Symbolic Control Analysis of Cellular Systems

Johann M. Rohwer[*], Timothy J. Akhurst and Jan-Hendrik S. Hofmeyr

Triple-J Group for Molecular Cell Physiology, Department of Biochemistry, Stellenbosch University, Private Bag X1, ZA-7602 Matieland, South Africa

E-Mail: *jr@sun.ac.za

Abstract

Metabolic Control Analysis (MCA) is a powerful quantitative framework for understanding and explaining the relationships between the global steady-state properties of a cellular system in terms of control coefficients, and the local properties of the individual components of the system in terms of elasticities. The elasticities are apparent kinetic orders, which derive directly from the kinetic properties of the enzymes. Since MCA relates elasticities to control coefficients through a matrix inversion, it allows one to predict and to quantify how the kinetics of individual enzymes affect the systemic behaviour of biological pathways. Most often this problem has been solved numerically, with algebraic and symbolic control analysis having been tackled less frequently. We present here a general implementation of the symbolic matrix inversion of MCA through symbolic algebraic computation. The algebraic expressions thus generated allow an in-depth analysis of where the control within a system lies and which parameters have the greatest effect on this control distribution, even if the exact values of the elasticities or control coefficients are unknown.

INTRODUCTION

Metabolic Control Analysis (MCA; see [1, 2]) is a powerful quantitative framework for analysing and quantifying the control and regulation of cellular pathways. It was developed independently by Kacser and Burns [3] and Heinrich and Rapoport [4] in the early 1970 s. One of the fundamentals of MCA is that it quantifies the steady-state behaviour of a system in terms of global properties (termed control coefficients) and local properties (termed elasticities). Mathematically, a control coefficient is defined as:

$$C_{v_i}^y = \frac{v_i}{y} \cdot \left(\frac{\partial y}{\partial v_i} \right)_{ss} = \left(\frac{\partial y/y}{\partial v_i/v_i} \right)_{ss} = \left(\frac{\partial \ln y}{\partial \ln v_i} \right)_{ss} \tag{1}$$

where y is any steady-state variable of the system (e. g. flux or species concentration) and v_i is the local rate of step i. The control coefficient thus quantifies how sensitive the variable y is to changes in local rate v_i. The subscript ss indicates that the entire system relaxes to a new steady state after the perturbation in v_i; hence, control coefficients are systemic properties.

An elasticity on the other hand, quantifies the effect of any molecular species or parameter that directly affects a unit step on the local rate through this step in isolation:

$$\varepsilon_{s_j}^{v_i} = \frac{s_j}{v_i} \cdot \left(\frac{\partial v_i}{\partial s_j} \right)_{s_k, s_l, \ldots} = \left(\frac{\partial v_i/v_i}{\partial s_j/s_j} \right)_{s_k, s_l, \ldots} = \left(\frac{\partial \ln v_i}{\partial \ln s_j} \right)_{s_k, s_l, \ldots} \tag{2}$$

where v_i is the local rate of unit step i in the system and s_j the concentration of any molecular species (substrate, product or effector) or parameter (e. g. K_m) that affects the step directly. The subscript $s_k, s_l, \ldots$ indicates that the concentrations of all other substrates, products and effectors are kept constant at their prevailing values while s_j is varied. Elasticities are apparent kinetic orders, which derive directly from the kinetics of the enzymes; their properties are local to the particular step and do not depend on the rest of the system.

A particular strength of MCA lies in the fact that it relates control coefficients to elasticities through a number of summation and connectivity relationships [3 – 5], thus enabling one to calculate systemic behaviour and control from the local properties of each of the components of the reaction network. These relationships have been combined into a variety of matrix equations (e. g. [6 – 8]), of which a particularly elegant form is that of Hofmeyr and Cornish-Bowden [9]:

$$\mathbf{C^i} \times \mathbf{E} = \mathbf{I} \quad \Rightarrow \quad \mathbf{C^i} = \mathbf{E}^{-1} \tag{3}$$

where $\mathbf{C^i} = \begin{bmatrix} \mathbf{C^{J_i}} & \mathbf{C^{s_i}} \end{bmatrix}^T$ is a matrix of independent flux- and concentration-control coefficients, and $\mathbf{E} = [\mathcal{K} \ -\varepsilon_s \mathcal{L}]$ is a matrix of all structural and local properties of the system. $\mathcal{K}$ is the scaled kernel matrix relating dependent fluxes to independent fluxes, ε_s is a matrix of elasticities, and $\mathcal{L}$ is the scaled link matrix relating the time-derivatives of the dependent species to the independent species – for further details, the reader is referred to [9, 10].

From Equation (3) it follows that the control coefficients can be calculated by inversion of the local matrix $\mathbf{E}$. Computational methods have traditionally focused on the numerical solution of this matrix inversion of MCA. A great number of simulation packages exist for computational systems biology (see http://sbml.org/), many of which can also do numerical MCA. Algebraically and symbolically, the problem has been tackled less frequently; the notable examples include control-pattern analysis [11] and the MetaCon program [12]. In both cases the aim was to derive algebraic relations between the control and elasticity coefficients. As it is often experimentally difficult to determine all parameter values for large models, it would be useful to develop a general approach for identifying key system parameters without knowing their actual values.

In this paper we present SymCA, a general implementation through symbolic algebraic computation of the matrix inversion of MCA (Equation (3)). The symbolic algebra manipulations are performed with Maxima, a powerful open-source algebra software. We have developed a Python interface to Maxima to access its symbolic capabilities from within PySCeS [13], the computational systems biology software developed in our group. This enabled the successful implementation of the $\mathbf{C^i = E^{-1}}$ inverse problem for systems of any size and complexity. As will be shown below, the algebraic expressions generated, which are factorized for easier interpretation, allow an in-depth analysis of where the control within a system lies. The rest of the paper is organized as follows: first, the design of the SymCA software is described, then its application is illustrated with two examples, and finally, the results are discussed in a broader context.

THE SYMCA SOFTWARE

Maxima (http://maxima.sourceforge.net/) is an open-source symbolic algebra software licenced under the GNU GPL. It runs on multiple operating systems and is capable of performing symbolic calculus, linear algebra, matrix manipulations and simplifications, factorization, etc. Maxima is a descendant of Macsyma, the famous algebra software developed at the MIT in the 1960 s. The program is implemented in common Lisp. Because of its open-source licence and multi-platform capabilities, we chose Maxima to integrate symbolic control analysis capabilities into PySCeS.

SymCA (Symbolic Control Analysis) refers to the combination of Python with PySCeS and Maxima. The code is released under an open-source licence together with the PySCeS project at http://pysces.sourceforge.net/. The operation of the software is summarized in Fig. 1.
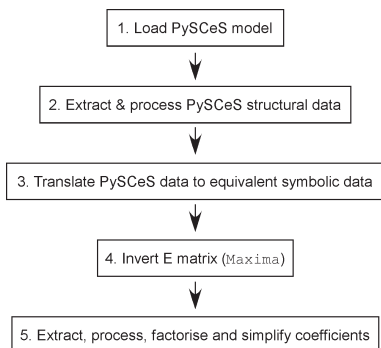
Figure 1. Operation of the SymCA software.

The symbolic algebra capabilities of Maxima are accessed through a Python interface, which we have developed using the subprocess module. The analysis is started by loading a model into PySCeS and performing a structural (stoichiometric) analysis. This extracts and processes the structural data and generates the $\mathcal{K}$, ε_s and $\mathcal{L}$ matrices, from which the **E** matrix (Equation (3)) is assembled. SymCA subsequently translates these data into equivalent *symbolic* data by substituting text strings for the elasticity and species names into the matrix entries. The symbolic **E** matrix is then passed to Maxima using our newly developed interface, where it is symbolically inverted and the results passed back to Python. In the final step, SymCA extracts, processes and simplifies the symbolic control coefficient expressions, which can then be output in various formats. The whole process, including the details of this simplification, are illustrated with an example (see SymCA by Example below).

Additional features

The SymCA software has a number of additional features built in for error checking and user convenience:

- The control coefficient expressions can be output in L^{A}T$_E$X [14] format for easy typesetting and incorporation into documents. The text format for loading the expressions into Maxima is also saved.

- The expressions are automatically *checked for correctness* by substituting the numerical values of the elasticities from the PySCeS model, calculating the control coefficient values, and comparing these to the control coefficients calculated numerically directly by PySCeS.

- SymCA allows *numerical substitution* (e.g. 0 or 1) of selected elasticity values to generate a simplified control coefficient expression subject to certain assumptions

about the model, such as a particular reaction being saturated with substrate (i.e. following zero-order kinetics) or operating in the first-order range.

- Expressions for *parameter-response coefficients* can be calculated from the partitioned response property [3]: $C_p^y = e_p^{v_i} \times C_{v_i}^y$, where p is an external parameter acting on step i. In fact, this merely involves multiplying the control coefficient expression by the parameter elasticity of step i towards p.

SymCA BY EXAMPLE

The operation of the SymCA software will be illustrated with two examples: first, a small "core" model, which has been designed to emphasize the salient features, and next, a more realistic model of fermentation pathways in *Lactococcus lactis*.
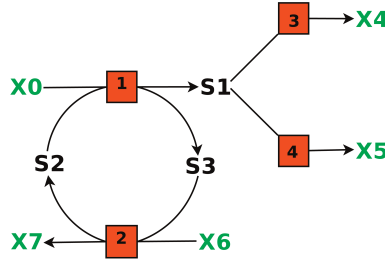
Simple metabolic pathway

Figure 2. A simple metabolic pathway with a branch-point and a moiety-conserved cycle [10].

Figure 2 shows a scheme of a "minimal" system containing an example of the most important structural features observed in metabolic pathways, i.e. a branch-point and a moiety-conserved cycle. This example has been treated in detail to explain the matrix method of MCA [10] and will also be used here to demonstrate the workings of the SymCA software.

For the system in Fig. 2, Equation (3) reads:

$$
\begin{bmatrix}
C_3^{J_3} & C_4^{J_3} & C_1^{J_3} & C_2^{J_3} \\
C_3^{J_4} & C_4^{J_4} & C_1^{J_4} & C_2^{J_4} \\
C_3^{S_1} & C_4^{S_1} & C_1^{S_1} & C_2^{S_1} \\
C_3^{S_2} & C_4^{S_2} & C_1^{S_2} & C_2^{S_2}
\end{bmatrix}
=
\begin{bmatrix}
1 & 0 & -\varepsilon_{s_1}^{v_3} & 0 \\
0 & 1 & -\varepsilon_{s_1}^{v_4} & 0 \\
\dfrac{J_3}{J_1} & \dfrac{J_4}{J_1} & -\varepsilon_{s_1}^{v_1} & \left(\varepsilon_{s_3}^{v_1}\dfrac{s_2}{s_3} - \varepsilon_{s_2}^{v_1}\right) \\
\dfrac{J_3}{J_2} & \dfrac{J_4}{J_2} & 0 & \left(\varepsilon_{s_3}^{v_2}\dfrac{s_2}{s_3} - \varepsilon_{s_2}^{v_2}\right)
\end{bmatrix}^{-1}
\qquad (4)
$$

For Equation (4) to be computed by SymCA, the **K** and **L** matrices (which are provided by PySCeS from the stoichiometric analysis) first have to be scaled to $\mathcal{K}$ and $\mathcal{L}$ as described in [10]. Subsequently, the matrix product $-\varepsilon_s\mathcal{L}$ is computed. Before proceeding with the symbolic matrix inversion, dependent fluxes are expressed in terms of independent ones according to $\mathbf{J} = \mathbf{KJ_i}$, which for the system in Fig. 2 reads:

$$\begin{bmatrix} J_3 \\ J_4 \\ J_1 \\ J_2 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 1 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} J_3 \\ J_4 \end{bmatrix} \tag{5}$$

Equation (5) shows that J_1 and J_2 are both equivalent to $J_3 + J_4$; thus, in the RHS of Equation (4) J_2 is replaced with J_1. The symbolic matrix **E** is then passed to Maxima for inversion. The determinant of **E** (also computed by Maxima) gives the common denominator of all the control coefficient expressions. For the example model in Fig. 2, this reads:

$$\sum = \frac{J_4\varepsilon_{s1}^4\varepsilon_{s3}^2}{s_3} - \frac{J_4\varepsilon_{s1}^4\varepsilon_{s2}^2}{s_2} - \frac{J_4\varepsilon_{s1}^4\varepsilon_{s3}^1}{s_3} + \frac{J_4\varepsilon_{s1}^4\varepsilon_{s2}^1}{s_2} + \frac{J_3\varepsilon_{s1}^3\varepsilon_{s3}^2}{s_3}$$
$$- \frac{J_3\varepsilon_{s1}^3\varepsilon_{s2}^2}{s_2} - \frac{J_3\varepsilon_{s1}^3\varepsilon_{s3}^1}{s_3} + \frac{J_3\varepsilon_{s1}^3\varepsilon_{s2}^1}{s_2} - \frac{J_1\varepsilon_{s1}^1\varepsilon_{s3}^2}{s_3} + \frac{J_1\varepsilon_{s1}^1\varepsilon_{s2}^2}{s_2} \tag{6}$$

To obtain the individual control coefficient expressions, the denominator (Equation (6)) is first factored out of the symbolic inverse of **E**. Next, the expressions are rearranged and simplified so that:

- elasticity terms are only *multiplied* by fluxes (i. e. not divided), and
- elasticities towards species in conserved moieties are always divided by the concentration of that *same species*.

As an example, we show the expressions thus generated for the control coefficients on flux J_3:

$$C_1^{J_3} = \left(\frac{J_1\varepsilon_{s1}^3\varepsilon_{s3}^2}{s_3} - \frac{J_1\varepsilon_{s1}^3\varepsilon_{s2}^2}{s_2} \right) \bigg/ \sum \tag{7}$$

$$C_2^{J_3} = \left(\frac{J_1\varepsilon_{s1}^3\varepsilon_{s2}^1}{s_2} - \frac{J_1\varepsilon_{s1}^3\varepsilon_{s3}^1}{s_3} \right) \bigg/ \sum \tag{8}$$

$$C_3^{J_3} = \left(\frac{J_4\varepsilon_{s1}^4\varepsilon_{s3}^2}{s_3} - \frac{J_4\varepsilon_{s1}^4\varepsilon_{s2}^2}{s_2} - \frac{J_4\varepsilon_{s1}^4\varepsilon_{s3}^1}{s_3} + \frac{J_4\varepsilon_{s1}^4\varepsilon_{s2}^1}{s_2} - \frac{J_1\varepsilon_{s1}^1\varepsilon_{s3}^2}{s_3} + \frac{J_1\varepsilon_{s1}^1\varepsilon_{s2}^2}{s_2} \right) \bigg/ \sum \tag{9}$$

$$C_4^{J_3} = \left(-\frac{J_4 \varepsilon_{s1}^3 \varepsilon_{s3}^2}{s_3} + \frac{J_4 \varepsilon_{s1}^3 \varepsilon_{s2}^2}{s_2} + \frac{J_4 \varepsilon_{s1}^3 \varepsilon_{s3}^1}{s_3} - \frac{J_4 \varepsilon_{s1}^3 \varepsilon_{s2}^1}{s_2} \right) \Big/ \sum \tag{10}$$

with Σ as in Equation (6). The automatic rearrangement and factorization of the terms has the consequence that the generated control coefficient expressions such as in Equations (7) – (10) always end up in a standard format that is easily interpretable: every term is a *control pattern* [11], which can be visualized as a "chain of local effects" through the pathway.

Fermentation pathways in Lactococcus lactis

To illustrate the application of SymCA with a more realistic model, we have analysed the fermentation pathways of *Lactococcus lactis*. The pathway shown in Fig. 3 is a simplified version of the model reported by Hoefnagel *et al.* [15] with the branch to acetolactate omitted.
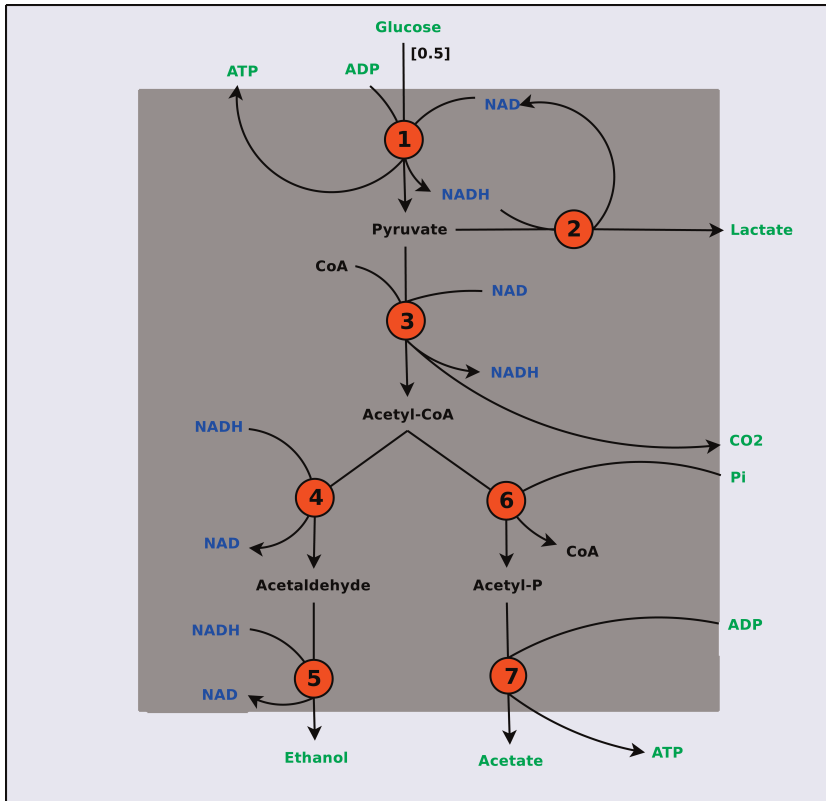

Figure 3. Fermentation pathways in *Lactococcus lactis.* Abbreviations for Equations (11) and (12): *PYR,* Pyruvate; *ACCOA,* Acetyl-CoA; *ACAL,* Acetaldehyde; *ACP,* Acetyl-P.

When processing this model with SymCA, the following expression for $C_1^{J_2}$ is generated, which we show by way of example:

$$
\begin{aligned}
C_1^{J_2} = \bigg(&-\frac{J_1 J_5 J_6 \varepsilon_{ACAL}^4 \varepsilon_{ACP}^7 \varepsilon_{COA}^6 \varepsilon_{NADH}^5 \varepsilon_{PYR}^2}{COA\ NADH} + \frac{J_1 J_5 J_6 \varepsilon_{ACAL}^5 \varepsilon_{ACP}^7 \varepsilon_{COA}^6 \varepsilon_{NADH}^4 \varepsilon_{PYR}^2}{COA\ NADH} \\
&+\frac{J_1 J_5 J_6 \varepsilon_{ACAL}^5 \varepsilon_{ACP}^7 \varepsilon_{COA}^6 \varepsilon_{NADH}^2 \varepsilon_{PYR}^3}{COA\ NADH} - \frac{J_1 J_5 J_6 \varepsilon_{ACAL}^5 \varepsilon_{ACP}^7 \varepsilon_{COA}^6 \varepsilon_{NADH}^3 \varepsilon_{PYR}^2}{COA\ NADH} \\
&-\frac{J_1 J_3 J_6 \varepsilon_{ACAL}^4 \varepsilon_{ACP}^7 \varepsilon_{COA}^6 \varepsilon_{NADH}^2 \varepsilon_{PYR}^3}{COA\ NADH} + \frac{J_1 J_3 J_6 \varepsilon_{ACAL}^4 \varepsilon_{ACP}^7 \varepsilon_{COA}^6 \varepsilon_{NADH}^3 \varepsilon_{PYR}^2}{COA\ NADH} \\
&+\frac{J_1 J_5 J_6 \varepsilon_{ACAL}^4 \varepsilon_{ACCOA}^6 \varepsilon_{ACP}^7 \varepsilon_{NADH}^5 \varepsilon_{PYR}^2}{ACCOA\ NADH} - \frac{J_1 J_5 J_6 \varepsilon_{ACAL}^4 \varepsilon_{ACCOA}^6 \varepsilon_{ACP}^7 \varepsilon_{NADH}^4 \varepsilon_{PYR}^2}{ACCOA\ NADH} \\
&\quad + \ldots \\
-\frac{J_1 J_3 J_6 \varepsilon_{ACAL}^4 \varepsilon_{ACCOA}^6 \varepsilon_{ACP}^7 \varepsilon_{NAD}^2 \varepsilon_{PYR}^3}{ACCOA\ NAD} &+ \frac{J_1 J_3 J_6 \varepsilon_{ACAL}^4 \varepsilon_{ACCOA}^6 \varepsilon_{ACP}^7 \varepsilon_{NAD}^3 \varepsilon_{PYR}^2}{ACCOA\ NAD} \bigg) \bigg/ \sum
\end{aligned}
\tag{11}
$$

(24 terms)

with the denominator Σ given by:

$$
\begin{aligned}
\sum = {} &\frac{J_3 J_5 J_6 \varepsilon_{ACAL}^4 \varepsilon_{ACP}^7 \varepsilon_{COA}^6 \varepsilon_{NADH}^5 \varepsilon_{PYR}^3}{COA\ NADH} + \frac{J_2 J_5 J_6 \varepsilon_{ACAL}^4 \varepsilon_{ACP}^7 \varepsilon_{COA}^6 \varepsilon_{NADH}^5 \varepsilon_{PYR}^2}{COA\ NADH} \\
&-\frac{J_1 J_5 J_6 \varepsilon_{ACAL}^4 \varepsilon_{ACP}^7 \varepsilon_{COA}^6 \varepsilon_{NADH}^5 \varepsilon_{PYR}^1}{COA\ NADH} - \frac{J_3 J_5 J_6 \varepsilon_{ACAL}^5 \varepsilon_{ACP}^7 \varepsilon_{COA}^6 \varepsilon_{NADH}^4 \varepsilon_{PYR}^3}{COA\ NADH} \\
&-\frac{J_2 J_5 J_6 \varepsilon_{ACAL}^5 \varepsilon_{ACP}^7 \varepsilon_{COA}^6 \varepsilon_{NADH}^4 \varepsilon_{PYR}^2}{COA\ NADH} + \frac{J_1 J_5 J_6 \varepsilon_{ACAL}^5 \varepsilon_{ACP}^7 \varepsilon_{COA}^6 \varepsilon_{NADH}^4 \varepsilon_{PYR}^1}{COA\ NADH} \\
&\quad + \ldots \\
+\frac{J_2 J_3 J_6 \varepsilon_{ACAL}^4 \varepsilon_{ACCOA}^6 \varepsilon_{ACP}^7 \varepsilon_{NAD}^2 \varepsilon_{PYR}^3}{ACCOA\ NAD} &- \frac{J_1 J_3 J_6 \varepsilon_{ACAL}^4 \varepsilon_{ACCOA}^6 \varepsilon_{ACP}^7 \varepsilon_{NAD}^1 \varepsilon_{PYR}^3}{ACCOA\ NAD} \\
-\frac{J_2 J_3 J_6 \varepsilon_{ACAL}^4 \varepsilon_{ACCOA}^6 \varepsilon_{ACP}^7 \varepsilon_{NAD}^3 \varepsilon_{PYR}^2}{ACCOA\ NAD} &+ \frac{J_1 J_3 J_6 \varepsilon_{ACAL}^4 \varepsilon_{ACCOA}^6 \varepsilon_{ACP}^7 \varepsilon_{NAD}^3 \varepsilon_{PYR}^1}{ACCOA\ NAD}
\end{aligned}
\tag{12}
$$

(56 terms)

The above expression is rather unwieldy and contains a very large number of terms, which makes it difficult to interpret. However, three aspects are noteworthy:

1. The expression is *mathematically correct,* providing an analytical solution for the control coefficient in terms of elasticities.

2. The expression is arranged and factorized so that all terms are similar and conform to the *standard format* described above.

3. The expression can be *simplified* subject to certain assumptions about the pathway.

To illustrate the last point, assume, for example, that $\varepsilon_{PYR}^1 = \varepsilon_{PYR}^2 = 0$ (i.e. that both reactions 1 and 2 are insensitive to changes in the pyruvate concentration). The expression for $C_1^{J_2}$ then simplifies to (automatically calculated by SymCA):

Symbolic Control Analysis of Cellular Systems

$$C_1^{J_2} = \frac{-J_1 J_5 \varepsilon_{ACAL}^5 \varepsilon_{NAD}^2 + J_1 J_3 \varepsilon_{ACAL}^4 \varepsilon_{NAD}^2 + J_1 J_5 \varepsilon_{ACAL}^5 \varepsilon_{NADH}^2 - J_1 J_3 \varepsilon_{ACAL}^4 \varepsilon_{NADH}^2}{J_3 J_5 \varepsilon_{ACAL}^4 \varepsilon_{NADH}^5 - J_3 J_5 \varepsilon_{ACAL}^5 \varepsilon_{NADH}^4 - J_2 J_5 \varepsilon_{ACAL}^5 \varepsilon_{NADH}^2 + J_1 J_5 \varepsilon_{ACAL}^5 \varepsilon_{NADH}^1}$$

$$+ J_2 J_3 \varepsilon_{ACAL}^4 \varepsilon_{NADH}^2 - J_1 J_3 \varepsilon_{ACAL}^4 \varepsilon_{NADH}^1 - J_3 J_5 \varepsilon_{ACAL}^5 \varepsilon_{NAD}^5 + J_3 J_5 \varepsilon_{ACAL}^5 \varepsilon_{NAD}^4 \qquad (13)$$

$$+ J_2 J_5 \varepsilon_{ACAL}^5 \varepsilon_{NAD}^2 - J_1 J_5 \varepsilon_{ACAL}^5 \varepsilon_{NAD}^1 - J_2 J_3 \varepsilon_{ACAL}^4 \varepsilon_{NAD}^2 + J_1 J_3 \varepsilon_{ACAL}^4 \varepsilon_{NAD}^1$$

The number of terms has been reduced from 24 to 4 in the numerator and from 56 to 12 in the denominator. Clearly, Equation (13) is of a more manageable size so that direct algebraic analysis and interpretation is possible.

DISCUSSION AND CONCLUSION

The framework of MCA has been a driving force in dispelling the notion of the "rate-limiting step" in metabolic pathways, as was elegantly illustrated by the pioneering experimental work of Groen and co-workers [16] on the control of mitochondrial oxidative phosphorylation. However, MCA goes beyond the mere quantification of control coefficients – in our view, the real power of MCA lies in its relation of control coefficients to elasticities, thus allowing us to infer systemic properties from the characteristics of the isolated system components.

In this paper we have described SymCA, a software that implements the symbolic matrix inversion of MCA in a generalized way and generates analytical expressions for the control coefficients of a pathway in terms of the elasticities. This has a number of uses:

- Computational analysis of biochemical pathways is becoming increasingly important in the burgeoning field of computational systems biology. An inspection of on-line model databases such as JWS Online ([17], http://jjj.biochem.sun.ac.za/) or BioModels ([18], http://www.ebi.ac.uk/biomodels) reveals that the number of available models grows monthly, but that they are also increasing in size and complexity. Here, MCA can become an increasingly important analysis tool by dissecting where the control in a pathway lies and identifying the factors that determine this control. For example, the expressions generated with SymCA can be used to identify key elasticities that are responsible for a particularly large (or small) value of a certain control coefficient. Furthermore, the analysis can easily be extended to parameter-response coefficients (see *Additional features* above), thus addressing the question of which parameters in a kinetic model have the largest effect on a particular observed behaviour, and how this effect is transmitted.

- *Fermentation pathways in Lactococcus lactis* (above), shows that the MCA expressions quickly become unwieldy once the model size grows beyond a few reactions. In such cases, assumptions can be introduced to simplify the expres-

sions, as illustrated above. Very often, it is known that a particular reaction is saturated with substrate or insensitive to product under cellular conditions (allowing the elasticity to be set to zero) or is operating in the first-order range because the substrate concentration is lower than the K_m (allowing the elasticity to be set to one). Moreover, certain elasticities can be set numerically to vary within bounds to explore how the control distribution in the network would be affected.

- The development of kinetic models is often hampered by the fact that not all of the kinetic parameters are known. The significance of symbolic MCA is that it is valid in general, and as such does not depend on the availability of particular parameter values. All that is required is the stoichiometry and mapping of the allosteric modifier interactions. Thus, general conclusions about the control structure of a pathway may be drawn even if not all the kinetic parameter details are known.

- As was already pointed out by Hofmeyr almost 20 years ago [11], the individual terms of a control coefficient expression can be visualized on the network as a "control pattern" that describes a "chain of local effects" corresponding to a particular route of regulation. Symbolic control analysis can thus help identify such routes of regulation in a complex network and quantify their relative importance (e. g. comparing feedback inhibition along the main chain of a pathway *vs.* an allosteric feedback loop).

It should be mentioned that programmatic symbolic control analysis is not new. The problem has been tackled by Thomas and Fell with the MetaCon [12] computer program. However, their approach is not completely general in the sense that the analysis of branched pathways always requires the manual selection of a reference flux before the expressions can be generated and selection of different reference fluxes leads to different expressions, whereas the matrix method on which SymCA is based does not have this limitation. Moreover, the integration of SymCA data structures within PySCeS means that it is easy to substitute some or all of the elasticities with numerical values, allowing for further analysis, simplification or validation of the expressions.

In conclusion, as the field of computational systems biology grows it can be anticipated that the complexity of models will increase, approaching the level of complexity of the modelled systems themselves. Analysis tools will become ever more essential for making sense of huge amounts of model data. Symbolic control analysis is one such tool the SymCA software presented here facilitates and contributes to this analysis.

ACKNOWLEDGEMENTS

REFERENCES

[1] Fell, D.A. *Understanding the Control of Metabolism*; Portland Press: London, 1996.

[2] Heinrich, R., Schuster, S. *The Regulation of Cellular Systems;* Chapman & Hall, New York, 1996.

[3] Kacser, H., Burns, J.A. (1973) The control of flux. *Symp. Soc. Exp. Biol.* **27**:65–104.

[4] Heinrich, R., Rapoport, T.A. (1974) A linear steady-state treatment of enzymatic chains. General properties, control and effector strength. *Eur. J. Biochem*. **42**:89–95.

[5] Westerhoff, H.V., Chen, Y.-D. (1984) How do enzyme activities control metabolite concentrations? An additional theorem in the theory of metabolic control. *Eur. J. Biochem.* **142**:425–430.

[6] Sauro, H.M., Small, J.R., Fell, D.A. (1987) Metabolic control and its analysis: extensions to the theory and matrix method. *Eur. J. Biochem.* **165**:215–221.

[7] Small, J.R., Fell, D.A. (1989) The matrix method of metabolic control analysis: Its validity for complex pathway structures. *J. Theor. Biol.* **136**:181–197.

[8] Westerhoff, H.V., Hofmeyr, J.-H.S., Kholodenko, B.N. (1994) Getting to the inside of cells using metabolic control analysis. *Biophys. Chem.* **50**:273–283.

[9] Hofmeyr, J.-H.S., Cornish-Bowden, A. (1996) Co-response analysis: A new experimental strategy for metabolic control analysis. *J. Theor. Biol.* **182**:371–380.

[10] Hofmeyr, J.-H.S. Metabolic control analysis in a nutshell. In *Proceedings of the 2nd International Conference on Systems Biology*, 2001, (Yi, T.-M., Hucka, M., Morohashi, M. and Kitano, H., Eds) Omnipress, Madison, WI, 2001; pp. 291–300.

[11] Hofmeyr, J.-H.S. (1989) Control-pattern analysis of metabolic pathways. Flux and concentration control in linear pathways. *Eur. J. Biochem.* **186**:343–354.

[12] Thomas, S., Fell, D.A. (1993) A computer program for the algebraic determination of control coefficients in metabolic control analysis. *Biochem. J.* **292**:351–360.

[13] Olivier, B.G., Rohwer, J.M., Hofmeyr, J.-H.S. (2005) Modelling cellular systems with PySCeS. *Bioinformatics* **21**:560–561.

[14] Lamport, L. *L^{A}T$_E$X: A Document Preparation System: User's guide and reference,* 2nd ed; Addison-Wesley Professional, Reading, MA, 1994.

[15] Hoefnagel, M.H.N., Starrenburg, M.J.C., Martens, D.E., Hugenholtz, J., Kleerebezem, M., Swam, I.I.V., Bongers, R., Westerhoff, H.V., Snoep, J.L. (2002) Metabolic engineering of lactic acid bacteria, the combined approach: kinetic modelling, metabolic control and experimental analysis. *Microbiology* **148**:1003–1013.

[16] Groen, A.K., Wanders, R. J.A., Westerhoff, H.V., van der Meer, R., Tager, J.M. (1982) Quantification of the contribution of various steps to the control of mitochondrial respiration. *J. Biol. Chem.* **257**: 2754 – 2757.

[17] Olivier, B.G., Snoep, J.L. (2004) Web-based kinetic modelling using JWS Online. *Bioinformatics* **20**:2143 – 2144.

[18] LeNovère, N., Bornstein, B., Broicher, A., Courtot, M., Donizelli, M., Dharuri, H., Li, L., Sauro, H., Schilstra, M., Shapiro, B., Snoep, J.L., Hucka, M. (2006) BioModels Database: a free, centralized database of curated, published, quantitative kinetic models of biochemical and cellular systems. *Nucleic Acids Res.* **34**:D 689 –D 691.