G(3,3))
    el(11,i) ← E(3,1,i) * ((g31p3 + g31p4 + g31p5 + g31p6 + g31p7 + g31p8 + g31p9) / G(3,1)) + E(3,2,i)
      + E(3,3,i) * ((g33p2 + g33p3 + g33p4) / G(3,3))
    el(12,i) ← E(2,1,i) * (g21p3 / G(2,1)) + E(3,1,i) * (g31p3 / G(3,1))
```

el(13,*i*) ← **E**(2,1,*i*) * (g21p5 / **G**(2,1)) + **E**(3,1,*i*) * (g31p5 / **G**(3,1))
el(14,*i*) ← **E**(2,1,*i*) * (g21p6 / **G**(2,1)) + **E**(3,1,*i*) * (g31p6 / **G**(3,1))
el(15,*i*) ← **E**(2,1,*i*) * (g21p9 / **G**(2,1)) + **E**(3,1,*i*) * (g31p9 / **G**(3,1))
el(16,*i*) ← **E**(2,3,*i*) * (g23p4 / **G**(2,3)) + **E**(3,3,*i*) * (g33p4 / **G**(3,3))
el(17,*i*) ← **E**(2,1,*i*) * ((g21p4 + g21p5 + g21p6 + g21p7 + g21p8) / **G**(2,1)) + **E**(2,2,*i*) * (g22p1 / **G**(2,2)) + **E**(2,3,*i*) * ((g23p3 + g23p4 + g23p5) / **G**(2,3)) + **E**(3,1,*i*) * ((g31p4 + g31p5 + g31p6 + g31p7 + g31p8) / **G**(3,1)) + **E**(3,2,*i*) + **E**(3,3,*i*) * ((g33p2 + g33p3 + g33p4) / **G**(3,3))
el(18,*i*) ← **E**(1,3,*i*) * (g13p2 / **G**(1,3)) + **E**(2,3,*i*) * ((g23p2 + g23p3) / **G**(2,3)) + **E**(3,3,*i*) * ((g33p1 + g33p3) / **G**(3,3))
el(19,*i*) ← **E**(2,3,*i*) * (g23p4 / **G**(2,3)) + **E**(3,3,*i*) * (g33p4 / **G**(3,3))
el(20,*i*) ← **E**(2,1,*i*) * ((g21p6 + g21p7 + g21p8) / **G**(2,1)) + **E**(3,1,*i*) * ((g31p6 + g31p7 + g31p8) / **G**(3,1)) + **E**(2,3,*i*) * ((g23p3 + g23p5) / **G**(2,3)) + **E**(3,3,*i*) * ((g33p2 + g33p3) / **G**(3,3))
el(21,*i*) ← **E**(1,1,*i*) * ((g11p1 + g11p2) / **G**(1,1)) + **E**(2,1,*i*) * ((g21p1 + g21p2) / **G**(2,1)) + **E**(1,3,*i*) * (g13p2 / **G**(1,3)) + **E**(2,3,*i*) * (g23p2 / **G**(2,3))
el(22,*i*) ← **E**(2,1,*i*) * ((g21p7 + g21p8) / **G**(2,1)) + **E**(3,1,*i*) * ((g31p7 + g31p8) / **G**(3,1)) + **E**(2,3,*i*) * (g23p3 / **G**(2,3)) + **E**(3,3,*i*) * (g33p3 / **G**(3,3))
el(23,*i*) ← **E**(2,1,*i*) * ((g21p4 + g21p5 + g21p6 + g21p7 + g21p8 + g21p9) / **G**(2,1)) + **E**(2,2,*i*) * (g22p1 / **G**(2,2)) + **E**(2,3,*i*) * ((g23p3 + g23p4 + g23p5) / **G**(2,3)) + **E**(3,1,*i*) * ((g31p4 + g31p5 + g31p6 + g31p7 + g31p8 + g31p9) / **G**(3,1)) + **E**(3,2,*i*) + **E**(3,3,*i*) * ((g33p2 + g33p3 + g33p4) / **G**(3,3))
el(24,*i*) ← **E**(1,2,*i*) + **E**(2,2,*i*) * (g22p2 / **G**(2,2))
el(25,*i*) ← **E**(2,2,*i*) * (g22p1 / **G**(2,2)) + **E**(3,2,*i*)
el(26,*i*) ← **E**(2,1,*i*) * ((g21p6 + g21p7 + g21p8) / **G**(2,1)) + **E**(2,2,*i*) * (g22p1 / **G**(2,2)) + **E**(2,3,*i*) * ((g23p3 + g23p5) / **G**(2,3)) + **E**(3,1,*i*) * ((g31p6 + g31p7 + g31p8) / **G**(3,1)) + **E**(3,2,*i*) + **E**(3,3,*i*) * ((g33p2 + g33p3) / **G**(3,3))
end

dcm ← — create directed cycle matrix

```
[0  0  0  0  0  1  0  1  0  0  0  0  0  0  0  0
 0  0  0  1  0  0  0  0  0  0  0  0  0  1  0  0
 0  0  0  0  0  0  0  0  0  0  1  0  0  0  0  0
 0  0  0  0  0  0  0  0  1  0  0  0  0  0  0  0
 0  0  1  0  0  0  0  1  0  0  0  0  0  0  0  1
 0  0  0  1  0  0  0  0  1  0  0  0  0  0  0  0
 0  0  1  0  0  0  0  1  1  0  0  0  0  0  0  1
 0  0  0  0  0  0  0  0  0  0  0  0  1  0  0  0
 1  0  0  0  1  1  0  0  0  1  0  1  0  0  1  0
 0  0  0  0  0  0  0  0  0  0  0  0  1  1  0  0
 0  0  0  0  0  1  0  0  0  0  1  0  0  1  0  0
 0  0  0  0  1  1  0  0  0  0  0  0  0  0  0  0
 0  0  0  0  0  0  0  0  0  0  0  1  0  0  0  0
 0  0  0  0  0  0  0  0  0  0  0  0  0  0  1  0
 0  0  0  0  0  0  0  0  0  1  0  0  0  0  0  0
 0  0  0  0  0  0  0  0  0  0  0  0  0  1  0  0
 0  0  0  0  0  0  1  0  0  0  1  1  1  1  1  1
 0  0  0  1  0  0  0  0  0  0  0  0  0  0  0  0
 0  0  0  0  0  0  0  0  0  0  0  0  0  1  0  0
 0  0  0  0  0  0  0  0  0  0  1  0  0  0  1  1
 0  0  1  0  0  0  0  0  1  0  0  0  0  0  0  0
 0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  1
 0  0  0  0  0  0  1  0  0  1  1  1  1  1  1  1
 0  1  0  0  1  0  0  0  0  1  0  1  1  0  1  1
 0  0  0  0  0  0  1  0  0  0  0  0  0  0  0  0
 0  0  0  0  0  0  1  0  0  0  1  0  0  0  1  1]
```

for *i*=1:3	for each eigenvalue
elps(:,*i*) ← **dcm** \ **el**(:,*i*)	calculate elasticities to loops in SILS
end	end
for *i* = 1: 16	for each loop in SILS
overall_elps(*i*) ← **cont**(1) * **elps**(*i*,1) + **cont**(2) * **elps**(*i*,2) + **cont**(3) * **elps**(*i*,3)	calculate overall elasticity
abs_elps += abs(overall_elps (1)) + **abs(overall_elps** 2)) + **abs(overall_elps**(3))	calculate sum of magnitudes
end	end
for *i* = 1: 16	for each loop in SILS
res_overall_el(*i*) **= overall_elps**(*i*) **/ abs_elps**	calculate rescaled overall elasticity
end	end

Appendix C.
C.1. Yeast model
Equations of Yeast model (based on Vensim notation)

TIME STEP = 0.01; Integration method: Euler

Stocks
Cells = INTEG (births-deaths, 1)
Alcohol = INTEG (alcoholgeneration, 0)

Flows
births = (Cells/divisiontime)*eff alc birth
deaths = (Cells/lifetime)*eff alc death
alcoholgeneration = Cells*alcoholpercellgeneration

Auxiliaries
eff alc birth = (-0.1*Alcohol)+1.1
eff alc death = EXP(Alcohol-11)
lifetime = 30
divisiontime = 15
alcoholpercellgeneration = 0.01

The change in the eigenvalues of Yeast model over time

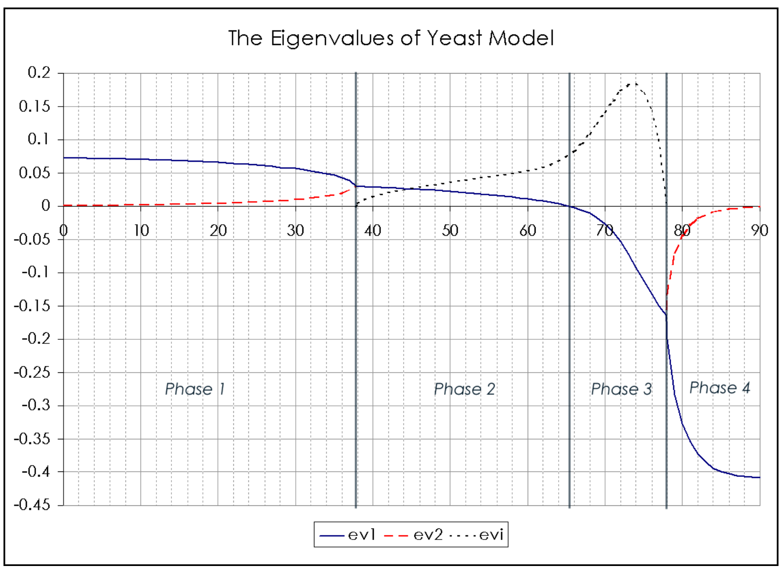

Figure C.1. Eigenvalues of Yeast model.

C.2. Predator-Prey model
Equations of Predator-Prey model (based on Vensim notation)

TIME STEP = 0.01 months; Integration method: Euler

Stocks
Predator = INTEG (predator birth-predator death, 1)
Prey = INTEG (prey birth-prey death, 2)

Flows
predator death = predator death rate*Predator
predator birth = predator interaction constant*Prey*Predator
prey birth = prey birth rate*Prey
prey death = prey interaction constant*Prey*Predator

Auxiliaries
predator interaction constant = 0.1
prey interaction constant = 0.2
predator death rate = 0.15
prey birth rate = 0.35

An alternative stock-flow representation of Predator-Prey model where the major loop L5 is concealed in the structure

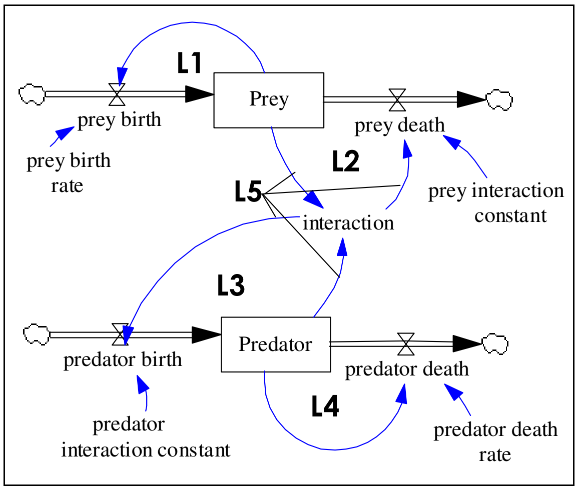

Figure C.2. An alternative stock-flow representation of Predator-Prey model.

The change in real and imaginary parts of the complex eigenvalue pair of Predator-Prey Model

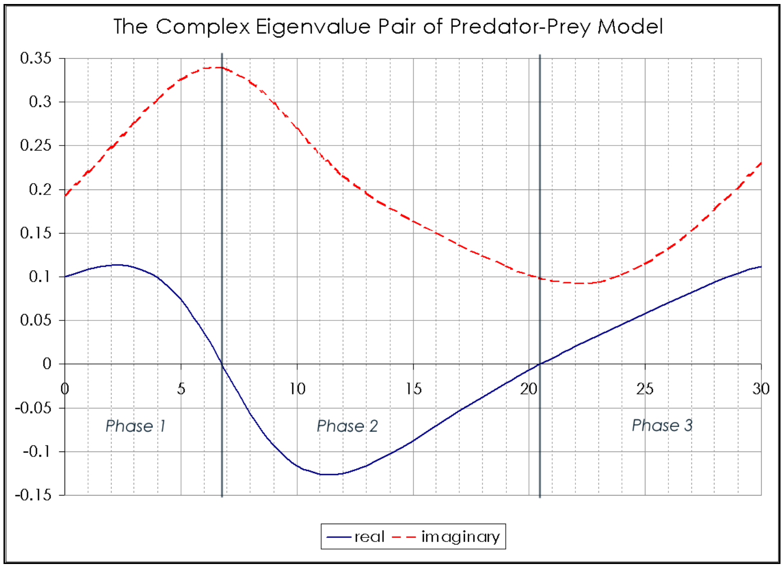

Figure C.3. Complex eigenvalue pair of Predator-Prey model.

C.3. Simple Long Wave model
Equations of Long Wave model (based on Vensim notation)

TIME STEP = 0.25 years; Integration method: Euler

Stocks
Capital = INTEG (Acquisitions-Depreciation, (capital output ratio*avg lifetime of capital)
/(avg lifetime of capital-capital output ratio))
Backlog = INTEG (Capital orders Backlog+goods orders-Production, normal delivery delay)
Supply = INTEG (Capital orders-Acquisitions, (Backlog/Production)*Depreciation)

Flows
Capital orders = Depreciation*relative orders
Acquisitions = (Supply*Production)/Backlog
Depreciation = Capital/avg lifetime of capital
Capital orders Backlog = Capital orders
Production = capacity*capacity utilization
goods orders = 1

Auxiliaries
capacity = Capital/capital output ratio
capacity utilization = capacity utilization fnc(desired production/capacity)
capital adjustment = (desired capital-Capital)/capital adjust time
desired capital = desired production*capital output ratio
desired orders = Depreciation+capital adjustment+supply adjustment
desired production = Backlog/normal delivery delay
desired supply line = Depreciation*(Backlog/Production)
relative orders = relative orders fnc(desired orders/Depreciation)
supply adjustment = (desired supply line-Supply)/supply adjust time

capacity utilization fnc(
[(0,0)-(2,1.1)],
(0,0),(0.2,0.3),(0.4,0.6),(0.6,0.8),(0.8,0.9),(1,1),(1.2,1.03),(1.4,1.05),(1.6,1.07),(1.8,1.09),(2,1.1))

relative orders fnc(
[(-1,0)-(40,6)],
(-1,0),(-0.5,0),(0,0.2),(0.5,0.5),(1,1),(1.5,1.5),(2,2),(2.5,2.5),(3,3),(3.5,3.5),(4,4),(4.5,4.4),(5,4.8),(5.5,5.2),
(6,5.5),(6.5,5.65),(7,5.7),(7.5,5.75),(8,5.8),(40,6))

avg lifetime of capital = 20
capital adjust time = 1.5
capital output ratio = 3
normal delivery delay = 1.5
supply adjust time = 1.5

The list of nodes, causal links and loops of Long Wave model

Table C.1. Nodes of Long wave model.

Nodes		
Acquisitions	capital adjustment	desired production
Backlog	Capital orders	desired supply line
capacity	Capital orders backlog	Production
capacity utilization	Depreciation	relative orders
capacity utilization fnc	desired capital	relative orders fnc
Capital	desired orders	Supply
		supply adjustment

Table C.2. Pathways originating from *Capital*.

Pathway no.	Variable sequence
c1c	Capital, capacity, Production, Acquisitions, Capital
c2c	Capital, capacity, capacity utilization, Production, Acquisitions, Capital
c3c	Capital, Depreciation, Capital
c1s	Capital, capacity, Production, Acquisitions, Supply
c2s	Capital, capacity, capacity utilization, Production, Acquisitions, Supply
c3s	Capital, Depreciation, Capital orders, Supply
c4s	Capital, capital adjustment, desired orders, relative orders, Capital orders, Supply
c5s	Capital, Depreciation, desired orders, relative orders, Capital orders, Supply
c6s	Capital, Depreciation, desired supply line, supply adjustment, desired orders, relative orders, Capital orders, Supply
c7s	Capital, capacity, Production, desired supply line, supply adjustment, desired orders, relative orders, Capital orders, Supply
c8s	Capital, capacity, capacity utilization, Production, desired supply line, supply adjustment, desired orders, relative orders, Capital orders, Supply
c9s	Capital, Depreciation, relative orders, Capital orders, Supply
c1b	Capital, capacity, Production, Backlog
c2b	Capital, capacity, capacity utilization, Production, Backlog
c3b	Capital, Depreciation, Capital orders, Capital orders backlog, Backlog
c4b	Capital, capital adjustment, desired orders, relative orders, Capital orders, Capital orders backlog, Backlog
c5b	Capital, Depreciation, desired orders, relative orders, Capital orders, Capital orders backlog, Backlog
c6b	Capital, Depreciation, desired supply line, supply adjustment, desired orders, relative orders, Capital orders, Capital orders backlog, Backlog
c7b	Capital, capacity, Production, desired supply line, supply adjustment, desired orders, relative orders, Capital orders, Capital orders backlog, Backlog
c8b	Capital, capacity, capacity utilization, Production, desired supply line, supply adjustment, desired orders, relative orders, Capital orders, Capital orders backlog, Backlog
c9b	Capital, Depreciation, relative orders, Capital orders, Capital orders backlog, Backlog

Table C.3. Pathways originating from *Supply*.

Pathway no.	Variable sequence
s1s	Supply, Acquisitions, Supply
s2s	Supply, supply adjustment, desired orders, relative orders, Capital orders, Supply
s1c	Supply, Acquisitions, Capital
s1b	Supply, supply adjustment, desired orders, relative orders, Capital orders, Capital orders backlog, Backlog

Table C.4. Pathways originating from *Backlog*.

Pathway no.	Variable sequence
b1b	Backlog, desired production, capacity utilization, Production, Backlog
b2b	Backlog, desired supply line, supply adjustment, desired orders, relative orders, Capital orders, Capital orders backlog, Backlog
b3b	Backlog, desired production, capacity utilization, Production, desired supply line, supply adjustment, desired orders, relative orders, Capital orders, Capital orders backlog, Backlog
b4b	Backlog, desired production, desired capital, capital adjustment, desired orders, relative orders, Capital orders, Capital orders backlog, Backlog
b1c	Backlog, Acquisitions, Capital
b2c	Backlog, desired production, capacity utilization, Production, Acquisitions, Capital
b1s	Backlog, Acquisitions, Supply
b2s	Backlog, desired production, capacity utilization, Production, Acquisitions, Supply
b3s	Backlog, desired production, capacity utilization, Production, desired supply line, supply adjustment, desired orders, relative orders, Capital orders, Supply
b4s	Backlog, desired production, desired capital, capital adjustment, desired orders, relative orders, Capital orders, Supply
b5s	Backlog, desired supply line, supply adjustment, desired orders, relative orders, Capital orders, Supply

Table C.5. Feedback loops in the Shortest Independent Loop Set of a simple Long wave model.

Loop no.	Loop name	Variable sequence
L1	Capital decay	Capital, Depreciation
L2	Supply line-1st order control	Supply, Acquisitions
L3	Economic growth	Production, Acquisitions, Capital, capacity
L4	Production scheduling	Backlog, desired production, capacity utilization, Production
L5	Capital expansion	Capital orders, Supply, Acquisitions, Capital, Depreciation
L6	Backlog expansion	Capital orders, Capital orders backlog, Backlog, Acquisitions, Capital, Depreciation,
L7	Supply line adjustment	Capital orders, Supply, supply adjustment, desired orders, relative orders
L8	Order fulfillment	Backlog, Acquisitions, Capital, capacity, Production
L9	Demand balancing	Acquisitions, Capital, capacity, capacity utilization, Production
L10	Steady-state Capital	Depreciation, relative orders, Capital orders, Supply, Acquisitions, Capital
L11	Hoarding	Capital orders backlog, Backlog, desired supply line, supply adjustment, desired orders, relative orders, Capital orders
L12	Capital replenishment	Depreciation, desired orders, relative orders, Capital orders, Supply, Acquisitions, Capital
L13	Capital adjustment	Capital, capital adjustment, desired orders, relative orders, Capital orders, Supply, Acquisitions
L14	Capital Self-ordering	Capital orders backlog, Backlog, desired production, desired capital, capital adjustment, desired orders, relative orders, Capital orders
L15	Steady-state Supply line	Depreciation, desired supply line, supply adjustment, desired orders, relative orders, Capital orders, Supply, Acquisitions, Capital
L16	Supply line adjustment (via Production)	Production, desired supply line, supply adjustment, desired orders, relative orders, Capital orders, Supply, Acquisitions, Capital, capacity

The list of causal links and the directed cycle matrix of simple Long wave model

Link no.	Variable sequence	Loop no.															
		L1	*L2*	*L3*	*L4*	*L5*	*L6*	*L7*	*L8*	*L9*	*L10*	*L11*	*L12*	*L13*	*L14*	*L15*	*L16*
cl1	Backlog - Acquisitions	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0
cl2	Backlog - desired production	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0
cl3	Backlog - desired Supply line	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
cl4	Capacity - capacity utilization	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
cl5	capacity - Production	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	1
cl6	capacity utilization - Production	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0
cl7	Capital - capacity	0	0	1	0	0	0	0	1	1	0	0	0	0	0	0	1
cl8	Capital - capital adjustment	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
cl9	Capital - Depreciation	1	0	0	0	1	1	0	0	0	1	0	1	0	0	1	0
cl10	capital adjustment-desired orders	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0
cl11	Capital orders-Capital orders Backlog	0	0	0	0	0	1	0	0	0	0	1	0	0	1	0	0
cl12	Depreciation - Capital orders	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0
cl13	Depreciation - desired orders	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
cl14	Depreciation-desired Supply line	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
cl15	Depreciation - relative orders	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
cl16	desired capital-capital adjustment	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
cl17	desired orders - relative orders	0	0	0	0	0	0	1	0	0	0	1	1	1	1	1	1
cl18	desired production-capacity utilization	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
cl19	Desired production-desired capital	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
cl20	desired Supply line-Supply adjustment	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	1
cl21	Production - Acquisitions	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0
cl22	Production - desired Supply line	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
cl23	relative orders - Capital orders	0	0	0	0	0	0	1	0	0	1	1	1	1	1	1	1
cl24	Supply - Acquisitions	0	1	0	0	1	0	0	0	0	1	0	1	1	0	1	1
cl25	Supply - Supply adjustment	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
cl26	Supply adjustment-desired orders	0	0	0	0	0	0	1	0	0	0	1	0	0	0	1	1

Figure C.4. List of causal links and directed cycle matrix of Long Wave model.

The partial derivative equations of simple Long wave model

$$\frac{\partial \dot{C}}{\partial C} = \frac{S}{cor * B}\left\{ f_{cu}(x) + C * \frac{df_{cu}(x)}{dx}\frac{\partial x}{\partial C}\right\} - \frac{1}{altc} \tag{C.1}$$

$$\frac{\partial \dot{C}}{\partial S} = \frac{C * f_{cu}(x)}{cor * B} \tag{C.2}$$

$$\frac{\partial \dot{C}}{\partial B} = \frac{S * C}{cor}\left\{ \frac{-f_{cu}(x)}{B^2} + \frac{df_{cu}(x)}{dx}\frac{\partial x}{\partial B}\right\} \tag{C.3}$$

$$\frac{\partial \dot{S}}{\partial C} = \frac{f_{ro}(\kappa)}{altc} + \frac{C}{altc}\frac{df_{ro}(\kappa)}{d\kappa}\frac{\partial \kappa}{\partial C} - \frac{S}{cor * B}\left\{ f_{cu}(x) + C\frac{df_{cu}(x)}{dx}\frac{\partial x}{\partial C}\right\} \tag{C.4}$$

$$\frac{\partial \dot{S}}{\partial S} = \frac{C}{altc}\frac{df_{ro}(\kappa)}{d\kappa}\frac{\partial \kappa}{\partial S} - \frac{C}{cor * B} f_{cu}(x) \tag{C.5}$$

$$\frac{\partial \dot{S}}{\partial B} = \frac{C}{altc}\frac{df_{ro}(\kappa)}{d\kappa}\frac{\partial \kappa}{\partial B} - \frac{S * C}{cor}\left\{ \frac{-f_{cu}(x)}{B^2} + \frac{1}{B}\frac{df_{cu}(x)}{dx}\frac{\partial x}{\partial B}\right\} \tag{C.6}$$

$$\frac{\partial \dot{B}}{\partial C} = \frac{f_{ro}(\kappa)}{altc} + \frac{C}{altc}\frac{df_{ro}(\kappa)}{d\kappa}\frac{\partial \kappa}{\partial C} - \left\{ \frac{f_{cu}(x)}{cor} + \frac{C}{cor}\frac{df_{cu}(x)}{dx}\frac{\partial x}{\partial C}\right\} \tag{C.7}$$

$$\frac{\partial \dot{B}}{\partial S} = \frac{C}{altc}\frac{df_{ro}(\kappa)}{d\kappa}\frac{\partial \kappa}{\partial S} \tag{C.8}$$

$$\frac{\partial \dot{B}}{\partial B} = \frac{C}{altc}\frac{df_{ro}(\kappa)}{d\kappa}\frac{\partial \kappa}{\partial B} - \frac{C}{cor}\frac{df_{cu}(x)}{dx}\frac{\partial x}{\partial B} \tag{C.9}$$

where $x = \frac{B}{ndl}\frac{cor}{C}$ and

$$\kappa = \left(\frac{C}{altc} + \frac{cor * B}{ndl * cat} - \frac{C}{cat} + \frac{B}{f_{cu}(x)}\frac{cor}{C}\frac{C}{altc}\frac{1}{sat} - \frac{S}{sat}\right) * \frac{altc}{C}.$$

The rest of the symbols are summarized in Table C.6.

Table C.6. Symbols used in Eqs. C.1-9.

Symbol	Corresponding variable/parameter in the model
B	*Backlog*
C	*Capital*
S	*Supply*
f_{cu}	Nonlinear function for *capacity utilization*
f_{ro}	Nonlinear function for *relative orders*
altc	*average lifetime of capital*
cor	*capital-output ratio*
ndl	*normal delivery delay*
cat	*capital adjustment time*
sat	*supply adjustment time*

The change in the eigenvalues of simple Long wave model over time

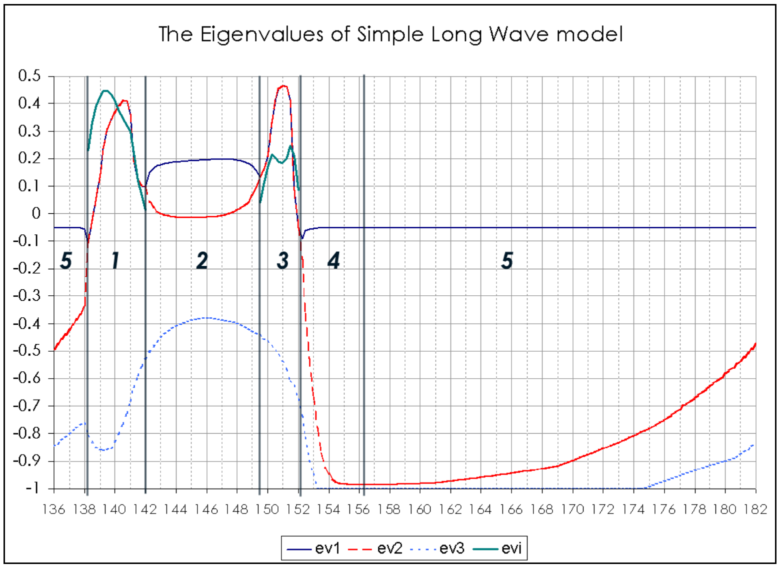

Figure C.5. Eigenvalues of simple Long wave model.

The set of equations for the causal link elasticities of simple Long wave model

$$e_{cl1} = e_{13} * \left(\frac{g_{13,b1c}}{g_{13}} \right) + e_{23} * \left(\frac{g_{23,b1s}}{g_{23}} \right) \tag{C.10}$$

$$e_{cl2} = e_{33} * \left(\frac{\left(g_{33,b1b} + g_{33,b3b} + g_{33,b4b} \right)}{g_{33}} \right) + e_{23} * \left(\frac{\left(\sum_{i=2}^{4} g_{23,bis} \right)}{g_{23}} \right) + e_{13} * \left(\frac{g_{13,b2c}}{g_{13}} \right) \tag{C.11}$$

$$e_{cl3} = e_{23} * \left(\frac{g_{23,b5s}}{g_{23}} \right) + e_{33} * \left(\frac{g_{33,b2b}}{g_{33}} \right) \tag{C.12}$$

$$e_{cl4} = e_{11} * \left(\frac{g_{11,c2c}}{g_{11}} \right) + e_{21} * \left(\frac{\left(g_{21,c2s} + g_{21,c8s} \right)}{g_{21}} \right) + e_{31} * \left(\frac{\left(g_{31,c2b} + g_{31,c8b} \right)}{g_{31}} \right) \tag{C.13}$$

$$e_{cl5} = e_{11} * \left(\frac{g_{11,c1c}}{g_{11}} \right) + e_{21} * \left(\frac{\left(g_{21,c1s} + g_{21,c7s} \right)}{g_{21}} \right) + e_{31} * \left(\frac{\left(g_{31,c1b} + g_{31,c7b} \right)}{g_{31}} \right) \tag{C.14}$$

$$\begin{aligned} e_{cl6} = {} & e_{11} * \left(\frac{g_{11,c2c}}{g_{11}} \right) + e_{13} * \left(\frac{g_{13,b2c}}{g_{13}} \right) + e_{21} * \left(\frac{\left(g_{21,c2s} + g_{21,c8s} \right)}{g_{21}} \right) + e_{23} * \left(\frac{\left(g_{23,b2s} + g_{23,b3s} \right)}{g_{23}} \right) \\ & + e_{31} * \left(\frac{\left(g_{31,c2b} + g_{31,c8b} \right)}{g_{31}} \right) + e_{33} * \left(\frac{\left(g_{33,b1b} + g_{33,b3b} \right)}{g_{33}} \right) \end{aligned} \tag{C.15}$$

$$\begin{aligned} e_{cl7} = {} & e_{11} * \left(\frac{\left(g_{11,c1c} + g_{11,c2c} \right)}{g_{11}} \right) + e_{21} * \left(\frac{\left(g_{21,c1s} + g_{21,c2s} + g_{21,c7s} + g_{21,c8s} \right)}{g_{21}} \right) \\ & + e_{31} * \left(\frac{\left(g_{31,c1b} + g_{31,c2b} + g_{31,c7b} + g_{31,c8b} \right)}{g_{31}} \right) \end{aligned} \tag{C.16}$$

$$e_{cl8} = e_{21} * \left(\frac{g_{21,c4s}}{g_{21}} \right) + e_{31} * \left(\frac{g_{31,c4b}}{g_{31}} \right) \tag{C.17}$$

$$e_{cl9} = e_{11} * \left(\frac{g_{11,c3c}}{g_{11}} \right) + e_{21} * \left(\frac{\left(g_{21,c3s} + g_{21,c5s} + g_{21,c6s} + g_{21,c9s} \right)}{g_{21}} \right) + e_{31} * \left(\frac{\left(g_{31,c3b} + g_{31,c5b} + g_{31,c6b} + g_{31,c9b} \right)}{g_{31}} \right) \tag{C.18}$$

$$e_{cl10} = e_{21} * \left(\frac{g_{21,c4s}}{g_{21}} \right) + e_{23} * \left(\frac{g_{23,b4s}}{g_{23}} \right) + e_{31} * \left(\frac{g_{31,c4b}}{g_{31}} \right) + e_{33} * \left(\frac{g_{33,b4b}}{g_{33}} \right) \tag{C.19}$$

$$e_{cl11} = e_{31} * \left(\frac{\left(\sum_{i=3}^{9} g_{31,cib} \right)}{g_{31}} \right) + e_{32} + e_{33} * \left(\frac{\left(\sum_{i=2}^{4} g_{33,bib} \right)}{g_{33}} \right) \tag{C.20}$$

$$e_{cl12} = e_{21} * \left(\frac{g_{21,c3s}}{g_{21}} \right) + e_{31} * \left(\frac{g_{31,c3b}}{g_{31}} \right) \tag{C.21}$$

$$e_{cl13} = e_{21} * \left(\frac{g_{21,c5s}}{g_{21}} \right) + e_{31} * \left(\frac{g_{31,c5b}}{g_{31}} \right) \tag{C.22}$$

$$e_{cl14} = e_{21} * \left(\frac{g_{21,c6s}}{g_{21}} \right) + e_{31} * \left(\frac{g_{31,c6b}}{g_{31}} \right) \tag{C.23}$$

$$e_{cl15} = e_{21} * \left(\frac{g_{21,c9s}}{g_{21}} \right) + e_{31} * \left(\frac{g_{31,c9b}}{g_{31}} \right) \tag{C.24}$$

$$e_{cl16} = e_{23} * \left(\frac{g_{23,b4s}}{g_{23}} \right) + e_{33} * \left(\frac{g_{33,b4b}}{g_{33}} \right) \tag{C.25}$$

$$\begin{aligned} e_{cl17} = {} & e_{21} * \left(\frac{\left(\sum_{i=4}^{8} g_{21,cis} \right)}{g_{21}} \right) + e_{22} * \left(\frac{g_{22,s1s}}{g_{22}} \right) + e_{23} * \left(\frac{\left(\sum_{i=3}^{5} g_{23,bis} \right)}{g_{23}} \right) \\ & + e_{31} * \left(\frac{\left(\sum_{i=4}^{8} g_{31,cib} \right)}{g_{31}} \right) + e_{32} + e_{33} * \left(\frac{\left(\sum_{i=2}^{4} g_{33,bib} \right)}{g_{33}} \right) \end{aligned} \tag{C.26}$$

$$e_{cl18} = e_{13} * \left(\frac{g_{13,b2c}}{g_{13}} \right) + e_{23} * \left(\frac{\left(g_{23,b2s} + g_{23,b3s} \right)}{g_{23}} \right) + e_{33} * \left(\frac{\left(g_{33,b1b} + g_{33,b3b} \right)}{g_{33}} \right) \tag{C.27}$$

$$e_{cl19} = e_{23} * \left(\frac{g_{23,b4s}}{g_{23}} \right) + e_{33} * \left(\frac{g_{33,b4b}}{g_{33}} \right) \tag{C.28}$$

$$e_{cl20} = e_{21} * \left(\frac{\left(\sum_{i=6}^{8} g_{21,cis} \right)}{g_{21}} \right) + e_{23} * \left(\frac{\left(g_{23,b3s} + g_{23,b5s} \right)}{g_{23}} \right) + e_{31} * \left(\frac{\left(\sum_{i=6}^{8} g_{31,cib} \right)}{g_{31}} \right) + e_{33} * \left(\frac{\left(g_{33,b2b} + g_{33,b3b} \right)}{g_{33}} \right) \quad (C.29)$$

$$e_{cl21} = e_{11} * \left(\frac{\left(g_{11,c1c} + g_{11,c2c} \right)}{g_{11}} \right) + e_{13} * \left(\frac{g_{13,b2c}}{g_{13}} \right) + e_{21} * \left(\frac{\left(g_{21,c1s} + g_{21,c2s} \right)}{g_{21}} \right) + e_{23} * \left(\frac{g_{23,b2s}}{g_{23}} \right) \quad (C.30)$$

$$e_{cl22} = e_{21} * \left(\frac{\left(g_{21,c7s} + g_{21,c8s} \right)}{g_{21}} \right) + e_{23} * \left(\frac{g_{23,b3s}}{g_{23}} \right) + e_{31} * \left(\frac{\left(g_{31,c7b} + g_{31,c8b} \right)}{g_{31}} \right) + e_{33} * \left(\frac{g_{33,b3b}}{g_{33}} \right) \quad (C.31)$$

$$e_{cl23} = e_{21} * \left(\frac{\left(\sum_{i=4}^{9} g_{21,cis} \right)}{g_{21}} \right) + e_{22} * \left(\frac{g_{22,s1s}}{g_{22}} \right) + e_{23} * \left(\frac{\left(\sum_{i=3}^{5} g_{23,bis} \right)}{g_{23}} \right)$$
$$+ e_{31} * \left(\frac{\left(\sum_{i=4}^{9} g_{31,cib} \right)}{g_{31}} \right) + e_{32} + e_{33} * \left(\frac{\left(\sum_{i=2}^{4} g_{33,bib} \right)}{g_{33}} \right) \quad (C.32)$$

$$e_{cl24} = e_{12} + e_{22} * \left(\frac{g_{22,s2s}}{g_{22}} \right) \quad (C.33)$$

$$e_{cl25} = e_{22} * \left(\frac{g_{22,s1s}}{g_{22}} \right) + e_{32} \quad (C.34)$$

$$e_{cl26} = e_{21} * \left(\frac{\left(\sum_{i=6}^{8} g_{21,cis} \right)}{g_{21}} \right) + e_{22} * \left(\frac{g_{22,s1s}}{g_{22}} \right) + e_{23} * \left(\frac{\left(g_{23,b3s} + g_{23,b5s} \right)}{g_{23}} \right)$$
$$+ e_{31} * \left(\frac{\left(\sum_{i=6}^{8} g_{31,cib} \right)}{g_{31}} \right) + e_{32} + e_{33} * \left(\frac{\left(g_{33,b2b} + g_{33,b3b} \right)}{g_{33}} \right) \quad (C.35)$$