BIOINFORMATICS ORIGINAL PAPER
Vol. 21 no. 9 2005, pages 2017–2026
doi:10.1093/bioinformatics/bti255

Systems biology

On the use of qualitative reasoning to simulate and identify metabolic pathways

Ross D. King[1,*], Simon M. Garrett[1] and George M. Coghill[2]

[1]Department of Computer Science, University of Wales, Aberystwyth, Wales, SY23 3DB, UK and
[2]Department of Computing Science, University of Aberdeen, Aberdeen, ABD24 3UE, UK

Received on January 23, 2004; revised on December 3, 2004; accepted on December 27, 2004
Advance Access publication January 10, 2005

ABSTRACT

Motivation: Perhaps the greatest challenge of modern biology is to develop accurate *in silico* models of cells. To do this we require computational formalisms for both simulation (how according to the model the state of the cell evolves over time) and identification (learning a model cell from observation of states). We propose the use of qualitative reasoning (QR) as a unified formalism for both tasks. The two most commonly used alternative methods of modelling biochemical pathways are ordinary differential equations (ODEs), and logical/graph-based (LG) models.

Results: The QR formalism we use is an abstraction of ODEs. It enables the behaviour of many ODEs, with different functional forms and parameters, to be captured in a single QR model. QR has the advantage over LG models of explicitly including dynamics. To simulate biochemical pathways we have developed 'enzyme' and 'metabolite' QR building blocks that fit together to form models. These models are finite, directly executable, easy to interpret and robust. To identify QR models we have developed heuristic chemoinformatics graph analysis and machine learning procedures. The graph analysis procedure is a series of constraints and heuristics that limit the number of ways metabolites can combine to form pathways. The machine learning procedure is generate-and-test inductive logic programming. We illustrate the use of QR for modelling and simulation using the example of glycolysis.

Availability: All data and programs used are available on request.

Contact: rdk@aber.ac.uk

1 INTRODUCTION

The completion of the sequencing of the key model genomes and the rise of post-genomic technologies have opened up the prospect of modelling cells *in silico* in unprecedented detail. Such models will be essential to integrate our growing biological knowledge, and have the potential to transform medicine and biotechnology. A good scientific model should both reflect the causal structure of the physical system under study, and be able to efficiently predict the outcome of new experiments. There are currently two key research questions in cellular modelling:

(1) What are the most appropriate computational formalisms for simulation of cellular processes?

(2) How can novel cellular models be identified (learned) directly from experimental data?

In this paper we demonstrate the utility of qualitative reasoning as a unified formalism for both simulating and identifying metabolic models.

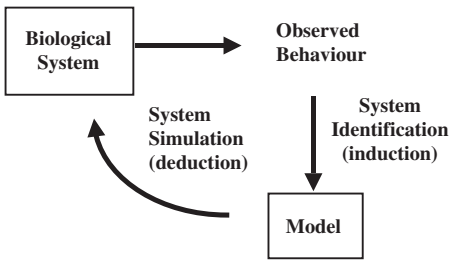

Fig. 1. The relationship between model simulation and system identification.

1.1 Model simulation versus model identification

It is important to clearly distinguish between model simulation and model identification (Fig. 1). To make experimental predictions we use a model in conjunction with a simulator. This is a form of deductive inference. For example, a dynamic model of glycolysis might tell you how the level of pyruvate in a cell varies over time as the amount of glucose increases. If the deductive predictions of a model are inconsistent with observed behaviour then the model is falsified.

The task of forming a model to explain a given set of experimental results is called model identification. This is a form of inductive inference. For example, if the levels of the metabolites in glycolysis are observed over a series of time steps, and from this data the reactions of glycolysis are inferred, this would be model identification. The application of automatic model identification is generally recognised to be essential for large-scale *in silico* modelling. Model identification is sometimes referred to as the 'inverse' problem in the bioinformatics literature. In the control engineering literature model identification is known as system identification and has been extensively studied (Ljung, 1999).

1.2 Cell modelling

There are three main types of cellular process modelled in bioinformatics: biochemical pathways (the metabolome), gene-networks (the transcriptome) and protein signal-transduction (the proteome).

*To whom correspondence should be addressed.

In vivo, these processes are interrelated and entangled, but they are still generally modelled separately. Abstractly, modelling all three types of process is similar, with models constraining the change over time of levels of metabolite, mRNA or protein.

Many different formalisms have been applied to cellular modelling: differential equations (Bakker *et al*., 1997; Eisenthal and Cornish-Bowden, 1998; Ferrell and Machleder, 1998; Alon *et al*., 1999; von Dassow *et al*., 2000; Santillan and Mackey, 2001), S-systems (Akutsu *et al*., 2000), Boolean networks (Kauffman, 1993; Somogyi and Sniegoski, 1996), logical networks (Karp *et al*., 1996), Bayesian networks (Friedman *et al*., 2000), Petri-nets (Matsuno *et al*., 2000), Π-calculus (Regev *et al*., 2001), etc. Given our present state of knowledge of cellular processes, there is no single best modelling formalism and the choice of formalism depends on the problem domain. It is a case of 'horses for courses'. For example, if in a metabolic pathway all the kinetic properties of the enzymes involved are known, an ordinary differential equation (ODE) model would be appropriate. However, if these kinetics properties are not known, an ODE model could be inappropriate; the use of arbitrary kinetics would produce a misleading impression of precision, and a simpler qualitative model would be preferred. Similarly for system identification, it is often appropriate to first learn a qualitative structural model, and then parameterize this to form an ODE.

1.3 Modelling metabolism

In this paper we focus on models of metabolism: the interaction of small molecules with enzymes (the domain of classical biochemistry). Such models are arguably the best established in biology. The core biochemical pathways are now known and the KEGG database (Ogata *et al*., 1999) includes ∼11 000 metabolites involved in ∼5500 reactions. However, much remains to be done: only a tiny fraction of the enzymes involving these core metabolites have been quantitatively characterized and an enormous amount of qualitative biochemistry has still to be discovered, e.g. there are estimated to be up to 200 000 distinct metabolites in the plant kingdom alone (Fiehn, 2001). The new science of metabolomics has opened up the possibility of experimentally measuring the metabolites in cells in unprecedented breadth and detail (Fiehn, 2001). The most exciting use of such metabolomics data is to identify novel biochemical pathways.

Two main formalisms have been applied to modelling metabolism: logical/graph-based (LG) models, and ODE. In LG models, pathways are represented as logical relations between enzymes, substrates and products. The pioneering work in this area was by Karp *et al*. (1996) on *Escherichia coli*—EcoCYC. A particularly useful LG system is KEGG (Ogata *et al*., 1999). With LG models it is possible to simulate metabolism at the genome scale, but only with low fidelity. Typical queries of a LG system are: 'What is the most connected metabolite?', 'Is it possible to synthesize a particular compound?', etc. Such models can be used to guide a biological experiment. For example, we have used such models with the robot scientist methodology to automate cycles of experimentation (King *et al*., 2004).

The standard way to model (non-spatial) physical systems is using ODEs and they have been successfully applied to simulating metabolic systems (Bakker *et al*., 1997; Eisenthal and Cornish-Bowden, 1998). Specialist programs are now available for modelling cellular ODE models, such as Gepasi (Mendes, 1997), Dbsolve (Goryanin *et al*., 1999) and E-Cell (Tomita *et al*., 1999). Despite the success of using ODEs to model pathways they have important drawbacks: they require a large number of kinetic parameters to be accurately known, and it is generally impossible to determine all the feasible qualitative states of the model. It is possible to apply ODE based models to whole genome systems. However to achieve this, strong assumptions need to be made about the system (e.g. optimality). Using this approach interesting experimental predictions are beginning to be made (Famili *et al*., 2003).

We propose an intermediate Qualitative reasoning (QR) level of simulation between the LG and ODE models. This QR formalism allows us to immediately form dynamic models from LG information without the need to quantitatively characterize the reactions. This is possible because our QR formalism is an abstraction of ODEs which enables the behaviour of many ODEs, with different functional forms and parameters, to be captured in a single QR model. Thus, we can apply QR modelling with confidence to many situations where ODE modelling would be inappropriate—because the necessary kinetic parameters are unknown. Considering again the example of simulating glycolysis, conclusions drawn from a QR model of glycolysis are generally true for any organism with the same set of reactions, while with an ODE model, the conclusions drawn are specific to the organisms from which the enzymatic kinetic parameters were taken. The downside of the increased applicability of QR is that the predictions made are no longer deterministic, and therefore a number of possible behaviours are produced. The first work on applying QR to the simulation of biological processes was done by Heidtke and Schulze-Kremer (1998a). They applied QR to modelling glycolysis, and then extended the work by simulating the genetic network of the lambda phage (Heidtke and Schulze-Kremer, 1998b). In our paper we extend their simulation approach with the use of metabolic components and show how QR can also be used for model identification.

Compared to the work on simulating metabolism, relatively little research has been done on model identification (LG or ODE). Perhaps the most notable work on identification is that of Arkin *et al*. (1997), who identified an LG model of the reactions in part of glycolysis from experimental data. The work of Reiser *et al*. (2001) presents a unified logical approach to simulation (deduction) and system identification (induction and abduction) in LG models. The identification of ODE models directly from data is a known hard problem (Ljung, 1999). This is especially so if intermediate variables are required in the model. An intermediate variable is a variable that is not observed but interacts with the observed variables. In real world problems it can rarely be ensured that all relevant variables are observed (measured). An interesting recent approach to identifying metabolic ODE models is that of Koza *et al*. (2001) who recast the cellular system as an electrical circuit identification problem, and used genetic programming to search through the space of possible models.

As with model simulation, for model identification QR is intermediate between LG and ODE modelling. This means that if we wish to identify an ODE model from the observations of levels of metabolites, the natural approach is to first learn the LG structure, then form a QR model with this structure and finally parameterize this QR model to an ODE. The parameterization problem is the simplest step, as there already exist many methods of fitting parameters to ODE models (Ljung, 1999).

The identification of QR models from examples is a challenge for current machine learning methods because the problem is relational (requires a first-order learning algorithm); the search space is very large (even for a relational problem); and the data is positive

only (when identifying a system, nature provides only examples of states of the system and not examples the system cannot be in). The most appropriate form of machine learning for QR identification is inductive logic programming (ILP), as QR models can be represented naturally in the form of simplified Prolog programs. In ILP, background knowledge, examples and theories are all described in logic programs (e.g. Prolog). ILP systems learn (induce) logical theories (programs) from examples by searching through a space of possible solutions (Mitchell, 1997). There is a growing literature on learning QR models using ILP (Bratko and Muggleton, 1991; Say and Kuru, 1996; Hau and Coiera, 1997). We have earlier developed the Qoph algorithm, which is a general purpose system identification system for QR models (Coghill *et al*., 2002, 2004).

The advantages of identifying QR models versus ODE models are:

- Machine Learning is generally based on search through a space (defined by a language) of possible solutions (Mitchell, 1997). The discrete state space of qualitative models is smaller than the space of ODE models.
- One of the main problems in numerical methods of system identification is parameter estimation. By its very nature qualitative reasoning does not have this problem because there is no need for the parameters. This reduces the workload of the inductive system by allowing it to concentrate on inducing just the structure of the model. It may therefore be possible to learn a qualitative model with less data than a numerical one, e.g. with fewer time steps in a time series.

2 METHODS

2.1 Qualitative modelling

Qualitative reasoning is a method of reasoning about the structure and behaviour of systems that are incompletely known. It was originally devised as a means of enabling AI systems to reason about the world from first principles rather than relying on heuristic rules (Hayes, 1979). In QR, incompleteness is dealt with by lowering the precision of the system variables to focus only on the qualitative differences in a variable's values (which in the most abstract case will be its sign). These qualitative values are formed by discretization of the real number line. Therefore, QR can be seen as a step towards developing a quantitative model, as it forms the abstract structure of the model, which can then be parameterized to form a quantitative model. By eliminating unnecessary detail, QR models allow the user to focus on the essentials of the model and to extract quickly the required understanding of the system being modelled.

Qualitative modelling has been utilized in a number of different application domains, for example diagnosis, training and control (Weld and de Kleer, 1990). In many ways QR models are similar to standard biological modelling approaches such as signal transduction or metabolic system diagrams. This is to be expected, for in many biological systems the key point of interest is what will happen if a variable increases, decreases or remains unchanged, not precisely how much it changes.

We use the QR system QSIM, a constraint based qualitative simulation algorithm (Kuipers, 1994). QSIM is the most highly developed constraint based QR system. In QSIM each model consists of a set of variables linked together via a set of constraints, called a qualitative differential equation. Each variable consists of a ⟨qmag, qdir⟩ pair. Where qmag is the qualitative magnitude of the variable and has a quantity space of varying resolution and consisting of alternating points (called landmark values) and intervals. The value qdir is the qualitative rate of change of the variable, which has a fixed, three-valued resolution [the three quantities being ↑ (for increasing), ↓ (for decreasing) and 0 (for steady)]. There are several kinds of constraints that can appear in a QSIM model: predicates representing the usual algebraic operations of addition, multiplication and sign inversion, and a derivative predicate stating that one variable is the derivative of another. Incompleteness in the knowledge of the model is captured by the monotonic function constraint (M±) between two variables, which declares that one variable monotonically increases (+) or decreases (−) with respect to another variable.

QSIM begins simulation from a given initial state. From this initial state a set of transitions are developed for each variable of the system—the qualitative equivalent of integration over time. In the behaviour tree that results from the simulation process, the times associated with the states in the tree form an ordered set of points and intervals. This transition phase is performed for each variable individually, as if they are independent of each other. The basis for this is that, because each variable is represented by a magnitude/derivative pair, it is effectively an abstraction of a first order approximation of the Taylor series describing the variable function. Thus a transition to the next qualitative value is based on Euler integration. After the transition rules have been applied to the variables, each variable will have associated with it a set of possible values that it could take in the next time point/interval. The mathematical foundation of the transition rules are the intermediate value theorem and the mean value theorem. Having generated all the possible values for the system variables in accordance with the transition rules, each of these values must be tested against the constraints of the system.

The advantage of QR simulation versus ODEs are:

- *Ease of understanding*: By eliminating unnecessary details, qualitative models allow the user to focus on the essentials of the model and to extract quickly the required understanding of the system being modelled. Comparison of equivalent QR and ODE models shows that QR has less information, and is therefore easier to understand. Of course, QR does not solve all comprehensibility problems and a complex QR model is difficult to understand.
- *Finite nature*: Unlike many real-valued numerical models, a qualitative model can only be in one of a finite number of states. This set of states is the model's complete envisionment.
- *Error Reduction*: All data gathered by physical measurement contain a degree of error due to the measuring process on top of this is added human error. Qualitative modelling can overcome or reduce many of these problems by considering the data's qualitative aspects alone.

2.2 Metabolic components

To apply qualitative modelling methodology to biological pathways of the size of glycolysis would require us to simulate and identify models with ∼100 algebraic primitives. Such large systems are awkward to deal with using a general QSIM simulator and would probably be impossible to identify using general system identification methods—due to their computational complexity. We have therefore used specific domain background knowledge to split the problem into manageable units (a heuristic divide-and-conquer strategy). In the case of modelling metabolic pathways there are essentially only two types of objects: metabolites and enzymes. We have therefore designed metabolic components (MCs) to model these to allow us to efficiently simulate and identify metabolic models.

The concentrations of metabolites vary over time as they are synthesized or utilized by enzymatically catalysed reactions. As a result, their concentration at time t is a function of their concentration at the previous time point or interval, and the amount that they are used or created by various enzyme reactions. This can be expressed as a simple summation in QSIM. The qualitative equation for the metabolite components (Fig. 2) is therefore,

$$\mathrm{d}M/\mathrm{d}t = \sum \mathrm{flow}_i .$$

When modelling enzymes, each enzyme is assumed to have ≤2 substrates and ≤2 products. If there are two substrates or products these are considered to form a substrate or product complex, such that the amount of the complex is proportional to the amount of the substrates or products multiplied together.

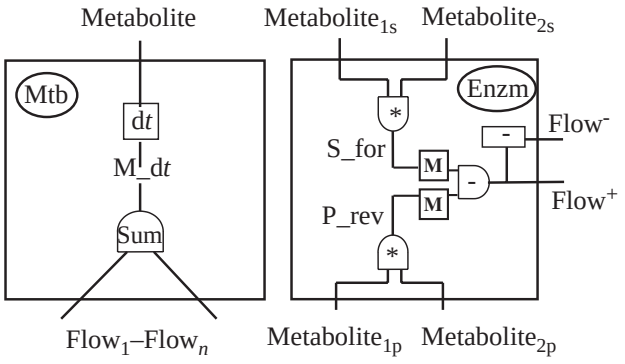

Fig. 2. The QSIM representations of the two MCs in metabolic pathways: metabolites (Mtb) and enzymes (Enzm). In the metabolite component, Flow_1–Flow_n is the list of the flows of synthesis and utilization of the metabolite (these flows come from the enzyme MCs); M_dt is the sum of Flow_1–Flow_n; and the concentration of the metabolite is related to the sum M_dt by a qualitative derivative relation. In the enzyme component, Metabolite_{1s} and Metabolite_{2s} are the levels of substrates; Metabolite_{1p} and Metabolite_{2p} are the products (the levels of metabolites come from the metabolite MCs); S_forw is the qualitative multiplication of Metabolite_{1s} and Metabolite_{2s}; P_rev is the qualitative multiplication of Metabolite_{1p} and Metabolite_{2p}; Flow^+ is the overall flow through the reaction (Flow^- is its negation which is required for the metabolite component).

This qualitatively models the probability that the substrates (or products) will collide with the enzyme with sufficient timeliness to be catalysed into the product complex (or substrate complex). The substrate complex is converted into the product complex, which then dissociates into the product metabolites and vice versa. The overall flow through the enzyme is the amount of substrate complex formed minus the amount of product complex formed. The qualitative equation for the enzyme components (Fig. 2) is

$$\text{Flow} = \text{M}^+(\Pi_{\text{Substrates}}) - \text{M}^+(\Pi_{\text{Products}})$$

This is an abstraction of standard kinetic equations (Cleland, 1963) and is an expression of the collision probabilities of the metabolites in the reaction. A key point to note is that this enzyme component is an abstraction of enzymes with different rate constants. The metabolite and enzyme components are dynamic and they do not simply model steady-state conditions. However, note that we assume, for simplicity, that enzymes are taken to exist in constant amounts—this is also assumed in most ODE modelling.

The above qualitative abstraction of enzymes and metabolites could be criticized in that it does not explicitly involve the interaction of the substrates and products with the enzyme. This was a deliberate pragmatic decision. QSIM can model ODEs of arbitrary complexity, and we could have modelled the interaction with the enzyme at any level of detail. However, in initial experiments using a more detailed model, we concluded that most of the states produced were transients of relatively little biological relevance. These states would be hard to observe in practice, and would have greatly complicated the model identification; so we decided to settle on a simpler model. The MC enzyme is not the 'mass action law': it is qualitative, and has no assumption of equilibrium. Note that any model of an enzymatic reaction short of the full quantum mechanical description is an abstraction (and even that does not properly model mass).

3 RESULTS

As a test case for biochemical pathway simulation and identification we have chosen glycolysis as it is probably the best understood of all pathways.

3.1 Simulating glycolysis using QR

In our qualitative model of glycolysis 15 metabolites are involved (Fig. 3): pyruvate (pyr), glucose (glc), phosphoenolpyruvate (pep), fructose 6-phosphate (f6p), glucose 6-phosphate (g6p), dihydroxyacetone phosphate (dhap), 3-phosphoglycerate (3pg), 1,3-bisphosphoglycerate (13BP), fructose 1,6-biphosphate (f16bp), 2-phosphoglycerate (2pg), glyceraldehyde 3-phosphate (g3p), ADP, ATP, NAD, and NADH. We have not included H^+, H_2O, or Orthophosphate as they are assumed to be ubiquitous—they are difficult to include because of the ≤ 2 substrate/product restriction.

The qualitative state of glycolysis is defined by the set of qualitative states of the 15 metabolites. Figure 4A details one such state of glycolysis. To understand this state consider the qualitative state of NAD: [nad, NAD_l : $0 \ldots \infty/\downarrow$, NAD_f:$-\infty \ldots 0/\downarrow$]. The meaning of this is that the level of NAD (NAD_l) is positive ($0 \ldots \infty$) and decreasing ($\downarrow$), and the flow into NAD (NAD_f) is negative ($-\infty \ldots 0$) and decreasing ($\downarrow$). Similar meanings apply to the other 14 metabolites. Note, the semantics of a metabolic level means that it must be between $0 \ldots \infty$; it cannot be negative, and the 0 state is uninteresting.

An important reason why it is interesting to consider such qualitative states is that it is experimentally much easier to measure qualitative metabolic states than quantitative ones. This is because the experiment is required to produce less information and therefore can be more robust. For example, it is easier to determine that the level of a metabolite is increasing than to determine by exactly how much it is increasing.

Figure 5 shows the glycolysis model in the actual logic-programming format used (with some syntactic sugar). The head of the model is a possible state of glycolysis, as described above. These states are constrained from being in any qualitative state by the enzyme and metabolite MCs in the body of the model. These MCs are implemented by two predicates, 'metabolite' and 'enzyme'. These correspond to the MCs described in the methods. For example:

- The metabolite MC metabolite (NAD_l, NAD_f, [$Enz6_f$, $-$]) states that the level (NAD_l) and flow (NAD_f) of the metabolite NAD is controlled by flow through the single enzyme number 6 ($Enz6_f$: Glyceraldehyde 3-phosphate dehydrogenase).
- The enzyme MC enzyme ([[$G3P_l$, NAD_l], [$13BP_l$, $NADH_l$]], $Enz6_f$) states that the flow through enzyme 6 ($Enz6_f$) affects the levels of the reactants ($G3P_l$, NAD_l) and the products ($13BP_l$, $NADH_l$) in opposite directions.

The model consists of 10 metabolite MCs and 15 metabolite MCs. The reactions can be constrained to be reversible or non-reversible. Comparing Figures 4 and 5, there is a one-to-one mapping. As the functional form and parameters of the model have been abstracted out, the model is simpler than an ODE model. Arbitrarily, complex metabolic pathways can be formed in this way—limited only by the computational limits of the Prolog compiler.

The qualitative glycolysis model can be used in three distinct ways:

- with a given initial state, in conjunction with a QSIM simulator, to simulate glycolysis qualitatively;
- to generate all possible qualitative states of glycolysis—the complete envisionment;

1. glucose + ATP ⇒ glucose 6-phosphate + ADP (*Hexokinase*)
2. glucose 6-phosphate ⇒ fructose 6-phosphate (*Phosphoglucose isomerase*)
3. fructose 6-phosphate + ATP ⇒ fructose 1,6-biphosphate + ADP (*Phosphofructokinase*)
4. fructose 1,6-biphosphate ⇒ dihydroxyacetone phosphate + glyceraldehyde 3-phosphate (*Aldolase*)
5. dihydroxyacetone phosphate ⇒ glyceraldehyde 3-phosphate (*Triose phosphate isomerase*)
6. glyceraldehyde 3-phosphate + NAD ⇒1,3-bisphosphoglycerate + NADH (*Glyceraldehyde 3-phosphate dehydrogenase*)
7. 1,3-bisphosphoglycerate+ ADP ⇒ 3-phosphoglycerate + ATP (*Phosphoglycerate kinase*)
8. 3-phosphoglycerate ⇒2-phosglycerate (*Phosphoglycerate mutase*)
9. 2-phosphoglycerate ⇒ phospoenolpyruvate (*Enolase*)
10. phospoenolpyruvate + ADP ⇒ pyruvate + ATP (*Pyruvate kinase*)

Fig. 3. The reactions included in our qualitative model of glycolysis. The reactions that consume ATP and NADH are not explicitly included.

state([
[nad, $NAD_{l:}0...\infty/\downarrow$, $NAD_{f:}-\infty...0/\downarrow$],
[nadh, $NADH_{l:}0...\infty/\uparrow$, $NADH_{f:}0...\infty/\uparrow$],
[atp, $ATP_{l:}0...\infty/\downarrow$, $ATP_{f:}-\infty...0/\downarrow$],
[adp, $ADP_{l:}0...\infty/\downarrow$, $ADP_{f:}-\infty...0/\downarrow$],
[pyr, $PYR_{l:}0...\infty/\uparrow$, $PYR_{f:}0...\infty/\downarrow$],
[glc, $GLC_{l:}0...\infty/\downarrow$, $GLC_{f:}-\infty...0/\uparrow$],
[pep, $PEP_{l:}0...\infty/\downarrow$, $PEP_{f:}-\infty...0/\downarrow$],
[f6p, $F6P_{l:}0...\infty/\downarrow$, $F6P_{f:}-\infty...0/\downarrow$],
[g6p, $G6P_{l:}0...\infty/\downarrow$, $G6P_{f:}-\infty...0/\downarrow$],
[dhap, $DHAP_{l:}0...\infty/\downarrow$, $DHAP_{f:}-\infty...0/\downarrow$],
[3pg, $3PG_{l:}0...\infty/\uparrow$, $3PG1_{f:}0...\infty/0$],
[13bp, $13BP_{l:}0...\infty/0$, $13BP_{f:}0/\uparrow$],
[f16bp, $F16BP_{l:}0...\infty/\uparrow$, $F16BP_{f:}0...\infty/\downarrow$],
[2pg, $2PG_{l:}0...\infty/\downarrow$, $2PG_{f:}-\infty...0/\downarrow$],
[g3p, $G3P_{l:}0...\infty/\uparrow$, $G3P_{f:}0...\infty/\uparrow$]]).
A

[nad, $NAD_{l:}0...\infty/\uparrow$, $NAD_{f:}\ 0...\infty/\uparrow$],
B

Fig. 4. Example glycolysis system states. In (**A**) a possible glycolysis system state is shown. If the flow and level of NAD is increasing, as shown in (**B**), the state can no longer be generated by a non-reversible version of glycolysis. Therefore, if such a state is observed then the non-reversible model of glycolysis is wrong (incomplete or incorrect).

- as an 'Oracle' to test whether glycolysis could be in any specific qualitative state.

Perhaps the most interesting use of qualitative models is as Oracles. Figure 4 shows an example of a state that glycolysis can be in (A) and one that it cannot (B). Two possible ways to change the qualitative glycolysis model to account for the production of state B are: to reverse the direction of reaction 6 (Glyceraldehyde 3-phosphate dehydrogenase), or to hypothesize a reaction that forms NAD. If you reverse the direction of reaction 6, then state A can no longer be produced, but state B can be produced. The model with a reaction that forms NAD is able to produce both states A and B. This opens up the possibility of an experiment to distinguish between the two possibilities.

The use of particular states to test models suggests a general mechanism of inferring from observations of system states the system that produced the states. This process is known as system identification.

3.2 System identification of glycolysis using LG and QR modelling

The specific system identification tasks we were interested in were: given observation of the metabolites involved in a pathway, how much can be inferred about that pathway; and given qualitative observations of metabolic states, how much can be inferred. Our methodology is described in Figure 6, where we describe two separate ways of identifying biochemical pathways. We make the following assumptions:

- The data are sparse and not necessarily measured as part of a continuous time series—only three consecutive time steps are required. This assumption is realistic given current experimental limitations in metabolomics. This rules out the possibility of numerical system identification approaches.
- Only metabolites of known structure are involved in the model. This is the strongest assumption we make. Even given the rapid advance of metabolomics (NMR, mass-spectroscopy, etc.), it is not currently realistic to assume that all the relevant metabolites in a pathway are observed and their structure determined.
- Only metabolites of known structure are involved in a particular pathway. This is a restriction because current metabolomics technology can observe more compounds than they can be structurally identified (a heuristic constraint).
- All reactions involve at most three substrates and three products (a heuristic constraint).
- For the qualitative states: we can measure the direction in the change of metabolite level. This requires sampling the level at least three times in succession.

3.2.1 The use of LG constraints to examine reactions We first considered the LG nature of the problem. The specific domain of metabolism imposes strong constraints on possible LG models. We used these heuristic constraints in the following way:

(1) Chemical reactions conserve matter and atom type (Valdes-Perez, 1994). For glycolysis we generated all possible ways of combining the 18 metabolites to form matter and atom type balance reactions ($\leq$3 reactants $\leq$3 products). This produced 172 possible reactions where the substrates balanced the products in the number and type of each element. (Note

```
state([[atp,ATP_l,ATP_f],          [adp,ADP_l,ADP_f],      [nad,NAD_l,NAD_f],
       [nadh,NADH_l,NADH_f],       [pyr,PYR_l,PYR_l],      [glc,GLC_l,GLC_f],
       [pep,PEP_l,PEP_f],          [f6p,F6P_l,F6P_f],      [g6p,G6P_l,G6P_f],
       [dhap,DHAP_l,DHAP_f],       [3pg,3PG_l,3PG_f],      [13bp,13BP_l,13BP_f],
       [f16bp,F16BP_l4,F16BP_f4],  [2pg,2PG_l,2PG_f],      [g3p,G3P_l,G3P_f]])   ←
    enzyme([[GLC_l,ATP_l],[G6P_l,ADP_l]],Enz1_f) ∧
    enzyme([[G6P_l],[F6P_l]],Enz2_f) ∧
    enzyme([[F6P_l,ATP_l],[F16BP_l4,ADP_l]],Enz3_f)∧
    enzyme([[F16BP_l4],[G3P_l,DHAP_l]],Enz4_f) ∧
    enzyme([[DHAP_l],[G3P_l]],Enz5_f) ∧
    enzyme([[G3P_l,NAD_l],[13BP_l,NADH_l]],Enz6_f)
    enzyme([[13BP_l,ADP_l],[3PG_l,ATP_l]],Enz7_f) ∧
    enzyme([[3PG_l],[2PG_l]],Enz8_f) ∧
    enzyme([[2PG_l],[PEP_l]],Enz9_f) ∧
    enzyme([[PEP_l,ADP_l],[PYR_l,ATP_l]],Enz10_f) ∧
    metabolite(ATP_l,ATP_f,[[Enz10_f,+],[Enz7_f,+],[Enz1_f,-],[Enz3_f,-]])∧
    metabolite(ADP_l,ADP_f,[[Enz1_f,+],[Enz3_f,+],[Enz10_f,-],[Enz7_f,-]])∧
    metabolite(NAD_l,NAD_f,[[Enz6_f,-]]) ∧
    metabolite(NADH_l,NADH_f,[[Enz6_f,+]]) ∧
    metabolite(PYR_l,PYR_l,[[Enz10_f,+]]) ∧
    metabolite(GLC_l,GLC_f,[[Enz1_f,-]]) ∧
    metabolite(PEP_l,PEP_f,[[Enz9_f,+],[Enz10_f,-]]) ∧
    metabolite(F6P_l,F6P_f,[[Enz2_f,+],[Enz3_f,-]]) ∧
    metabolite(G6P_l,G6P_f,[[Enz1_f,+],[Enz2_f,-]]) ∧
    metabolite(DHAP_l,DHAP_f,[[Enz4_f,+],[Enz5_f,-]]) ∧
    metabolite(3PG_l,3PG_f,[[Enz7_f,+],[Enz8_f,-]]) ∧
    metabolite(13BP_l,13BP_f,[[Enz6_f,+],[Enz7_f,-]]) ∧
    metabolite(F16BP_l4,F16BP_f4,[[Enz3_f,+],[Enz4_f,-]]) ∧
    metabolite(2PG_l,2PG_f,[[Enz8_f,+],[Enz9_f,-]] ∧
    metabolite(G3P_l,G3P_f,[[Enz5_f,+],[Enz4_f,+],[Enz6_f,-]]).
```

Fig. 5. The qualitative model of glycolysis in Prolog form. The head of the model is the observed state of system, this consists of 15 metabolite levels (subscript l) and flows (subscript f). Lower-case names are constants and upper-case variables. ← is the reverse logical implication symbol, i.e. A ← B means if B then A. The body of the model defines the constraints on the system. The enzyme constraints relate the levels of substrates and products in a reaction with the flow through the enzyme. The metabolite constraints relate the level and flow of metabolites with the flow through enzymes that produce (+) and consume (−) them. ∧ is logical 'and'. Note that neither the order in the model of the metabolite HLCs, nor enzyme HLCs is important.

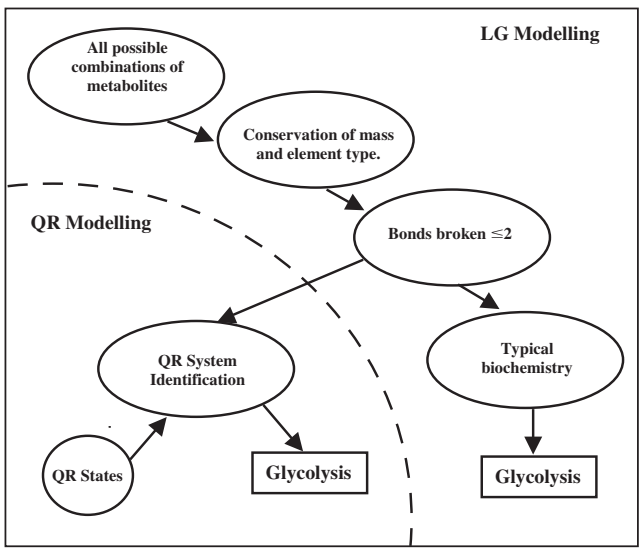

Fig. 6. Our system identification methodology. It is a combination of LG and QR heuristic based system identification.

that this number does not include any identity reactions, A ⇔ A, or reactions which are combinations of simpler reactions, i.e. if the reactions A ⇔ C and B ⇔ D existed, the reaction A + B ⇔ C + D would not be counted.) The number 172 compares well with the ∼2 300 000 possible reactions which would naively be possible.

(2) Typical biochemical reactions only make/break a few bonds and cannot arbitrarily rearrange atoms to make new compounds. For glycolysis we examined all 172 possible reactions to test their chemical plausibility. A reaction was considered plausible if it broke one bond per reactant. This analysis was done originally by hand and we have subsequently developed a general computer program that can automate this task. Of the 172 balanced reactions 18 were considered chemically plausible. Figure 7 shows the list of these reactions that are not in glycolysis, and examples of infeasible reactions that were discarded.

(3) Biochemical reactions follow only a limited number of types of organic reaction: phosphoryl transfer, phosphoryl shift, isomerization, dehydration, aldol cleavage, etc. (Stryer, 1996). For glycolysis, we examined the 18 reactions to see if they

Decoy reactions that are chemically feasible
1,3-bisphosphoglycerate + H_2O → 3-phosphoglycerate + P_i
1,3-bisphosphoglycerate + glucose → 3-phosphoglycerate + glucose 6-phosphate
fructose 1,6-biphosphate + H_2O → fructose 6-phosphate + P_i
fructose 1,6-biphosphate + pyruvate → fructose 6-phosphate + phospoenolpyruvate
fructose 1,6-biphosphate + glucose → glucose 6-phosphate + fructose 6-phosphate
fructose 1,6-biphosphate + 3-phosphoglycerate → 1,3-bisphosphoglycerate + fructose 6-phosphate
glucose 6-phosphate + H_2O → glucose + P_i
glucose 6-phosphate + pyruvate → phospoenolpyruvate + glucose

Example reactions that were considered infeasible
fructose 6-phosphate + H_2O → glucose + P_i
2-phosphoglycerate + fructose 1,6-biphosphate → 1,3-bisphosphoglycerate + fructose 6-phosphate
phospoenolpyruvate + glucose + P_i → 2-phosphoglycerate + glucose 6-phosphate
glucose 6-phosphate + pyruvate + P_i → 1,3-bisphosphoglycerate + glucose

Fig. 7. Plausible and implausible biochemical reactions. The upper list of eight plausible reactions (low number of bonds broken) were used as decoys with the ten reactions in glycolysis. The four implausible reactions (high number of bonds broken) are taken from the 172 reactions that have balanced in element type. The QR problem is to distinguish between the correct model and ones containing plausible reactions.

were 'typical of biochemistry'. Of the 18 reactions, only the 10 actual glycolysis reactions were typical of biochemistry.

3.2.2 The use of QR constraints to examine models In the above methodology, the final step is the hardest to justify. We therefore wished to investigate the use of machine learning to identify the correct glycolysis model from the decoy reactions. It is important to note that in doing this, we are examining whole models, not individual reactions as we did with LG constraints.

A number of general model heuristic constraints are applicable to the problem (Coghill *et al*., 2002a, 2004):

- *Accuracy*: The correct model should be able to produce all of the observed states.
- *Parsimony*: Smaller models are favoured over bigger ones. The rationale for this use of Ockham's razor is, as there are fewer small models, the chance of one of them fitting the data is less.
- *Non-disjoint*: This constraint ensures that the model is unified and not two or more disjoint model parts. The expectation is that if a set of measurements is being made within a particular context, they will emanate from the same system and not from two or more separate systems. Of course, one may be mistaken in this assumption. Therefore, models filtered under this heuristic can be cached and revisited if no suitable models have been found.

We used a simple 'generate and test' approach to learning. We did not explore the possibility of using general learning search heuristics (refinement operators) to move through the search space—these have been extensively studied in ILP. However, these would have complicated the methodology and it is unclear whether they would have added much beyond the heuristics we were already using, as the computational limiting step is the cover test (Section 4.1). For the first computational experiment we used the ten reactions of glycolysis and the eight decoy reactions that were considered chemically feasible (Fig. 7). All these reactions, in the absence of evidence to the contrary, are considered to be irreversible. We first generated all possible ways of combining the 18 reactions which connected all the 15 main substrates in glycolysis (models are non-disjoint). This generated 27 254 possible models with $\leq$10 reactions—it was not necessary to look for models with more reactions than that of the target (parsimony), as the models can be generated in size order. The smallest number of reactions necessary to include all 15 metabolites was of size 5. All of the 27 254 models involved the reaction, glyceraldehyde 3-phosphate + NAD $\Leftrightarrow$ 1,3-bisphosphoglycerate + NADH (reaction 6); so we could immediately conclude that this reaction occurred in glycolysis.

We used qualitative states of glycolysis with our QR simulator (in a pseudorandom manner) to test these models. Note that we did not use a numerical simulator. The problem of quantitative to qualitative conversion is dealt with in the discussion (Section 4.1) and examined in our previous work (Coghill *et al*., 2002a, 2004). The 27 254 possible models were then tested against these states and if a model could not generate a particular state it was removed from consideration (accuracy constraint). In real wet experiments there is always the possibility of noise. In this paper we have ignored the problem of experimental noise as this has been dealt with in our previous work (Coghill *et al*., 2002, 2004).

Note that the flows of the metabolites through each enzyme are not observed—they are intermediate variables. All that we observe are the overall levels and flows of the metabolites. This makes the system identification task much harder.

The main difficulty with this generate-and-test approach is that it can be very computationally expensive to test if a model can generate a particular state. The task is one of deductive inference and for which there is no generally efficient solution. For efficiency, we used the fast YAP Prolog compiler. We also formed compiled down versions of the enzyme and metabolites MCs (input/output look-up tables), and compiled down parts of QSIM. The result was that for some states and models it was possible to show that a state could or could not be formed in a fraction of a second, but for other states and models the computation took days. Again for efficiency, we adopted a resource allocation method that employed increasingly computationally expensive tests, i.e. forming filter tests with exponentially increasing numbers of example states. One approach we did not use, but plan to exploit in the future is the fast subsumption method of Maloberti and Sebag (2001) which exploits a transforms of the problem into one of general constraint search.

After several months of computed time on a 65 node Beowulf cluster we reduced the 27 254 possible models to 35 (a $\sim$736-fold

reduction). These models included the target model (glycolysis), and the 34 other models that could not be qualitatively distinguished from it. These reactions form the main core of glycolysis. Examining the 35 models also revealed that the correct model had the least cycles, but we do not know if this is a general phenomenon.

In system identification there are only positive examples available to learn from—we observe example states of the system, but not states that the system cannot be in. In machine learning this is technically known as positive only learning. In our earlier work on the Qoph algorithm (Coghill *et al*., 2002) we have used the approach to 'positive only' learning of Muggleton (1995). The information theoretic idea used is that the true model should cover fewer random states than the alternatives which cover the same true states. We tried to apply this idea to glycolysis. We computationally generated thousands of random states (using a uniform distribution) and tested to see how many of these were covered by the 35 models. Note that this approach does not require any new observations (wet experiments). However, despite the large number of random states generated, we were unable to find any state that was covered by any of the models. We believe this reflects the vast size of the state space, which means that the probability that any model covers any random state is very low. We therefore tested a natural modification of this information theoretic approach. We found that if we produced (random) states from glucogenesis (glycolysis driven in the reverse direction), then the true model of glycolysis covered fewer examples than any of the 34 alternatives and so can be identified as the target model. This approach is related to perturbation based system identification methods and it would be interesting to explore their empirical and theoretical relationships. However, this approach (and the perturbation methods) has the disadvantage over the Muggleton approach of requiring new experimental observations.

4 DISCUSSION

4.1 Quantitative to qualitative states

Our QR approach is designed to simulate and identify qualitative models from qualitative data. It can therefore be directly applied to the many biological systems that are naturally qualitative. Most traditional molecular biology falls into this category.

For systems that are naturally quantitative we believe that QR models can be still be useful (Section 1.5). In such cases there is a need to map from observed quantitative states to qualitative states. In this paper we have generally assumed that the conversion of any quantitative data has already been performed. However, in previous work on QR simulation and identification we have demonstrated a proof of principle method of automatically converting quantitative to qualitative data. This work was based on the standard system science problems of models of u-tubes, cascaded tanks, coupled tanks and damped springs (Coghill *et al*., 2002, 2004). For all four standard problems we demonstrated system identification from both noise-free and noisy numerical data. The numerical data were obtained from numerical simulations of the four physical systems, where each simulation was constructed using the same relations as the qualitative models, with the addition of a parameter for each monotonic function relation. This gave a linear relation between the two variables. More complex, non-linear functions might have been used, but linear functions provided a good approximation of the known behavior of these systems. To convert from quantitative data to qualitative states we adopted the simple approach of numerical differentiation by means of a central difference approach to produce a qualitative state. Note that the use of a minimum of three successive time steps is required to obtain one qualitative state. The data produced was typically noisy, either inherently or through the process of differentiation. We therefore performed smoothing of the first and second derivative using a Blackman filter. In addition, as floating-point values are unlikely to be exactly zero, we have found it advantageous to filter further to eliminate small fluctuations around this value. However, in addition to generating correct qualitative states (true positives) the conversion could produce errors: states generated may not correspond to true states (false positives); and some true states may not be generated (false negatives). We have shown that qualitative system identification is robust to these errors provided key observations are made (Coghill *et al*., 2002, 2004).

4.2 Computational complexity

A major limitation of the systems identification task is the time taken (several months on a Beowulf cluster) to reduce the models from the $\sim$27 000 possible ones generated using chemoinformatic constraints, to the single correct one using the qualitative state constraints. While it would be preferable for this process to be faster, it is important to note that identifying a system with 10 reactions and 15 metabolites from scratch is an extremely hard system identification task. We doubt if any human could achieve it and we believe it would be a 'challenge' for all the system identification methods we are aware of. It is difficult to compare system identification methods and we believe there is a need for competitions such as the CASP protein structure prediction contests, and those run by KDD to compare methods. We note that in practical systems identification in biology, it is far more common to start with a partial model, which is then added to (theory completion) or modified (theory revision).

The computational time of identification is dominated by the time taken to test if a particular model can produce certain observed states: examining $\sim$27 000 models is not unusual for a machine learning program, but it is unusual for a program to take hours to test if individual examples are covered. The slow speed of our identification method is therefore not a problem with our learning method (i.e. how it searches the space of possible models), but is intrinsic to the complex relationship between a model and the states it defines.

Our cover-test method is, in the worst case, exponential in the maximum size of the model. Our empirical average case results are much better than this, but it notoriously hard to analyse average cases, and we do not have a good theoretical understanding of these. Note that our lack of an efficient method, i.e. polynomial algorithm, to determine cover is not because we are using qualitative states. We believe that the inherent difficulty of this task applies to both for quantitative and qualitative models. In some areas of mathematics moving from the discrete to the real domain can simplify problems—this is the basis of much of the power of analysis. However, there is currently little evidence for being the case in cell modelling, and quantitative models would seem to aggravate the problem. As cover tests are essentially deductions the question whether a set of axioms and rules (computer program/model) can output a particular logical sentence (observed state) is in general non-computable. However, in real biological systems, as they are bounded in space and time, non-computability is not a problem, and we expect all system identification methods to struggle with the task.

We believe that the general problem of the computational complexity of determining whether an *in silico* model of a biological system

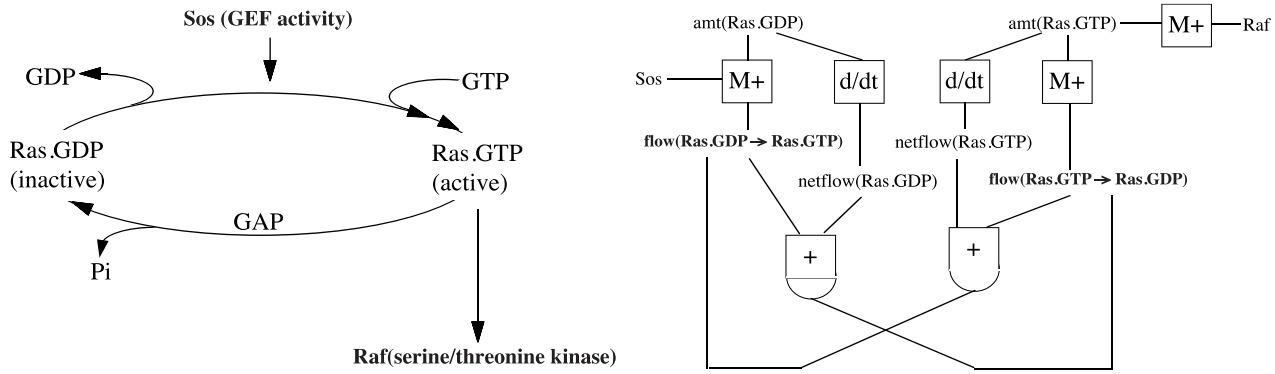

Fig. 8. The Ras.GTP–Ras.GDP cycle in standard biological format and the more precise QR formalism.

can or cannot produce an observed state is a key roadblock in developing Systems Biology. As the size of cells models increase, this problem can only be expected to become more intense. It may be that there will specific solutions for cellular modelling that avoid these complexity issues, but this is not currently clear. What is certainly to be expected is that cellular model simulation will require large-scale computing resources and close collaboration between modellers and experimentalists will be necessary to select experimental states that can be used to test models tractably.

4.3 Extension of qualitative reasoning to other omes

It would be quite straightforward to apply qualitative modelling to simulate and identify gene-network and protein signal-transduction pathway models. Signal transduction pathways would be the easiest to model. In such models the qualitative rate of flows and rate of change of protein concentrations would change. For example, the standard Ras.GTP–Ras.GDP cycle (Fig. 8) can be modelled using standard QSIM (without the need for metabolic components). To model gene-networks would be more complex and would probably require the formation of specific metabolic components for such elements as transcription factors, DNA binding sites, etc. Ultimately, models of biochemical pathways, gene-networks and protein signal-transduction will need to be fully unified. This will at first have to be done qualitatively and QR may provide a suitable framework for this.

ACKNOWLEDGEMENTS

Simon Garrett was funded by BBSRC/EPSRC Bioinformatics Initiative grant BIO10479. The authors would like to thank Stephen Oliver and Douglas Kell for their helpful advice.

REFERENCES

Akutsu,T., Miyano,S. and Kuhara,S. (2000) Inferring qualitative relations in genetic networks and metabolic pathways. *Bioinformatics*, **16**, 727–734.

Alon,U., Surette,M.G., Barkai,N. and Libler,S. (1999) Robustness in bacterial chemotaxis. *Nature*, **397**, 169–171.

Arkin,A., Shen,P. and Ross,J. (1997) A test case of correlation metric construction of a reaction pathway from measurements. *Science*, **277**, 1275–1279.

Bakker,B.M., Michels,P.A.M., Opperdoes,F.R. and Westerhoff,H.V. (1997) Glycolysis in bloodstream from *Trypanasoma brucei* can be understood in terms of the kinetics of the glycotic enzymes. *J. Biol. Chem.*, **272**, 3207–3215.

Bratko,I. and Muggleton,S. (1991) Learning qualitative models of dynamic systems. In *Proceedings of the 8th International Workshop on Machine Learning*, Morgan Kaufmann, pp. 385–388.

Cleland,W.W. (1963) Prediction of inhibition patterns. *Biochim. Biophys. Acta.*, **67**, 173–187.

Coghill,G.M., Garret,S.M. and King,R.D. (2002) Learning qualitative models in the presence of noise. In *Proceedings of the 16th International Workshop on Qualitative Reasoning*, Sitges, Barcelona, Spain.

Coghill,G.M., Garret,S.M. and King,R.D. (2004) Learning qualitative metabolic models. In de Mantaras,R.L. and Saitta,L. (eds), *Proceedings of the 16th European Conference on Artificial Intelligence*, IOS Press, pp. 445–449.

von Dassow,G., Meir,E., Munro,E. and Odell,G.M. (2000) The segment polarity network is a robust developmental module. *Nature*, **406**, 188–192.

Eisenthal,R. and Cornish-Bowden,A. (1998) Prospects for antiparisitic drugs: the case of *Trypanasoma brucei*, the causative agent of African sleeping sickness. *J. Biol. Chem.*, **273**, 5500–5505.

Famili,I., Forster,J., Nielsen,J. and Palsson,B.O. (2003) *Sacchromyces cerevisiae* phenotypes can be predicted by using constraint-based analysis of a genome-scale reconstructed metabolic network. *Proc. Natl Acad. Sci. USA*, **100**, 13134–13139.

Ferrell,J.E. and Machleder,E.M. (1998) The biochemical basis for an all-or-none cell fate switch in *Xenopus* oocytes. *Science*, **280**, 895–898.

Fiehn,O. (2001) Combining genomics, metabolome analysis, and biochemical modelling to understand metabolic networks. *Comp. Funct. Genom.*, **2**, 155–168.

Friedman,N., Linial,M., Nachman,I. and Pe'er,D. (2000) Using Bayesian networks to analyze expression data. *J. Comput. Biol.*, **7**, 601–620.

Goryanin,I., Hodgman,T.C. and Selkov,E. (1999) Mathematical simulation and analysis of cellular metabolism and regulation. *Bioinformatics*, **15**, 749–758.

Hau,D.T. and Coiera,E.W. (1997) Learning qualitative models of dynamic systems. *Machine Learning*, **26**, 177–211.

Hayes,P. (1979) Naïve physics manifesto. In Michie,D. (ed.), *Expert systems in the micro-electronic age*, Edinburgh University Press, Edinburgh, pp. 242–270.

Heidtke,K.R. and Schulze-Kremer,S. (1998a) BioSim—A new qualitative simulation environment for molecular biology. In *Proceedings of the Sixth International Conference on Intelligent Systems for Molecular Biology*, AAAI Press, Menlo Park, pp. 85–94.

Heidtke,K.R. and Schulze-Kremer,S. (1998b) Design and implementation of a qualitative modell of lambda phage infection. *Bioinformatics*, **14**, 81–91.

Karp,P.D., Riley,M., Paley,S.M., Pellegrini-Toole,A. and Krummenacker,M. (1996) EcoCyc: Encyclopedia of *Escherichia coli* genes and metabolism. *Nucleic Acids Res.*, **24**, 32–39.

Kauffman,S.A. (1993) *The Origins of Order, Self-organisation and Selection in Evolution*, Oxford University Press, Oxford, UK.

King,R.D., Whelan,K.E., Jones,F.M., Reiser,P.G.K., Bryant,C.H., Muggleton,S.H., Kell,D.B. and Oliver,S.G. (2004) Functional genomic hypothesis generation and experimentation by a robot scientist. *Nature*, **427**, 247–252.

Koza,J.R., Mydlowec,W., Lanza,G., Yu,J. and Keane,M.A. (2001) Engineering of metabolic pathways from observed data using genetic programming. In *Proceedings of the Pacific Symposium on Biocomputing*, Mauna Lani, Hawaii, pp. 434–445.

Kuipers,B. (1994) *Qualitative Reasoning*, MIT Press, Cambridge, MA.

Ljung,L. (1999) *System Identification*, Prentice-Hall, New-Jersey.

Maloberti,J. and Sebag,M. (2001) Theta-subsumption in a constraint satisfaction perspective. In *Proceedings of the Inductive Logic Programming Conference*, Springer-Verlag, Berlin, pp. 164–178.

Matsuno,H., Doi,A., Nagasaki,M. and Miyano,S. (2000) Hybrid Petri net representation of gene regulatory network. In *Proceedings of the Pacific Symposium on Biocomputing*, Mauna Lani, Hawaii, pp. 41–52.

Mendes,P. (1997) Biochemistry by numbers: simulation of biochemical pathways with Gepasi 3. *Trends Biochem. Sci.*, **22**, 36–363.

Mitchell,T. (1997) *Machine Learning*, McGraw-Hill, New York.

Muggleton,S. (1995) Inverse entailment and Progol. *New Generation Comput.*, **13**, 245–286.

Ogata,H., Goto,S., Fujibuchi,W., Bono,H. and Kanehisa,M. (1999) KEGG: kyoto encyclopedia of genes and genomes. *Nucleic Acids Res.*, **27**, 29–34.

Regev,A., Silverman,W., Shapiro,E. (2001) Representation and simulation of biochemical processes using the π-calculus process algebra. In *Proceedings of the Pacific Symposium on Biocomputing*, Mauna Lani, Hawaii, pp. 459–470.

Reiser,P.K., King,R.D., Kell,D.B., Muggleton,S.H., Bryant,C.H. and Oliver,S.G. (2001) Developing a logical model of yeast metabolism. *Electronic Transactions in Artificial Intelligence*, **5**, 233–244.

Santillan,M. and Mackey,M.C. (2001) Dynamic regulation of the tryptophan operon: a modelling study and comparison with experimental data. *Proc. Natl Acad. Sci. USA*, **98**, 1364–1369.

Say,A.C.C. and Kuru,S. (1996) Qualitative system identification: deriving structure from behaviour. *Artificial Intelligence*, **83**, 75–141.

Somogyi,R. and Sniegoski,C. (1996) Modelling the complexity of genetic networks: understanding mutagenic and pleiotropic regulation. *Complexity*, **1**, 45–63.

Stryer,L. (1996) *Biochemistry*, W.H. Freeman and Company.

Tomita,M., Hashimoto,K., Takahashi,K., Shimuzu,T.S., Matsuzaki,Y., Miyoshi,F., Saito,K., Tanida,S., Yugi,K., Venter,J.C. and Hutchinson,J.C. (1999) E-CELL: software environment for whole-cell simulation. *Bioinformatics*, **15**, 72–84.

Valdes-Perez,R.E. (1994) Heuristics for systematic elucidation of reaction pathways. *J. Chem. Informat. Comput. Sci.*, **34**, 976–983.

Weld,D. and de Kleer,J. (1990) *Readings in Qualitative Reasoning about Physical Systems*, Morgan Kaufmann, Palo Alto.