Algorithms for Reconstructing and Reasoning about Chemical Reaction Networks

Yong Ju Cho

Dissertation document submitted to the faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy
in
Computer Science and Applications

Naren Ramakrishnan, Chair
Yang Cao
David Bevan
T. M. Murali
Anil Vullikanti

Dec 11, 2012
Blacksburg, Virginia

Keywords: Chemical reaction networks, bistability, data mining, time series modeling.

Algorithms for Reconstructing and Reasoning about Chemical Reaction Networks

Yong Ju Cho

(ABSTRACT)

Recent advances in systems biology have uncovered detailed mechanisms of biological processes such as the cell cycle, circadian rhythms, and signaling pathways. These mechanisms are modeled by chemical reaction networks (CRNs) which are typically simulated by converting to ordinary differential equations (ODEs), so that the goal is to closely reproduce the observed quantitative and qualitative behaviors of the modeled process.

This thesis proposes two algorithmic problems related to the construction and comprehension of CRN models. The first problem focuses on reconstructing CRNs from given time series. Given multivariate time course data obtained by perturbing a given CRN, how can we systematically deduce the interconnections between the species of the network? We demonstrate how this problem can be modeled as, first, one of uncovering conditional independence relationships using buffering experiments and, second, of determining the properties of the individual chemical reactions. Experimental results demonstrate the effectiveness of our approach on both synthetic and real CRNs.

The second problem this work focuses on is to aid in network comprehension, i.e., to understand the motifs underlying complex dynamical behaviors of CRNs. Specifically, we focus on bistability—an important dynamical property of a CRN—and propose algorithms to identify the core structures responsible for conferring bistability. The approach we take is to systematically infer the instability causing structures (ICSs) of a CRN and use machine learning techniques to relate properties of the CRN to the presence of such ICSs. This work has the potential to aid in not just network comprehension but also model simplification, by helping reduce the complexity of known bistable systems.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

Current success in systems biology research has helped unravel the complexity of important biological processes [1]. Even a simple organism such as *Escherichia coli* has thousands of genes and active proteins. Detailed mechanisms of this and other organisms are being discovered day by day and are helping comprehend the complexity of processes in these organisms [10].

Construction and simulation of mathematical models [42] is one of the important tools available to study complex processes such as the cell cycle, circadian rhythms, and signaling pathways [2]. Modelers typically begin with a chemical reaction network (CRN), convert them to ordinary differential equations (ODEs), and finally simulate the ODEs to obtain multivariate time series of molecular species. The model is typically adjusted till it closely reproduces the quantitative and qualitative behaviors presented in real data (e.g., phenotypes) [18].

However, the growing complexity of CRN models portends many problems. First, modeling and simulation of large biomolecular systems is quite a challenge. For instance, the molecules and receptors often have vast numbers of binding sites and increases in binding sites typically result in exponential increases in the number of species to consider [10]. The number of parameters to be estimated increases as the size of the model grows. It is non-trivial to obtain accurate parameters for many biological processes; hence, often only rough bounds of such parameters can be estimated. As a result, increasing the size of a model is most likely to introduce more uncertainties in the model [10]. Furthermore, solving such a large model needs vast amount of computational resources. Multiple time scales inherent in chemical reactions also add computational complexities because such differences in time scales introduce stiffness into the model and disrupt stability of numerical solutions [29]. Finally, the growing complexity of models also hinders interpreting experimental results. The model

itself becomes so complex that it does not capture the underlying mechanism intuitively.

In this dissertation, we study two research questions related to the construction and comprehension of CRN models.

1.2 Research Questions

The first question is how to reconstruct a CRN from time series data. This will aid in automatic construction of models in a data-driven manner rather than 'by hand' modeling. In addition, the network reconstruction problem can also be used for network/model reduction. Given a (complex) model, we can generate time series data from it, and mine the resulting data to obtain a potentially simpler network that still retains the essential characteristics of the original model. In Chapter 3, we present an algorithm to reconstruct CRNs from time series data using buffering experiments. Because there are an exponential set of molecule combinations that can be buffered, efficient and effective algorithms are needed that can systematically explore the space of possibilities.

The second research question pertains to network comprehension, specifically to identify the core of a CRN which is responsible for bistability. Bistability is an important functional motif frequently shown in biological processes [45]. Aside from their intrinsic mathematical and chemical significance, bistable CRNs are of particular biological interest because they can retain a 'memory' of past inputs and cellular decisions. Chemical stimuli can trigger a state change from one stable state to another and the current state of the chemical system therefore serves as a memory of this earlier stimulus. Understanding the molecular basis of biochemical switches is hence a building block to comprehending the information processing capabilities of complex signaling pathways. From a database of bistable CRNs [32], we present an algorithmic approach to identify the core structures crucial for conferring bistability.

This dissertation explores the above questions through a combination of mathematical modeling, numerical simulation, and data mining. Our work sheds light into the key functionings of biological systems and the algorithms developed here can be used as building blocks in larger-scope computational modeling and discovery software.

1.3 Organization of this document

Chapter 2 presents an overview of CRN modeling and mathematical formalisms. Chapter 3 introduces a network reconstruction algorithm using buffering experiments and addresses the first research question above. In Chapter 4, the second research question is explored, along with proposed approaches and evaluation methodologies. Finally, Chapter 5 identifies opportunities for further research.

Chapter 2

Background

2.1 A Biochemical Switch and Bistability

The ODE model of a CRN with n species and r reactions is :

$$\frac{dx}{dt} = N\nu(x), \tag{2.1}$$

where $N \in R^{n\times r}$ is a stoichiometric matrix and $\nu \in R^n$is a vector of reaction rates in the system.

A system of ODEs is multistationary iff there is more than one real solution x which satisfy:

$$\frac{dx}{dt} = N\nu(x) = 0, \tag{2.2}$$

and roots of the equation are called steady states. The stability of the fixed points are related with signs of eigenvalues of Jacobian $J = \frac{\partial N\cdot v(x)}{\partial x}$. A fixed point is stable iff the real part of each eigenvalue is negative. Eigenvalues of a Jacobian is acquired from the characteristic equation $det(J - \lambda I) = 0$. A system is bistable iff the system has two stable steady states.

Nonlinear behavior of a dynamic system can be explained as change of non-wandering sets with respect to change of a control parameter in the system. One of the examples is toggle switch like behavior and can be often seen in some biological processes. In [41] , the authors introduce several chemical reaction networks analogous to electrical circuit components. One of the introduced systems is shown in Figure 2.1[41]. The left box in the figure illustrates a system with mutual inhibition between R and E. E is an enzyme which stimulates degradation of R. R also stimulates the reaction $EP \longrightarrow E$, thus forming a mutual inhibition loop. The dynamics of the system is modeled with ODEs as in the following:

$$\frac{dR}{dt} = 0.05S - 0.1R - 0.5E(R)\cdot R$$

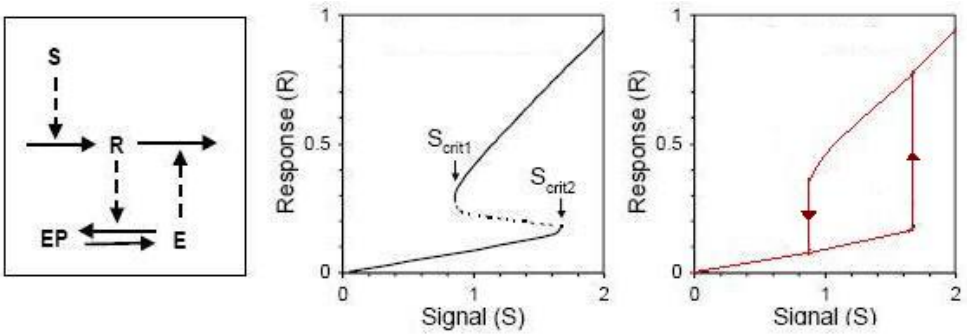


Figure 2.1: An examples of CRN with mutual inhibition loop and bifurcation diagram of the system from [41].

where $E(R) = G(1, 0.2R, 0.5, 0.5)$; G is 'Goldbeter-Koshland function' [16] here. The bifurcation diagram of the system can be seen in the middle box. A point (x, y) on the curves in the diagram represents a fixed point $R = x$ and $S = y$. If the point is on solid part of the curve, the fixed point is locally stable around the point. If the point is on the dashed part, the point is not stable. As shown in the middle box of the figure, the stable fixed points form two separated curves. We can say that the system is in high/low state if the system is in a steady state belong to the upper/lower curve.

Suppose the system is relaxed with some initial condition where $S = 0$. After the system reaches a steady state, S is slowly increased up to S_{crit1}. There is only one stable fixed point in this region so the system stays in the low state. As S is increased from S_{crit1} to S_{crit2}, one more stable fixed point and one saddle is emerged in addition to the existing fixed point. In other words, a saddle node bifurcation takes place at $S = S_{crit1}$. When S reaches to S_{crit2}, a stable fixed point and a saddle annihilates each other, making the system monostable; a saddle node bifurcation happens here too. If S is increased further, R jumps to the global attractor in the upper curve. The jump is irreversible due to local stability of the fixed point. The system stays in the high state while S is decreased to S_{crit1}. If S is decreased further so that $S < S_{crit2}$, the system makes transition to the low state. The right box in the figure shows the hysteresis loop of this system. The response R to the signal S depends on the path it takes before and it can be considered as a biochemical memory switch.

One of real world examples showing the toggle switch-like behavior is cell division cycle of budding yeast. The cell-division cycle is modeled as irreversible transitions between two stable states in [8].

Figure 2.2.[8] show consensus model of cell division cycle in budding yeast. The intuitive understating of the mechanism of cell division cycle is hindered because of the complexity of the model. The authors simplified the mechanisms of cell division cycle as mutual inhibition between B-type cyclins and G1 stabilizers as shown Figure 2.3[8].

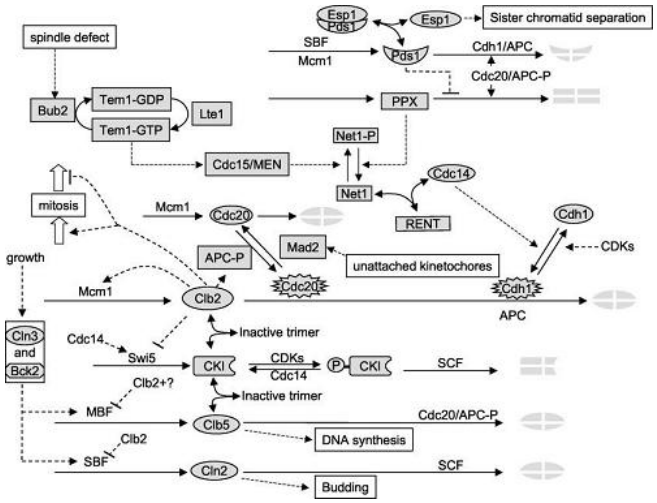


Figure 2.2: Concensus model of cell division cycle of budding yeast from [8]

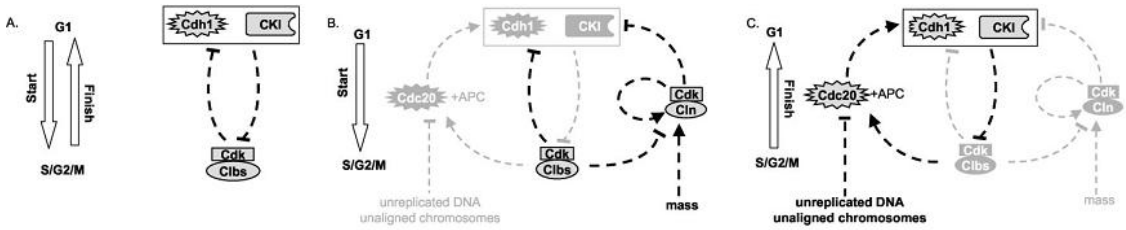


Figure 2.3: Simplified mechanism of cell division cycle in budding yeast from [8]

The transition from G1 phase to S/G2/M phase called start transition and illustrated in Figure 2.3 B. The start transition is triggered by accumulation of Cln3-dependent kinase due to cell mass growth. Similarly, the transition from M phase to G1 phase shown in Figure 2.3C. is called finish transition. The transition is facilitated by accumulation of Cdc20 and APC complex due to completion of chromosome alignment.

The two stable states corresponding to G1 phase and G2/S/M states are qualitatively different with each other and such drastic change can be considered as a form of bifurcations. Cell mass growth and chromosome alignment can be seen as the control signal causing the bifurcation in the above example.

Even though both examples have a mutual inhibition as its core mechanism of the switching, their mathematical model and mechanisms of the transition are quite different. There can be myriad of CRNs having toggle switch-like property with various topologies and parameter ranges. However, bistability is one of necessary conditions of such a biochemical switch.

There are numerical methods finding bistability of a system given parameter range as discussed earlier. Obviously, the problem of finding steady states is equivalent to a root finding problem and can be solved using various numerical root finding methods. Homotopy continuation is one of the methods used in [32]. In [32], the authors also used a simulation based method to find steady states.

2.1.1 Instability causing structure of a CRN

For a linear system $\dot{x} = Ax$ and its Jacobian $J = \frac{\partial Ax}{\partial x}$, $b_{i,j} = J(i,j)$ represents how concentration of species i affects the rate of increase in species j. An directed graph $G_J = (V,E)$ can be defined from J such that V is a set of all independent species and $(i,j) \in E$ iff $\frac{\partial \dot{x_j}}{\partial x_i}$ is nonzero.

There are also some approaches to deduce properties of the system from the coefficients of $det(J-\lambda I) = 0$. A simple cycle in G_J is equivalent to nonzero elements in J whose sequence of row indexes is a cyclic permutation of column indexes [38]. For example, the cycle composed of the set of edges {(1,2), (2,4), (4,3), (3,1)} is corresponding to nonzero elements ($b_{2,1}$, $b_{4,2}$, $b_{3,4}$, $b_{1,3}$) and the sequence of row indexes (2,4,3,1) is a cyclic permutation of the column indexes (1,2,4,3).

Such a cycle is called a *circuit* [38]. A circuit can be represented with product of the associated nonzero elements in Jacobian[38]. The above circuit in the example is represented with $b_{2,1}b_{4,2}b_{3,4}b_{1,3}$. Two circuits are *disjoint* iff there is no vertex that the two circuits share[38]. A *union* of circuits is a set of pairwise disjoint circuit[38]. The size of a circuit is determined by the number of vertices that the circuit covers. The circuit is closely related with coefficients of polynomials in the characteristic equation of the Jacobian. For the characteristic equation

$$\lambda^n + a_1\lambda^{n-1} + a_2\lambda^{n-2} + ... + a_{n-1}\lambda + a_n = 0, \tag{2.3}$$

the coefficient a_i $(1 \leq i \leq n)$ is:

$$\begin{aligned}
a_1 &= (-1)^1 \sum_{i,j}(b_{i,i}) \\
a_2 &= (-1)^2(\sum_{i,j} b_{i,i}b_{j,j} - \sum_{i,j} b_{i,j}b_{j,i}) \\
a_3 &= (-1)^3(\sum_{i,j,k} b_{i,i}b_{j,j}b_{kk} - \sum_{i,j,k} b_{i,i}b_{j,k}b_{k,j} + \sum_{i,j,k} b_{ij}b_{j,k}b_{k,i}) \\
&... \\
a_n &= (-1)^n Det(J)
\end{aligned}$$

[44]. Each nonzero term in a_i represents an union of disjoint circuits up to size i[38]. The coefficients of characteristic equation can be calculated using Bocher's formula such as the following:

$$a_n = -\frac{1}{n}\{a_{n-1}tr(A) + a_{n-2}tr(A^2) + ... + tr(A^n)\}$$

In [19], it is shown that the signs of real parts of all eigenvalues are negative iff all coefficient of the characteristic equations are positive and $H_j > 0$ where $j \geq 2$ and H_j is leading j-principal minor of Hurwitz determinant such as the following:

$$H = \begin{vmatrix} a_1 & a_3 & a_5 & a_7 & ... \\ 1 & a_2 & a_4 & a_6 & ... \\ 0 & a_1 & a_3 & a_5 & ... \\ 0 & 1 & a_2 & a_4 & ... \\ ... & ... & ... & ... & ... \end{vmatrix}$$

This condition is called Routh-Hurwitz stability criterion. the authors introduce a systemic way of detecting reactions which make the system violate the criterion in [44] and call such reactions as Instability Causing Structures (ICS). This algorithm can be applied to the example system in Figure 2.1. The example system consists of $r_1 : \xrightarrow{S} R$, $r_2 : R \xrightarrow{E}$, $r_3 : E \xrightarrow{R} EP$,$r_4 : EP \longrightarrow E$. Since S is the control parameter and $EP + E = EP_0 + E_0 = constant$, the concentration of R and E become state variables of this dynamic system. Then,

$$\frac{dx}{dt} = \begin{bmatrix} v_1(S) - v_2(R,E) \\ -v_3(R,E) + v_4(EP) \end{bmatrix}$$

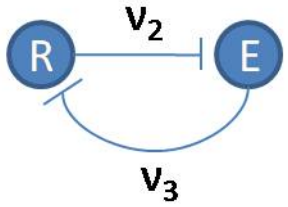


Figure 2.4: Interaction graph of 'toggle switch'

where $x = \begin{bmatrix} R \\ E \end{bmatrix}$ and v_i is the reaction rate function of r_i. Jacobian of this system is

$$J = \begin{bmatrix} -v_{2,R} & -v_{2,E} \\ -v_{3,R} & -v_{4,E} \end{bmatrix}$$

where$v_{i,j} = \frac{\partial v_i}{\partial j}$. Figure 2.4. shows the interaction graph derived from J and clearly showing mutual inhibition between R and E due to r_2 and r_3. The characteristic equation of this system is $\lambda^2 + (v_{2,R} + v_{4,E})\lambda + v_{2,R}v_{4,E} - v_{2,E}v_{3,R}$ so $a_1 = v_{2,R} + v_{4,E}$ and $a_2 = v_{2,R}v_{4,E} - v_{2,E}v_{3,R}$ should be positive to satisfy Routh-Hurwitz stability criterion. In case that the partial dertivatives of the reaction rates to concentration of species are all positive, circuit $v_{2E}v_{3R}$ becomes the instability causing structure in this system and happen to be the mutual inhibition mechanism. Monostability of the system where $S \leq S_{crit1}$ or $S \geq S_{crit2}$ can be explained with deactivation of the mutual inhibition loop in the region; if $S \leq S_{crit1}$, the concentration of R becomes low and turn off r_3, thus breaking the inhibition loop. The loop is deactivated where $S \geq S_{crit2}$ because large R convert entire E to EP, resulting deactivation of r_3. This example shows that there might be a connection between instability causing structures and switching mechanisms of a bistable system.

This explanation on the state transition is very similar with the explanation on the start/finish transition in 2.3. The start and finish transition is explained by deactivation of a feedback loop as in the toggle-switch example. If the relation between ICSs and core mechanisms is identified, it can contribute to automate the process of model reduction of complex bistable systems.

2.2 Model Reduction using Time Scale Analysis

Time scale seperation of chemical reactions in a CRN is very common in biochemical processes[15]. Sets of reactions or lumps of species are associated with different time scales at different point in a biological proces and activeness of dynamics in specific time scale changes in time. The set of reactions or lump of species associated with a time scale is

called time scale mode. The number of active time scale modes can be thought as dimension of the manifold in state space where the system resides. Some biological processes such as cell division cycle consist of alternating excitation and relaxation period. The dimension of manifold increases during the excitation period and the system quickly approaches to slow manifold of lower dimension during the relaxation period. There are approaches exploiting the separation in time scale to reduce complexity of a model[39, 25, 26]. In this section, some of basic concepts of time scale analysis and model reduction approaches based on the time scale seperation are discussed.

2.2.1 Reduction of Chemical Kinetic Models

Simplifying chemical kinetic models is a well known problem in chemical engineering. [29] provides a review on various approaches on this problem. An ODE model of a chemical reaction network can be expressed as in the following:

$$\frac{dx}{dt} = f(x,p), \tag{2.4}$$

$$\tag{2.5}$$

where $x \in R^n$, $p \in R^k$ is the vector of states, and parameters of the system. The states are often equivalent to the independent chemical species in the system. The objective of model reduction is to convert such a model into

$$\frac{d\hat{x}}{dt} = \hat{f}(\hat{x},p), \tag{2.6}$$

$$y = \hat{g}(\hat{x}). \tag{2.7}$$

where $\hat{x} \in R^c$ is the vector of states in the reduced system.

The converted model is simplified by removing redundant state variables and/or combining states variables such that $c < n$ or replacing f with $\hat{f}$ approaximating f.

The goodness of the reduced model is often measured by comparing outputs of the original model and the reduced model. Sum of squared errors is often used as the measure but various qualitative and quantitative measures are used for the comparison.

Quasi Steady State Approximation (QSSA) is one of approaches to reduce a model using time scale separation [29]. Suppose there are two sets of reactions exist for a system and the rates of change of species in one of the sets are much faster than the species in the other set. Let F be the set of indexes of slow species and S is the set of indexes of the slow species. Since the fast species have very small relaxation time compared to the relaxation time of slow species, the fast species are assumed to reach to steady-states immediately. Then,

$$f_f(x,p,u) = 0, f \in F \tag{2.8}$$

The concentration of fast species can be calculated from the concentration of slow species with the above equation and the original ODE model is reduced into a ODE model of slow species and algebraic equations for the fast species.

Quasi Equilibrium (QE) approaximation is to find reversible reactions whose forward and reverse reaction rates are approaximately balanced for given time scale[37]. From the found QEs, algebraic equations are derived and used to reduce the model as in QSSA.

System of Linear Differential Equation and Right Eigenvector

The solution of linear differential equation $\dot{x} = Ax$ is $x(t) = e^{At}x(0)$. From the definition of matrix exponential,

$$e^A = \sum_{n=0}^{\infty} \frac{A^n}{(n+1)!}$$

For a right eigenvector v of a matrix A, $Av = \lambda v$ and

$$\begin{aligned} e^A v &= \sum_{i=0}^{\infty} \frac{1}{(i+1)!} A^i v \\ &= \sum_{i=0}^{\infty} \frac{\lambda^i}{(i+1)!} v \end{aligned}$$

where λ is a eigenvalue of A. If A is diagonalizable and has n linearly independent eigenvectors, the eigenvectors can be used as basis of n dimensional Euclidean space. Then, $x(0)$ can be decomposed into a linear combination of the eigenvectors. The following equality is satisfied in this case[39]:

$$\begin{aligned} x(t) &= e^{At}x(0) \\ &= e^{At}Vc \\ &= \sum_{j=1}^{n}\sum_{i=0}^{\infty} \frac{(t\lambda_j)^i}{(i+1)!} c_j v_j \\ &= \sum_{j=1}^{n} c_j e^{\lambda_j t} v_j \\ &= [v_1 v_2 ... v_n][c_1', c_2' ... c_n']^T \end{aligned}$$

where v_j $(1 \leq j \leq n)$ is a eigenvector of A, $x(0) = c[v_1 v_2 ... v_n]$, and $c_j' = c_j e^{\lambda_j t}$. As can be seen in the equation, projection of $x(0)$ on v_j decayed or amplified with rate of λ_j. That is, The component of the initial state in the direction of v_j is associated with time scale $\frac{1}{\lambda_j}$. A

direction in state space is in column space of a stoichiometric matrix for an ODE model of a CRN so the time scale associated with a right eigenvector can be used to see time scales of reactions for a given point in the state space.

Time Scale Modes

If A is diagonalizable, there is a T such that

$$\begin{aligned}\dot{\hat{x}} &= T^{-1}\dot{x} \\ &= T^{-1}Ax \\ &= T^{-1}ATT^{-1}x \\ &= T^{-1}AT\hat{x} \\ &= \Lambda\hat{x}\end{aligned}$$

where $\Lambda = T^{-1}AT$ is a diagonal matrix and $\hat{x} = T^{-1}x$. T and T^{-1} can be calculated with

$$T = \begin{bmatrix} v_1 & ...v_n \end{bmatrix}$$

$$T^{-1} = \begin{bmatrix} w_1^T \\ ... \\ w_n^T \end{bmatrix}$$

where v anf w is a right and left eigenvector of A and $v_i w_j = \delta_{ij}$ $(1 \leq i, j \leq n)$. Λ is a diagonal matrix whose main diagonal is λ_1,λ_2,...,λ_n. Then, $\dot{\hat{x}}_i = \lambda_i \hat{x}_i$ for $1 \leq i \leq n$ and

$$\hat{x}(t) = \begin{bmatrix} e^{\lambda_1 t}\hat{x}_1(0) \\ e^{\lambda_2 t}\hat{x}_2(0) \\ ... \\ e^{\lambda_n t}\hat{x}_n(0) \end{bmatrix}$$

Each $\hat{x}_i$ is called a time scale mode and is exponentially amplified or decayed with rate of $e^{\lambda_i t}$. Sine $\hat{x}$ is just a vector of linear combinations of x defined as $\hat{x} = T^{-1}x$, it means that left eigenvector w_i represents a lump of state variables associated with time scale λ_i. Also, the following equation holds

$$\begin{aligned}x(t) &= e^{tA}x(0) \\ &= e^{tT\Lambda T^{-1}}x(0) \\ &= Te^{t\Lambda}T^{-1}x(0)\end{aligned}$$

Distance from Slow Manifold

Linear dynamic system $\dot{x} = Ax$ is discussed so far. Local linear approximation of a nonlinear system $\dot{x} = F(x)$ can be done as in the following:

$$\frac{d(x + \Delta x)}{dt} = F(x) + J_F(x)\Delta x + o(|\Delta x|)$$

Since $\frac{dx}{dt} = F(x)$,

$$\Delta \dot{x} = J_F(x)\Delta x$$

The discussion made in previous sections can be applied here by simply settting A as $J_F(x)$. That is,

$$\Delta x(t) = Te^{t\Lambda}T^{-1}\Delta x(0)$$

where $J_F(x) = T\Lambda T^{-1}$.

Time derivative of $\hat{x}$ is

$$\begin{aligned} \frac{d\hat{x}}{dt} &= T\frac{dx}{dt} \\ &= TF(x) \end{aligned}$$

Also, the jacobian is

$$\begin{aligned} \frac{\partial(TF(x))}{\partial \hat{x}} &= \frac{\partial(TF(x))}{\partial x} \cdot \frac{\partial x}{\partial \hat{x}} \\ &= TJ_F(x)T^{-1} \end{aligned}$$

So local linear approximation of $\dot{\hat{x}}$ is

$$\begin{aligned} \frac{d(\hat{x} + \Delta\hat{x})}{dt} &= TF(x) + TJ_F(x)T^{-1}\Delta\hat{x} \\ \frac{d\hat{x}}{dt} + \frac{d\Delta\hat{x}}{dt} &= TF(x) + TJ_F(x)\Delta x \\ \frac{d\Delta\hat{x}}{dt} &= TJ_F(x)\Delta x \\ &= T\frac{d\Delta x}{dt} \end{aligned}$$

On the other hand,

$$\begin{aligned} \frac{d(\hat{x} + \Delta\hat{x})}{dt} &= T\frac{dx}{dt} + T\frac{d\Delta x}{dt} \\ &= TF(x) + T\frac{d\Delta x}{dt} \end{aligned}$$

When mode i is collapsed onto the $n-1$ dimensional slow manifold in n dimensional phase space, following equation holds for the point x^s on the slow manifold.

$$\begin{aligned} \frac{d\hat{x}_i^s}{dt} &= T_{(i,:)}\frac{dx^s}{dt} \\ &= T_{(i,:)}F(x^s) \\ &= 0 \end{aligned}$$

Let $\Delta\hat{x}_i = \hat{x}_i - \hat{x}_i^s$. Then,

$$\begin{aligned} \frac{d\hat{x}_i}{dt} &= T_{(i,:)}\frac{d(\hat{x}_i^s + \Delta\hat{x}_i)}{dt} \\ &= T_{(i,:)}F(x_s) + T\frac{d\Delta x}{dt} \\ &= \frac{d\Delta\hat{x}_i}{dt} \end{aligned}$$

From the above equation,

$$\frac{d\Delta\hat{x}_i}{dt} = T_{(i,:)}F(x)$$

Also,

$$\frac{d\Delta\hat{x}_i}{dt} = \lambda_i\Delta\hat{x}_i$$

Hence,

$$\Delta\hat{x}_i = \frac{T_{(i,:)}F(x)}{\lambda_i|T_{(i,:)}x|} \tag{2.9}$$

$\Delta\hat{x}_i$ can be used as a distance of a mode from the slow manifold [39]. $|T_{(i,:)}x|$ in the denominator is for normalization. The distance measure can be used to know how active a mode is. The closer a mode to the slow manifold is, the less active the mode is. Time scale modes of a system can be classified into active and inactive modes using this distance measure and the number of active modes can be used to calculated a dimension of a slow manifold.

A can be only diagonalized iff there are n independent eigenvectors of A. The concept of decomposing a solution of a linear differential equation with multiple modes associated with different decay rates can be generalized for any real square matrix A using invariant subspace decomposition corresponding Jordan blocks of A. In this case, $T_{(i,:)}$ in (2.9) can be chosen to be a Schur vector [26].

2.2.2 Detection of Quasi Steady-State and Quasi Equilibrium

QSSA and QEA are frequently used model reduction methods based on time scale analysis but finding QSS and QE usually needs experience and intuition about the modeled system[37]. Time scales associated with reactions and lumps of species can be found using the left and right eigenvectors as discussed before. Intrinsic Low-Dimensional Manifold (ILDM) [26] is an approach to reduce a complex model using time scale seperation of lumps of species. ILDM finds data points satisfying constraints $T_{(i,:)}F(x) \approx 0$ for given number of dimension n_s. In case of using continuation method, fixed points are used as the initial guess of the manifold. Then, n_s states are picked and quadratic polynomial of $n - n_s$ states fitting the found data points are found. Since QSS and QE are assumptions on time scale seperation of lumps of species and reactions, it is possible to apply the time scale decomposition to detect QSS and QE. In [37], an approach is introduced to find QSS and QE using Intrinsic Low-Dimensional Manifold (ILDM) and modified version of ILDM.

This chapter has introduced key concepts and background to understand the proposed algorithmic approaches.

Chapter 3

Network Reconstruction

3.1 Introduction

Algorithms in computational biology and bioinformatics are helping rapidly yield new insights into biological and biochemical processes. While much of today's excitement is focused on analyzing data from high-throughput screens (e.g., microarrays, RNAi assays), significant research is also being conducted in constructing and simulating mathematical models of key biological processes, such as the cell cycle [8, 31], circadian rhythms, and entire signaling pathways [5]. These models capture not only qualitative properties of the underlying process but also quantitative traits as revealed by mutant experiments [36]. As shown in Fig. 3.1, such mathematical modeling typically begins with a chemical reaction network (CRN), which is then converted to a set of simultaneous ordinary differential equations (ODEs), which are then numerically simulated to yield time series profiles of the participating molecular species. These profiles are then matched with real data and the model is adjusted to account for discrepancies. More sophisticated methods involving bifurcation plots and phase portraits shed further insight into the qualitative dynamics of the underlying system.

In this thesis, we study the inverse problem, i.e., analyzing time series profiles of the molecular species to reconstruct the CRN (see Fig. 3.1, dotted lines). This finds uses in not just systems biology, as studied here, but also in any domain where chemical reaction systems form the origins of the underlying numerical model (ODE), such as petrochemical plant engineering, environmental engineering, food processing, and manufacturing.

Reconstructing CRNs is relevant not just for system identification but also for model reduction. For instance, it is well acknowledged that models of key biological processes are notoriously complex and difficult to comprehend for humans [5]. A key task therefore is to reduce the reaction system to a smaller system, involving fewer reactions and/or molecules, but yet retain the essential dynamical properties of the system. Given a complex mathematical model of, say, a biochemical process, we can simulate the model to generate data and

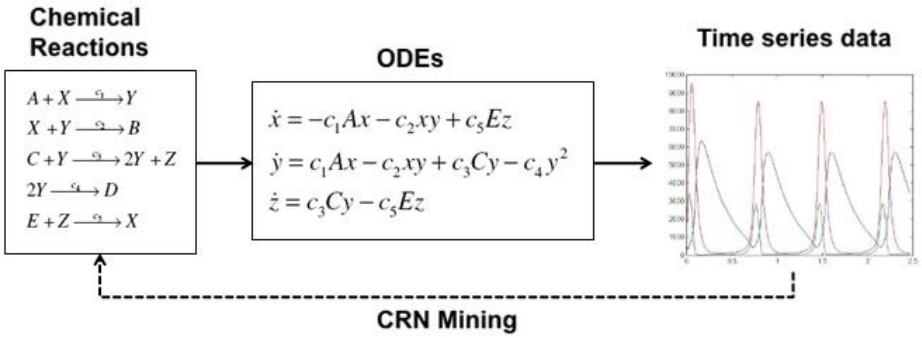


Figure 3.1: CRN mining is the inverse problem of reverse-engineering a set of chemical reactions that can reproduce the dynamics observed in a given time series dataset.

reconstruct a (potentially) smaller model by mining the generated dataset. Such a model → data → model transformation is currently a hot topic in computational systems biology [28].

Pertinent data for mining CRNs can hence be gathered from either experimental observations or computational simulation. The former is the subject of works such as [34] and requires 'wet-lab' machinery as described in [4]. In this thesis, we focus on data from computational simulations of mathematical models for three reasons: the ease of generating data on demand from the given CRN in a controlled fashion, the capability to systematically perturb the CRN and observe the modified dynamics, and the desire to verify our algorithms on some 'ground truth.' Table 3.1 summarizes the input-output description of the network reconstruction problem studied here as well as the methods available to observe, interrupt, or otherwise modify the behavior of the system. This setting of the CRN mining problem is pertinent in computational modeling and systems biology contexts.

Our primary contributions in this thesis are four fold. First, we introduce CRN mining as a new KDD problem and cast CRN mining as the task of mining an undirected graphical model followed by annotating edges and groups of edges with chemical reaction type information. In essence, we capture the dynamics of the network by modeling each species as a random variable and by looking for independence relations between them.

A key issue in mining graphical models among a given set of random variables is to decide whether to detect dependencies or (conditional) independencies. If we choose to detect dependencies, we must take care to distinguish between direct and indirect dependencies. To avoid this issue, classical algorithms (e.g., see [6]) are hence almost exclusively based on detecting independencies, either by explicitly identifying such constraints and summarizing them into a network, or by defining the score of a network based on such relationships and searching in the space of networks. Our second contribution is to show how the novel setting of CRN mining permits us to mine dependencies and yet avoid detecting indirect dependencies, a feature not achievable in traditional (discrete) graphical model mining contexts. Further, our algorithm for CRN mining involves a $O(n^2)$ computation (where n is the number of species) in contrast to algorithms that have exponential running time complexity in the worst case for mining graphical models.

Our third contribution is the notion of 'sensitivity tables' as pattern matching constraints to identify reaction types, such as whether it is a reversible or irreversible reaction, enzyme catalyzed or not, and the precise ratios between the molecules of reactants and products. We hasten to add that we cannot unambiguously distinguish between all possible chemical reaction types and we precisely state the distinctions that we are (un)able to make.

Finally, we demonstrate the application of CRN mining to reconstructing many important biochemical networks in systems biology applications, including prokaryotic gene expression regulation and the CDC-Cyclin2 interaction forming the core of the budding yeast cell cycle.

3.2 Related Research

Most pertinent related research can be found in the systems biology, mathematical modeling, and bioinformatics literature. The 1997 paper by Arkin, Shen, and Ross in *Science* [4] is credited with creating interest in CRN mining; it also presented an all-pairs correlation method for reconstructing the underlying network, with applications to the glycolysis metabolic process. However, the method described in [4] cannot distinguish between direct and indirect dependencies and can thus result in spurious edges. In addition, it assumes that all species are eventually connected and hence cannot recognize disconnected components, such as the simultaneous set of chemical reactions: $\{A \longleftrightarrow B, C \longleftrightarrow D\}$.

There have been many papers that were motivated by the Arkin, Shen, and Ross work described above. For instance, Wiggins and Nemenman [43] present a method to analyze time

Table 3.1: Setting of the CRN mining problem.

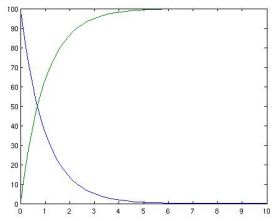
Given
Number of species
Identities of species
To find
Reaction network
Properties of individual reactions
Perturbation capabilities
Can buffer given species (either singly or in subsets)
Can knock-out given species (either singly or in subsets)

series to infer process pathway, which can be construed as representing calling invocations of one pathway by another. However, their method is aimed at producing a general network of relationships from genomic data and not at reconstructing chemical reaction networks. A more theoretical approach is taken in [27] but its strong guarantees of the soundness of network reconstruction are obtained by restricting the focus to discrete dynamical systems, which capture the functional behavior of regulatory networks but not CRNs. More recently, Karnaukhov et al. [21] focus on the reaction identification problem by assuming a general parameterized form for the kinetics of the reaction and fitting rate constants by least squares fitting. This work builds on earlier work by the same authors [20]. CRN mining as studied here subsumes reaction identification as a sub-goal.

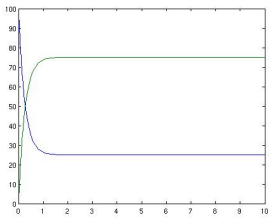
Thus, our formulation of CRN mining is novel for its attempt to model both the dependence structure of chemical species and the properties of individual reactions.

3.3 Some Chemistry for Data Miners

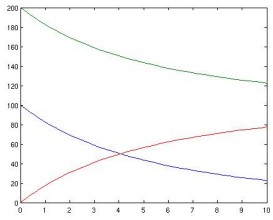
Before we present our algorithm for reconstructing chemical reaction networks, we review some basic chemistry and established practices in the mathematical modeling of chemical reactions. This is the subject of many excellent books, such as [23] which especially focus on modeling for bioinformatics applications. For the data mining audience, we present an abridged version of this literature involving only topics necessary to understand the ensuing algorithm.



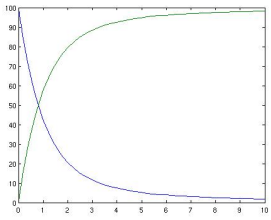
(a) $A \xrightarrow{k_1} B$



(b) $A \underset{k_2}{\overset{k_1}{\longleftrightarrow}} B$



(c) $A + B \xrightarrow{k_1} C$



(d) $S \xrightarrow{E} P$

Figure 3.2: Dynamics of reactions 3.1, 3.3, 3.5, and 3.12, respectively. Parameters used in the above plots: (a) $k_1 = 1$, $x_A(0) = 100$ and $x_B(0) = 0$. (b) $k_1 = 3$, $k_2 = 1$, $x_A(0) = 100$ and $x_B(0) = 0$. (c) $k_1 = 0.001$, $x_A(0) = 100$, $x_B(0) = 200$ and $x_C(0) = 0$. (d) $k_1 = 1$, $k_{-1} = 10$, $k_2 = 1$, $x_S(0) = 100$ and $x_P(0) = 0$.

3.3.1 Modeling a Single Reaction

The simplest example of a chemical reaction is the irreversible isomerization reaction

$$A \xrightarrow{k_1} B. \tag{3.1}$$

where k_1 denotes the rate at which species A is converted into B. If the concentrations of the species A and B are represented by x_A and x_B, the dynamics of (3.1) can be formulated by a set of ordinary differential equations (ODEs)

$$\begin{cases} \frac{dx_A}{dt} &= -k_1 x_A, \\ \frac{dx_B}{dt} &= k_1 x_A. \end{cases} \tag{3.2}$$

A typical trajectory of x_A and x_B in this simple system is shown in Figure 3.2 (a).

The reaction (3.1) is a special case of the reversible isomerization reactions

$$A \underset{k_2}{\overset{k_1}{\longleftrightarrow}} B. \tag{3.3}$$

The corresponding ODEs are:

$$\begin{cases} \frac{dx_A}{dt} &= -k_1 x_A + k_2 x_B, \\ \frac{dx_B}{dt} &= k_1 x_A - k_2 x_B. \end{cases} \tag{3.4}$$

A typical trajectory for this system is shown in Figure 3.2 (b).

Both reactions (3.1) and (3.3) are linear. The simplest nonlinear example is the bimolecular reaction

$$A + B \xrightarrow{k_1} C. \tag{3.5}$$

The corresponding ODEs are given below.

$$\begin{cases} \frac{dx_A}{dt} &= -k_1 x_A x_B, \\ \frac{dx_B}{dt} &= -k_1 x_A x_B, \\ \frac{dx_C}{dt} &= k_1 x_A x_B. \end{cases} \tag{3.6}$$

A typical trajectory of equation (3.6) is shown in Figure 3.2 (c).

The kinetics in reactions (3.1), (3.3) and (3.5) are simple mass action kinetic laws. But equations can be more complicated. Consider the enzyme-substrate reactions

$$E + S \underset{k_{-1}}{\overset{k_1}{\leftrightarrows}} ES \xrightarrow{k_2} E + P. \tag{3.7}$$

Here E represents enzyme species, whose total concentration $E_0 = x_E + x_{ES}$ remains as a constant in this chemical process. The corresponding ODEs are

$$\begin{cases} \frac{dx_S}{dt} &= -k_1 x_E x_S + k_{-1} x_{ES}, \\ \frac{dx_E}{dt} &= -k_1 x_E x_S + (k_{-1} + k_2) x_{ES}, \\ \frac{dx_{ES}}{dt} &= k_1 x_E x_S - (k_{-1} + k_2) x_{ES}, \\ \frac{dx_P}{dt} &= k_2 x_{ES}. \end{cases} \tag{3.8}$$

When k_1 and k_{-1} are much larger than k_2, we can assume the first two reactions in (3.7) reach partial equilibrium. This partial equilibrium assumption can be formulated by

$$k_1 x_E x_S = k_{-1} x_{ES}. \tag{3.9}$$

When k_2 is in a similar magnitude of k_{-1}, the equilibrium assumption (3.9) does not hold any more. But a steady state assumption can be made. It assumes that the concentration of ES remains a steady state after a transient period, which is formulated as

$$k_1 x_E x_S = (k_{-1} + k_2) x_{ES}. \tag{3.10}$$

It turns out that (3.9) is a special case of (3.10). Let $k_M = \frac{k_2 + k_{-1}}{k_1}$. With the assumption that E_0 is much smaller than x_S, we can derive

$$\frac{dx_P}{dt} = \frac{k_2 E_0}{k_M + x_S} x_S. \tag{3.11}$$

Let $k = \frac{k_2 E_0}{k_M + x_S}$. The equation (3.11) is called the Michaelis-Menten equation. It reduces the enzyme-substrate reaction (3.7) into a simple reaction

$$S \xrightarrow{E} P. \tag{3.12}$$

denoting that substrate S is catalyzed by enzyme E to form product P. But (3.12) is fundamentally different from the simple reaction (3.1) because it follows the nonlinear enzyme kinetics (3.11). A typical trajectory of the reaction (3.12) is shown in Figure 3.2 (d).

3.3.2 Modeling Sets of Reactions

A chemical reaction network (CRN) is composed of many reactions. Suppose N species are involved in M reaction channels in a CRN. Let the concentration of these species be denoted by x_i, $i = 1, \cdots, N$ and the reaction channels be denoted by R_j, $j = 1, \cdots, M$. The dynamics of the system can be formulated as

$$\frac{dx}{dt} = f(x), \tag{3.13}$$

where $f_i(x) = \sum_{j=1}^{M} \nu_{ij} r_j(x)$. Here ν is called the stoichiometric matrix. ν_{ij} is the unit change of x_i caused by the reaction channel R_j and $r_j(x)$ is the reaction rate function for the reaction channel R_j. For example, in the simple reaction (3.1), there are two species and one reaction channel. $\nu = [-1,\ 1]$ and $r_1(x) = k_1 x_A$. In the bimolecular reaction (3.5), $\nu = [-1,\ -1,\ 1]$ and $r_1(x) = k_1 x_A x_B$. In the reduced enzyme-substrate reaction (3.12), $\nu = [-1,\ 1]$ and $r_1(x) = \frac{k_2 E_0}{k_M + x_S}$.

But often the state space in (3.13) can be reduced by applying conservation laws and partial equilibrium or steady state assumptions. Examples of the partial equilibrium assumption and steady state assumption are given in (3.9) and (3.10) for the enzyme-substrate reaction (3.7). Conservation laws can be applied for all examples shown above. For example, for reaction systems (3.1) and (3.3), the sum of x_A and x_B remains as a constant. That can be formulated as

$$x_A + x_B = C_0. \tag{3.14}$$

With this conservation law, we only need to formulate the dynamics of one variable. The other can be directly calculated from (3.14). Thus the dimension of the state space in both equations (3.2) and (3.4) can be reduced by 1. In the bimolecular reaction (3.5), there are two conservation laws

$$\begin{cases} x_A + x_C &= C_0, \\ x_B + x_C &= C_1. \end{cases} \tag{3.15}$$

With the two constraints, the dimension of the state space in equation (3.6) can be reduced to 1.

For a complex CRN, the ODEs and the algebraic constraints can be put together. Then we obtain a set of differential-algebraic equations (DAEs)

$$x' = f(x, y), \tag{3.16}$$
$$0 = g(x, y), \tag{3.17}$$

where (3.16) is the differential part and (3.17) is the algebraic part.

3.3.3 Sensitivity Analysis

Sensitivity analysis is widely used in optimization, parameter estimation, uncertainty and stability analysis. (Here we demonstrate its applications to data mining and network reconstruction.) For a CRN represented by a set of DAEs, the system often contains uncertainty due to unknown kinetic rates, environment fluctuations, and other unknown possible reaction pathways. They can be represented as parameters in DAEs. We can rewrite the equation (3.16-3.17) as

$$x' = f(x, y, p), \tag{3.18}$$
$$0 = g(x, y, p), \tag{3.19}$$

with initial conditions $x_0 = x_0(p)$ and $y_0 = y_0(p)$. Sensitivity reflects the change rates of the state variables x and y with respect to the change in the parameter p, which are calculated by $\frac{dx}{dp}$ and $\frac{dy}{dp}$.

The sensitivity functions $\frac{dx}{dp}(t)$ and $\frac{dy}{dp}(t)$ can be obtained from the numerical time series data or estimated by finite difference methods during the process of solving the original DAEs and derived sensitivity equations. Software such as DASPK (in Fortran) [7] and CVODES (which comprises the CVODE [9], KINSOL, and IDE software components in C) have inbuilt capabilities to perform sensitivity analysis of DAEs.

However, one advantage of our algorithm is its robustness. We do not require an accurate measurement of the sensitivity. Instead, just the signs of the sensitivity are needed. Since the sensitivity is a function of time, the sign is taken at the time point where the function has the maximum absolute value. Moreover, we have set the threshhold as 10^{-8}. If the absolute value of a sensitivity is below this threshhold, it is labeled as zero. An end user can adjust the threshhold value based on how reliable the measurement is. In this way, we can avoid some false alarms from noise data. The cost is that sometimes we may not be able to detect some reaction if its sensitivity is lower than the threshhold.

3.4 Using Systematic Probing to Identify CRNs

Referring back to the experimental context in Table 3.1, we present an approach to reconstructing chemical reaction networks by systematically perturbing the network to identify

relationships between the given species. (Although such perturbations are well studied in biochemistry, leading to the notion of minimal cut sets in biochemical networks [22], they have primarily been used for engineering flux patterns, not for CRN mining.) As Table 3.1 shows, there are two main classes of perturbations available: buffering and knock-out experiments.

3.4.1 Buffering experiments

Buffering involves providing enough supply (intake) of some species, thus forcing it to stay constant. In the corresponding DAEs, this is equivalent to replace the corresponding differential equation by a simple algebraic equation. Note that buffering will break the corresponding conservation constraints.

For example, consider a simple chain reaction system

$$A \xrightarrow{k_1} B \xrightarrow{k_2} C. \tag{3.20}$$

The corresponding equations are

$$\begin{cases} \frac{dx_A}{dt} &= -k_1 x_A, \\ \frac{dx_C}{dt} &= k_2 x_B, \\ x_A + x_B + x_C &= C_0. \end{cases} \tag{3.21}$$

If we perturb the initial value of A (let $x_A(0) = p$), we can calculate the corresponding change resulted in C (by $\frac{dx_C}{dp}$). We then know A and C are connected in the system. If B is buffered, x_B stays as a constant. Then the equations become

$$\begin{cases} \frac{dx_A}{dt} &= -k_1 x_A, \\ \frac{dx_C}{dt} &= k_2 x_B, \\ x_B &= B_0. \end{cases} \tag{3.22}$$

We conduct the sensitivity analysis again and we will get $\frac{dx_C}{dp} = 0$! This shows that after B is buffered, A and C become disconnected. We can then conclude about the structure of this network: A affects C through B.

3.4.2 Knock-out experiments

A second type of perturbation that is common in biology is the knock-out, i.e., to remove a molecule completely by rendering it inactive or unable to participate in the reaction. Engineered biological systems by knocking out key molecules are referred to as *mutants*. In the corresponding DAE, knock-outs correspond to a special form of buffering, namely replacing the respective species variables to zero.

However, knock-outs, while useful at understanding loss-of-function, are not very revealing for reconstructing CRNs. For instance, compare the chain reaction:

$$A \longrightarrow B \longrightarrow C$$

with the enzyme catalyzed reaction:

$$A \xrightarrow{B} C$$

By buffering B, we can distinguish between the two cases by detecting whether $\frac{dx_C}{dp} = 0$ (first case) or whether $\frac{dx_C}{dp} > 0$ (second case). Here p is the initial value of A as before. However, if we knock out B from the respective equations, both of them result in $\frac{dx_C}{dp} = 0$! For this reason, in this thesis, we exclusively focus on buffering as a means to probe CRNs.

3.4.3 CRNs and Graphical Models

The above observations hint at the relationship between CRNs and undirected graphical models [24]. We first setup the correspondence between a given CRN and a corresponding graphical model. Two terms are defined here to denote two types of reactions depending on the number of species involved in each reaction. *Bi-reaction/Tri-reaction* is a reaction connecting two/three species. For ease of presentation, in the following lemmas and results, we assume only bi-reaction reactions although our algorithmic implementation and experimental results involve both bi-reactions and tri-reactions.

Definition 1. *Given a CRN $\mathcal{N}$ (a set of molecular species and a set of chemical reactions between them) we define the undirected graph $\mathcal{G}(\mathcal{N})$ corresponding to $\mathcal{N}$ as the graph whose nodes corresponds to the species in $\mathcal{N}$ and whose edges connect nodes that participate in a common reaction.*

Note that different CRNs might induce the same undirected graphical model. For instance, the reaction sets $A \longleftrightarrow B \longleftrightarrow C$ and $A \longrightarrow B \longrightarrow C$ induce the same graph even though the former involves reversible reactions and the latter involves irreversible reactions. Nevertheless, the following results demonstrate that mining graphical models is an useful first step to reconstructing CRNs.

Lemma 3.4.1. *Given a network $\mathcal{N}$ and its undirected graph $\mathcal{G}(\mathcal{N})$, node n_1 is conditionally independent of node n_2 given a set of nodes n_X in $\mathcal{G}(\mathcal{N})$ iff the following applies: after buffering n_X in $\mathcal{N}$, the sensitivity of n_1 to n_2 (and vice versa) is zero.*

A direct application of Lemma 3.4.1 would require us to search through an exponential set of possible conditioning contexts. Instead, as stated earlier, we will seek to identify dependencies.

Lemma 3.4.2. *Given a network $\mathcal{N}$ and its undirected graph $\mathcal{G}(\mathcal{N})$, an edge exists between node n_1 and node n_2 in $\mathcal{G}(\mathcal{N})$ iff the following applies: the sensitivity of n_1 to n_2 (or vice versa) after buffering all other molecules in $\mathcal{N}$ is non-zero.*

Unlike Lemma 3.4.1, Lemma 3.4.2 requires only a search through $O(n^2)$ conditioning contexts. Then why don't traditional Markov network learning algorithms utilize a similar approach? This is because to verify each of the $O(n^2)$ conditional dependencies, the conditioning set involve $n-2$ variables and, even if each variable takes on only two values, we will have to investigate 2^{n-2} settings for conditioning contexts. Besides the exponential complexity, projecting to $n-2$ variables typically will retain very few tuples, typically not sufficient to estimate dependence. Other works such as [6] acknowledge these issues and, in fact, incorporate the size of the conditioning context in their analysis of algorithm complexity. However, in CRN mining, these limitations do not apply since there is a proportional, rather than exponential, cost to a buffering experiment w.r.t. the size of the conditioning context (i.e., the number of buffered molecules). Furthermore, the limitations of sample data sizes do not obviously arise in a buffering experiment.

3.5 Algorithms for Chemical Reaction Network Reconstruction

Our approach to CRN reconstruction begins by first reconstructing the underlying graphical model (Algorithm 1: InferGraphicalModel) followed by cataloging the individual edges or groups of edges into reactions (Algorithm 2: FindReactions). These are detailed next.

Algorithm 1 InferGraphicalModel

Input: V, ODE_v
Output: S
  for all $i, j \in V (i < j)$ **do**
    $(S(i,j), S(j,i)) \leftarrow \text{BufferedSim}(i, j, V - \{i, j\}, ODE_V)$
  end for

3.5.1 Reconstructing Network Topology

InferGraphicalModel takes as input V, the set of all chemical species whose dynamics are given by the system of ODEs in ODE_V. As stated earlier, it conducts a $O(n^2)$ buffered simulation to identify sensitivities between all pairs of molecules (in both directions). Here, $S(i,j)$ denotes the sensitivity of j to the initial concentration of i. InferGraphicalModel produces as output the sensitivity matrix S whose non-zero entries encode the graphical model. As can be seen in Fig. 3.3 from the next section, it is clear that all tri-reactions have

Algorithm 2 FindReactions

Input: V, S
Output: Bi, Tri
  for all $i, j \in V (i < j)$ **do**
    if $|S(i, j)| \geq stol$ or $|S(j, i)| \geq stol$ **then**
      $E \leftarrow E \cup \{i, j\}$
    end if
  end for
  Initialize all elements of CV to be 0
  $SI \leftarrow sign(S, stol)$
  for all $e_k, e_m \in E\,(k < m)$ **do**
    if e_k and e_m share a vertex b s.t. $e_k = \{a, b\}$ and $e_m = \{b, c\}$ and Tri.find$(\{a, b, c\}) =$ false **then**
      reactions $\leftarrow$ LookupTriReaction$(\{a, b, c\}, SI)$
      if reactions is not empty **then**
        Tri.add$(\{a, b, c\},$ reactions$)$
        set $CV(\{a, b\}), CV(\{b, c\}), CV(\{c, a\})$ to be 1
      end if
    end if
  end for
  for all $e = \{h, i\} \in E$ **do**
    if $CV(\{h, i\}) = 0$ **then**
      reactions $\leftarrow$ LookupBiReaction$(\{h, i\}, SI)$
      if reactions is not empty **then**
        Bi.add$(\{h, i\},$ reactions$)$
      end if
    end if
  end for

corresponding size 3 consecutive subgraphs in in the inferred graphical model, thus justifying the next algorithm.

The next algorithm, FindReactions, takes as input the set of chemical species as before and the just computed sensitivity matrix S. It produces as output the list of detected bi-reactions in Bi and tri-reactions in Tri. First, it thresholds the sensitivity matrix S into SI. The array CV is used to hold a CoVer for the molecular species and their dependencies, i.e., to see if a dependency detected in InferGraphicalModel has been 'explained' by a chemical reaction. Initially no dependencies are explained, hence CV, indexed by the dependencies, is initialized to zero. Algorithm FindReactions then proceeds to look for tri-reactions that fit the sensitivity profiles computed in SI (using Table 3.3, explained in the next section) and if a suitable reaction is found, the array CV is updated suitably. Only after all trimolecular combinations are exhausted does it proceed to look for bi-reactions. At this point, it is important to mention that the algorithm LookupTriReaction (not detailed here) searches through all permutations of the given triple of molecules in establishing a correspondence to sensitivity profiles.

3.5.2 Reconstructing Reaction Properties

It remains to be detailed how LookupTriReaction and LookupBiReaction work. The advantage to these algorithms is that they use sensitivities between pairs of molecules which can actually be computed alongside the reconstruction algorithm. Tables 3.2 and 3.3 contain the relevant information for disambiguating reaction types. The same information is also summarized graphically in Fig. 3.3. Rather than go through each entry sequentially, we explain below how the sensitivity table patterns can be used to make important distinctions.

Sensitivity changes with time. Let $s_{A,B}(t)$ be the time series of sensitivity of B to the initial concentration of A. We first discretize this time series into '+', '-', and 0 values. The sign of the sensitivity profile, $s(A,B)$, is then defined as the sign of $s_{A,B}(t_i)$ where t_i is the time point at which $|s_{A,B}(t_i)|$ is maximum. We index into Tables 3.2 and 3.3 using these signs and identify reaction types. Recall that Table 3.2 is meant to be used for identifying reactions between pairs of molecules *after* Table 3.3 has been used to identify reactions between triples. Also, Table 3.3 is richer in detail than Table 3.2 since it gives the signs of sensitivities of six basic tri-reactions: $A \xrightarrow{B} C$, $A \longleftrightarrow B + C$, $A \longrightarrow B + C$, $A \xrightarrow{A} B + C$, $A \xrightarrow{B} B + C$, and $A + B \longrightarrow C$, and under three different buffering conditions.

We should point out that not all distinctions can be made unambiguously. For instance, in Table 3.2[1], there are five possible reactions but only three distinct sensitivity patterns. Hence some rows lead to multiple hypotheses. A direction of future work is to develop a

[1]A note about the asterisk in this table: due to the process of enzyme-substrate complex formation, the entry $s(B,A)$ is negative for the initial reaction and later changes its sign to a plus as shown in Table 3.2. If we assume that the (initial) concentration of B is much smaller than the concentration of A, then this entry can be treated as a '+'.

constraint engine that can reason about such multiple hypotheses, across adjacent sensitivity profiles, to achieve greater discrimination of detection.

Reversible versus Irreversible

Distinguishing between reversible and irreversible reactions is straightforward, e.g., Table 3.2 can be readily used to distinguish between $A \longrightarrow B$ and $A \longleftrightarrow B$ by assessing the sign of $s(B, A)$.

Multiple reactants

This situation requires us to distinguish between the tri-reaction $A + B \longrightarrow C$ and the combined set of two bi-reactions $\{A \longrightarrow C,\ B \longrightarrow C\}$. $s(A, B)$ and $s(B, A)$ are zero for the two bi-reactions but $s(A, B)$ and $s(B, A)$ are negative in the tri-reaction, thus enabling the distinction.

Multiple products

This situation is the converse of the previous case. Note that $A \longrightarrow B+C$ and the combined set of two bi-reactions $\{A \longrightarrow B,\ A \longrightarrow C\}$ have the same signs of sensitivities according to Tables 3.2 and 3.3. Thus, $A \longrightarrow B + C$ and $\{A \longrightarrow B,\ A \longrightarrow C\}$ cannot be distinguished in our approach.

Stoichiometry

Stoichiometry refers to the relative ratios of molecules that participate in a reaction. Thus, the only distinction between the reactions: $A \longleftrightarrow B$ and $2A \longleftrightarrow B$ is one of stoichiometry. Using only the signs of the sensitivity entries, these reactions cannot be disambiguated. On the other hand, if information about the magnitude of the sensitivity is available, e.g., if we know that $\frac{s_{A,A}(t)}{s_{B,A}(t)} \approx c$ and $\frac{s_{A,B}(t)}{s_{B,B}(t)} \approx c$, then we can conclude the existence of reaction $cA \longleftrightarrow B$ in steady state.

Enzyme catalysis

An enzyme-substrate reaction can be modeled with either mass action kinetics or Michaelis-Menten kinetics. When the enzyme-substrate reaction is modeled with mass action kinetics, the sensitivity profiles are identical for $A \xrightarrow{B} C$ and $A+B \longrightarrow C$ (see row 3 of Table 3.3). On the other hand, if the enzyme-substrate reaction is modeled with Michaelis-Menten kinetics, then these reactions can be disambiguated (see row 4 of Table 3.3).

Table 3.2: The Bimolecular sensitivity table used to identify chemical reactions involving 2 molecules.

Reaction	s(A,B)	s(B,A)
$A \longrightarrow B$ or $A \xrightarrow{A} B$	+	0
$A \longleftrightarrow B$ or $2A \longleftrightarrow B$	+	+
$A \xrightarrow{B} B$	$+^*$	-

Auto-catalysis

Auto-catalysis is the situation where a molecule catalyzes a reaction that it itself participates in. It is easier to detect if the catalyst is the product, rather than the reactant. For instance, as can be seen in Table 3.2, $A \longrightarrow B$ and $A \xrightarrow{A} B$ have the same sensitivity profile, whereas $A \longrightarrow B$ and $A \xrightarrow{B} B$ can be distinguished. Similarly, in Table 3.3, $A \longrightarrow B + C$ and $A \xrightarrow{A} B + C$ have the same sensitivity profile (see row 2) and thus cannot be distinguished.

Detecting Groups of Reactions

The last two rows of Table 3.3 are especially designed to detect common groups of reactions. The '+' sign for $s(C, A)$ in both these rows helps detect the existence of a loop back from molecule C to A which is not the case, for instance, in rows 3 and 4 of Table 3.3. Within the last two rows, further disambiguation about rate laws can be made using the sign of $s(A, B)$.

More Complex Dynamics

By capturing more of the dynamics, these tables can be put to further use in reaction identification. For instance, consider the task of distinguishing $A \xrightarrow{B} C$ from $A + B \longrightarrow C$ (using rows 3 and 6 of Table 3.3). When A is buffered, $s(A, C)$ and $s(B, C)$ grow boundlessly in $A \xrightarrow{B} C$. Whereas, in $A + B \longrightarrow C$, $s(A, C)$ is limited by B. Hence, $s(A, C)$ stops increasing after reaching steady state.

3.6 Limitations and Possible Solutions

Thus far, we have made two critical assumptions that are necessary to the success of our reconstruction algorithm:

1. Between a given pair or triple of molecules, there is at most one reaction.

Table 3.3: The 'All but 2' sensitivity table used to identify chemical reactions involving 3 molecules.

Reaction(s)	A buffered		B buffered		C buffered	
	s(B,C)	s(C,B)	s(A,C)	s(C,A)	s(A,B)	s(B,A)
$A \longleftrightarrow B + C$	-	-	+	+	+	+
$A \longrightarrow B + C$ or $A \xrightarrow{A} B + C$	0	0	+	0	+	0
$A \xrightarrow{B} C$ or $A + B \longrightarrow C$	+	0	+	0	-	-
$A \xrightarrow{B} C$ (Michaelis-Menten)	+	0	+	0	0	-
$A \xrightarrow{B} B + C$	+	0	+	0	+	-
$A \xrightarrow{B} C$ or $A + B \longrightarrow C$ with $C \longrightarrow A$	+	0	+	+	-	-
$A \xrightarrow{B} C$ with $C \longrightarrow A$ (Michaelis-Menten)	+	0	+	+	0	-

2. The rate laws governing the reactions fall into the categories of either the mass-action formulation (equations 3.2) or Michaelis-Menten kinetics (equation 3.11).

These assumptions are not difficult to surmount but their removal is beyond the scope of this thesis. Consider for instance the network in Fig. 3.4 governing how cells in frog egg extracts divide. The core of this network involves a clique of four nodes (molecules) with six overlapping reactions between them! To recognize such a circuit, where dynamics between a given set of molecules are best explained by multiple reactions, we must be able to decompose observed sensitivity profiles into additive combinations of smaller components, each of which corresponds to a basic reaction. The second problem is applicable in situations where reaction rates do not fall into the two basic types studied here. For instance, rate laws can be highly non-linear and involve more than one enzyme to catalyze a given reaction. Further, very fast rate constants can cause drastic changes in concentrations, too quick to be detectable by analyzing data.

Both these problems can be alleviated by numerical modeling of sensitivity profiles rather than the discrete approach of sensitivity tables as studied here. For instance, numerical optimization can be used to find fits to parameterized reaction laws and by repeatedly modeling the residual, we can detect multiple reactions spanning a given set of molecules. The last two rows of our 'All but 2' sensitivity table (Table 3.3) provide a limited capability in this regard and which we have used in the studies described below.

3.7 Experimental Results

Our experimental results are focused on reconstructing key CRNs underlying important biological processes (see Table 3.4). Here we depict the number of species and reactions for each system but hasten to add that the complexity of a CRN cannot be judged merely on

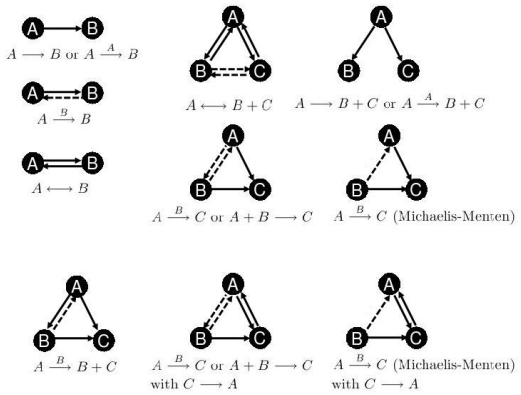


Figure 3.3: A graphical notation (not meant to be a probabilistic graphical model) of the information from Tables 3.2 and 3.3. A solid arrow from node X to node Y exists if sensitivity of Y to initial value of X is positive. A dashed arrow from node X to node Y exists if sensitivity of Y to initial value of X is negative. Larger arrowheads indicate higher levels of sensitivity. No arrow denotes a sensitivity of zero.

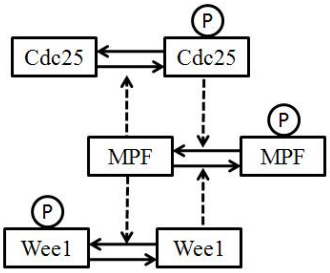


Figure 3.4: CRN governing cell-cycle transitions in frog egg extracts.

these factors alone. For instance, the rather innocuous looking system from Fig. 3.1, referred to as the 'Oregonator', forms the model for many reaction-diffusion systems and can exhibit very complex dynamics including sustained oscillations. It is hence the range of qualitative behaviors that can be exhibited by the system that constitutes its complexity.

For each CRN studied here, we formulated the corresponding ODE as described in Section 3.3, and generated data corresponding to each ODE using the CVODE software [9]. All rate law equations were modeled using either mass action kinetics or Michaelis Menten kinetics. For each pair of molecules, the buffering algorithm buffers all but these two molecules, and the sensitivity profiles between these molecules are computed. A tolerance of 10^{-8} was used to discretize the computed sensitivities. This information drives the reconstruction of topology and reaction characteristics. The results are evaluated using metrics of recall (number of correctly reconstructed reactions as a fraction of true reactions) and precision (number of correctly reconstructed reactions as a function of all reconstructed reactions). In assessing correctness, to allow partial matches, we evaluate reversible reactions in both directions (i.e., if the algorithm reconstructs the reaction in only one direction, we count it as one out of two reactions inferred correctly).

Table 3.4: Summary of CRNs reconstructed and evaluation statistics.

Model	# species	# reactions	Recall	Precision	ODE and sensitivity solution time (10^{-3}s)	CRN mining time (10^{-3}s)
CDC-Cyclin2 interaction loop(Fig. 3.5)	6	6	0.83	0.83	42.3	0.27
Arkin's computational circuit(Fig. 3.6)	7	6	1	1	167	0.51
Prokaryotic gene expression model	9	8	0.875	0.875	97.6	0.56
Frog egg extracts (Fig. 3.4)	8	8	0.75	0.857	58	0.38
Generic yeast cell cycle model(Fig. 3.7)	16	21	0.857	0.88	637	2.31

The CRNs considered here span a variety of model systems in biology. The CDC-Cyclin2 interaction loop (Fig. 3.5 [40]) is the core signaling pathway driving progression through the cell cycle. It is embedded inside the larger yeast cell cycle model described in Fig. 3.7 [11]. A less complex model drives cell cycle transitions in frog egg extracts, as described earlier in Fig. 3.4. Two other models considered here are a CRN underlying gene expression regulation in prokaryotes, which are primitive organisms such as bacteria that do not contain membrane-bound organelles (not shown due to space considerations) and a CRN meant to serve as a generic logic gate (Fig. 3.6).

As Table 3.4 reveals, our algorithm achieves consistently high values of recall and precision across these CRNs. The three reasons it fails to find correct reactions or infers spurious reactions are: the inherent inability to distinguish between certain types of reactions (as discussed earlier), rapid reaction rates that mistakenly cause the algorithm to infer lack of connectivity between some species, and the restriction to at most one reaction between a given pair or triple of molecules. Even with these caveats, it is clear that the algorithm can be used as a primitive to identify key circuits underlying a collection of molecules.

Table 3.4 also tabulates the time taken to reconstruct each CRN. This includes the time taken for the network discovery aspect, inference of reaction properties, plus the time involved in solving the ODE as well as the associated buffering experiments. Therefore the time taken to reconstruct the CRNs is a function of not just the size of the CRN but also the stiffness of the underlying ODE. (A stiff equation requires that the ODE integrator use an extremely small stepsize due to components varying at different time scales or because of underlying numerical instability.) As a result, although there is an underlying $O(n^2)$ complexity to CRN inference, larger models do not necessarily cause proportional increases in time.

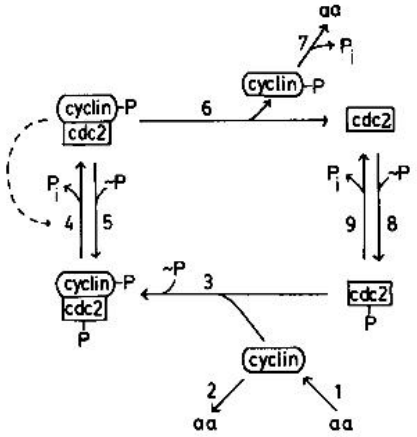


Figure 3.5: The CDC-Cyclin2 interaction loop forming the core of the budding yeast cell cycle. Courtesy John Tyson.

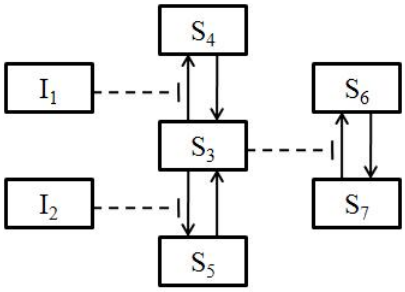


Figure 3.6: A CRN designed to serve as a computational element (i.e., as a logic gate).

3.8 Discussion

We have presented a novel application of data mining methodology to chemical reaction system identification with a marriage of numerical methods and graphical models. Our work is the first to address CRN mining using KDD concepts and methodology.

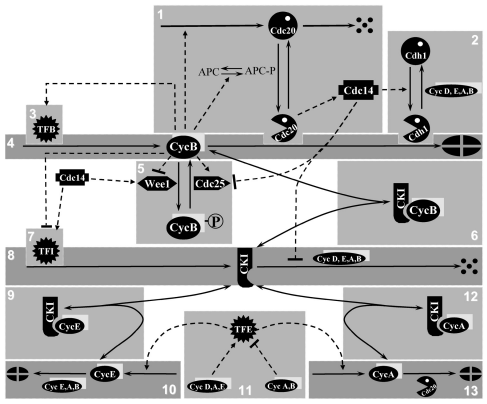


Figure 3.7: Generic CRN of the budding yeast cell cycle. Regulatory modules are given by the shaded rectangles. The different symbols denote different classes of proteins, e.g., the 'PacMan' denotes active forms of regulated proteins. Courtesy Attila Csikasz-Nagy.

Chapter 4

Finding Bistable Cores

4.1 Introduction

The biochemical switch is an important motif frequently found in biological processes. There are many types of studies to find conditions of bistability by mathematically deducing structural properties of bistable systems. For instance, conditions for bistability for a special class of systems called input-ouput monotone systems are introduced in [3]. The found conditions are only applicable to a subset of bistable systems, and hence cannot be used to find the core structures of an arbitrary bistable CRN. A broader goal of our research is to understand how interactions between molecular species in a CRN co-occurr with switching behavior of a bistable system.

Table 4.1 depicts the inputs and outputs of the core finding problem. As can be seen in the table, the inputs are the target CRN, time scale of interest, order of interactions to consider, and trajectories during switching transitions. Time scale separation is very common in a

Table 4.1: Setting of the Bistable core finding problem.

Given
A bistable CRN
Time scale of interest
Order of interactions
A trajectory
To find
Subnetwork of the given CRN which is responsible for the transition dynamics with the given time scale

biochemical system and 'fast' dynamics are often considered to be not necessary to reproduce. In addition, all possible combinations of interactions between species in the input CRN can become very large even for a moderate size CRN so it is necessary to consider only interactions up to a certain order. From these inputs, the objective is to find parts of the input CRN including interactions active during switching transitions.

Switching transitions of a bistable system co-occurs with losing local stability of a stable steady state where it has been. There has been prior research relating topology of a CRN with stability of steady state. Instability causing structure analysis (ICSA), described earlier in this dissertation, is one of such approaches; feedback loops making some of the eigenvalues of the system Jacobian to be positive are identified by conducting symbolic analysis on the coefficients of the characteristic polynomial. A positive feedback is a well known necessary condition of bistability and such positive feedback loops correspond to instability causing structures[44]. Finding ICSs of a bistable system reveals core structure of a bistable switch.

ICSA is majorly used to analyze a specific system of interest to identify candidate feedback loops which can make a steady state unstable. ICSA alone is not sufficient to identify *which* of the found ICSs can induce actual switching transitions.

The goal of this research hence is to develop an automated way to discover ICSs driving the switching behavior of for a given bistable system. We exploit the availability of a large number of bistable CRNs in the CSPACE database [32] so that we can employ large-scale simulation over this database to identify switching-inducing ICSs.

4.2 Approach

4.2.1 Database of Chemical Stability Space

The CSPACE database is available at http://docss.ncbs.res.in. In [32], 12 basic reaction types are defined and synthetic CRNs are generated by combining reactions of those types. Parameter values making the generated networks bistable are searched and recorded in the database with the network configurations.

A record of a bistable configuration in DOCSS consists of model no, reaction signature, and values of parameters. The following is an example of such a record:

M116: |AabX|DbaX|Jacb| 0.064 2.36 0.266 0.018 0.019 2.360 0.709 1.882 0.030

Here, M116 is the model number identifying the record. |AabX|DbaX|Jacb| is the signature of reactions where each token separated by '|' symbol denotes a reaction. The first symbol in the token represent the type of the reaction and next two symbols are names of chemical

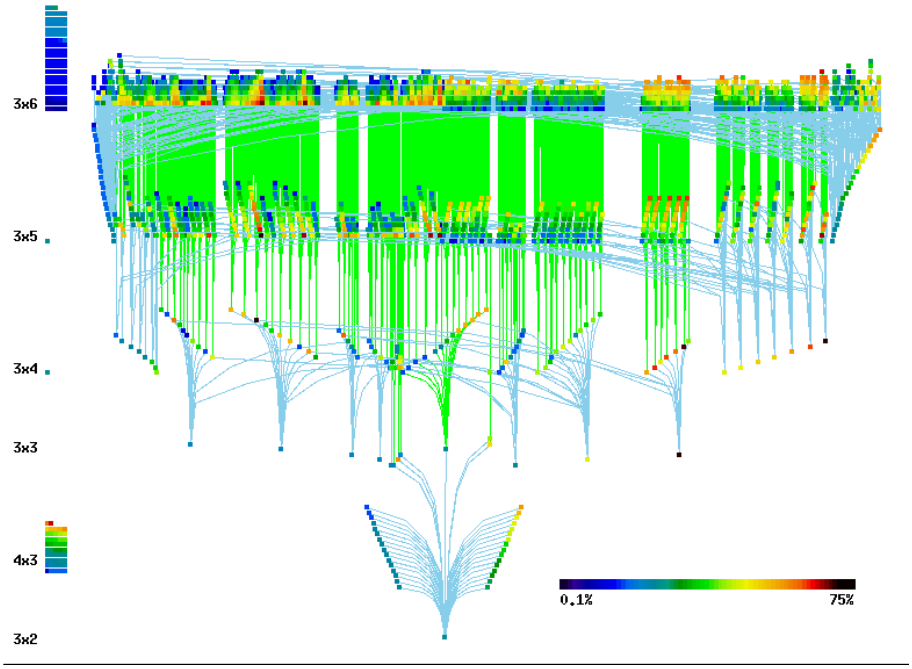


Figure 4.1: Directed acyclic graph of bistable configurations[32]

species involved in the reaction. The basic reaction types are defined reactions of three distinct chemical species so the last symbol is either the name of the species or placeholder, depending on the reaction type. For example, |AabX|DbaX|Jacb| denotes a CRN of three reactions - AabX(a <==> b), DbaX(b –a–> a), and Jacb(a–c–> b). Models are stored with additional information such as equilibria, stability, zeroness, multiplicity of the equilibria. A model can be searched by propensity and signature string tokens it contains using DOCSS. The models are also stored in SBML format.

Fig. 4.1 (taken from [32]) depicts a visualization of all the bistable configurations in DOCSS. Each node in the graph represents a bistable configuration and the edges between the nodes correspond to subset relationships between the nodes; if there is an edge between two nodes, the reactions for a node are a subset of reactions for the other node. The color of nodes in the graph denotes the propensity of the nodes. Roughly speaking, propensity of a node indicates the ease with which we can find parameter values to make the configuration bistable. The nodes are positioned depending on their size. Texts on the leftmost column denote size of the nodes in each tier. Thus, 3x2 indicates that the nodes in that tier have three molecules and two reactions. As can be seen in the figure, the configurations form a forest of trees and a large portion of the forest is covered by a tree rooted from a 3x2 node. The root of this tree is a configuration called M101 and has '|DabX|Jbca|' as its signature. DOCSS not only provides bistable configurations but also their structural relation so can be used to find structural patterns of bistable systems.

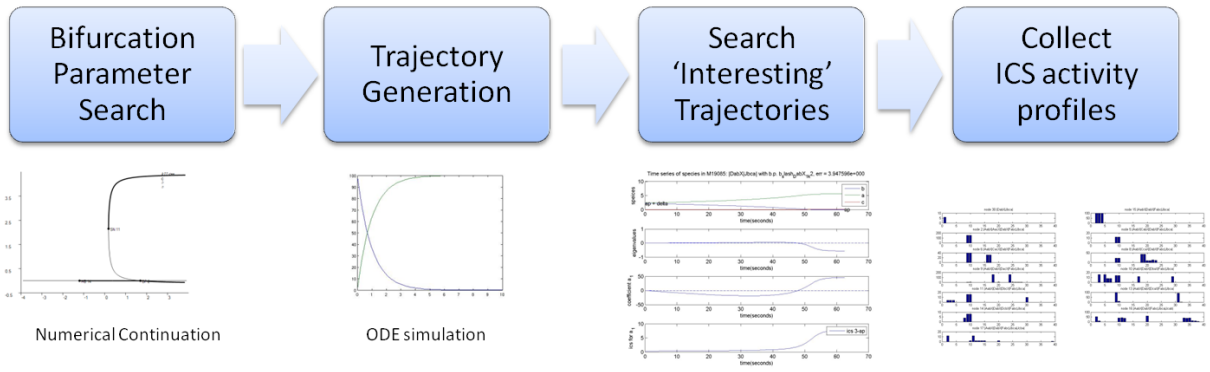


Figure 4.2: Collection of ICS activity profiles

The approach we use to tackle our core finding problem is a two step process.

For a tree of bistable CRNs:
Step 1. Prepare activity profiles of ICSs over trajectories for transitions from one stable state to another.
Step 2. Find substructures or rules retained in the CRNs from the activity profile.

4.2.2 Collection of ICS activity profiles

The first step thus involves collecting activity profiles of ICSs.

This step consists of 4 sub processes. Figure 4.2 provides an overview of these subprocesses. First, 1-parameter bifurcations are searched for, using a numerical continuation package. The found parameter is used to induce switching from a stable state to another stable state in the next step. The generated trajectories are examined according to some criteria to filter trajectories of interest. In the last step in this stage, algebraic terms corresponding to ICSs over the selected trajectories are evaluated. The details of each process are explained in the following. M101 with signature '|DabX|Jbca|' which is the root node of the largest subtree in 4.1 is used as an example for the explanation.

In [32], a number of bistable CRNs are found. Table 4.2 shows one of the found bistable CRNs called M101. M101 consists of two catalytic reactions: $b \xrightarrow{c} c$ and $c \xrightarrow{a} b$. Each reaction consists of three elementary steps as shown in the table. cc and ac are enzyme substrate complexes of $b \xrightarrow{c} c$ and $c \xrightarrow{a} b$, respectively.

	Elementary Step	Catalytic Reaction
r_1	$b + c \xrightarrow{k_1} cc$	$b \xrightarrow{c} c$
r_2	$cc \xrightarrow{k_2} b + c$	$b \xrightarrow{c} c$
r_3	$cc \xrightarrow{k_3} 2c$	$b \xrightarrow{c} c$
r_4	$c + a \xrightarrow{k_4} ac$	$c \xrightarrow{a} b$
r_5	$ac \xrightarrow{k_5} c + a$	$c \xrightarrow{a} b$
r_6	$ac \xrightarrow{k_6} b + a$	$c \xrightarrow{a} b$

Table 4.2: Reactions of M101

For $x = \begin{pmatrix} [b] & [a] & [c] & [cc] & [ac] \end{pmatrix}^T$, the ODE model of M101 is

$$\begin{aligned} \dot{x} &= N \cdot v(x) \\ &= \begin{pmatrix} -1 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & -1 & 1 & 1 \\ -1 & 1 & 2 & -1 & 1 & 0 \\ 1 & -1 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & -1 & -1 \end{pmatrix} \begin{pmatrix} k_1[b][c] \\ k_2[cc] \\ k_3[cc] \\ k_4[c][a] \\ k_5[ac] \\ k_6[ac] \end{pmatrix} \end{aligned}$$

The model can be replaced with a DAE by applying conservation analysis on the model. Conserved moieties of M101 are

$$\begin{aligned} C_1 &= 0.5[b] - 0.5[a] + 0.5[c] + [cc] \\ C_2 &= [a] + [ac] \end{aligned}$$

In case of selecting ac and cc as the dependent species,

$$\begin{pmatrix} \dot{[b]} \\ \dot{[c]} \\ \dot{[a]} \end{pmatrix} = \begin{pmatrix} -1 & 1 & 0 & 0 & 0 & 1 \\ -1 & 1 & 2 & -1 & 1 & 0 \\ 0 & 0 & 0 & -1 & 1 & 1 \end{pmatrix} \begin{pmatrix} k_1[b][c] \\ k_2[cc] \\ k_3[ac] \\ k_4[c][a] \\ k_5[ac] \\ k_6[ac] \end{pmatrix}$$

and the conservation equations forms the DAE model.
The DAE model is used to find bifurcation parameters of the model.

The first process of the collection step, as desribed earlier, is the search for a bifurcation parameter.

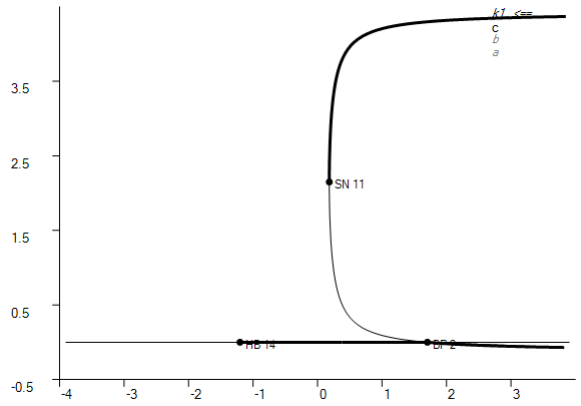


Figure 4.3: 1-parameter bifurcation diagram of M101 with bifurcation parameter k_1

1-parameter limit point bifurcation are searched using numerical continuation package called MATCONT[12]. MATCONT is a MATLAB based numerical continuation package using Moore-Penrose continuation algorithm and supports automatic detection of singularities such as equilibrium, limit point, hopf, limit cycle, periodic doubling, and limit point cycle [12]. This package can find a curve of equilibria for an input ODE model. ODE model $\dot{x} = N \cdot v(x)$ can be extended by including parameter α such that $\dot{x} = N \cdot v(x, \alpha)$ where α is one of the reaction rate coefficients. MATCONT can find (x, α) in the vicinity of (x_0, α_0) satisfying $N \cdot v(x, \alpha) = 0$ with varying α given $(x_0,\ \alpha_0)$. $\alpha$is called an active parameter. A curve of equilibria can be found by repeating this step until the number of repetitions or *alpha* reaches to a preset limit. For each model, reaction rate coefficients and one of steady states from DOCSS are used to set a run of the curve generation. For all reaction rate coefficients, a curve is generated with the steady state and one of reaction rate coefficients as the starting point and checked whether it has any fold.

For the example model, k_1 is one of the found bifurcation parameters with limit point bifurcation. Figure 4.3 shows a bifurcation diagram of M101 with k_1 being the bifurcation parameter. In the diagram, x axis is k_1, and y axis is c in steady state. Thickness of the curves in the diagram shows stability of the steady states; thick/thin curve corresponds to stable/unstable steady states. Saddle node bifurcation can be seen in the diagram at $k_1 = 0.18$. Decreasing value of k_1 from the found fold can induce a transition from high c state and low c state.

The next processes for the collection step are generation and selection of trajectories.

The found bifurcation parameters with the saddle nodes are used to generate trajectories of switching transitions with ODE simulations. The trajectory is generated by perturbing the bifurcation parameters around the saddle nodes and checked the trajectories are for

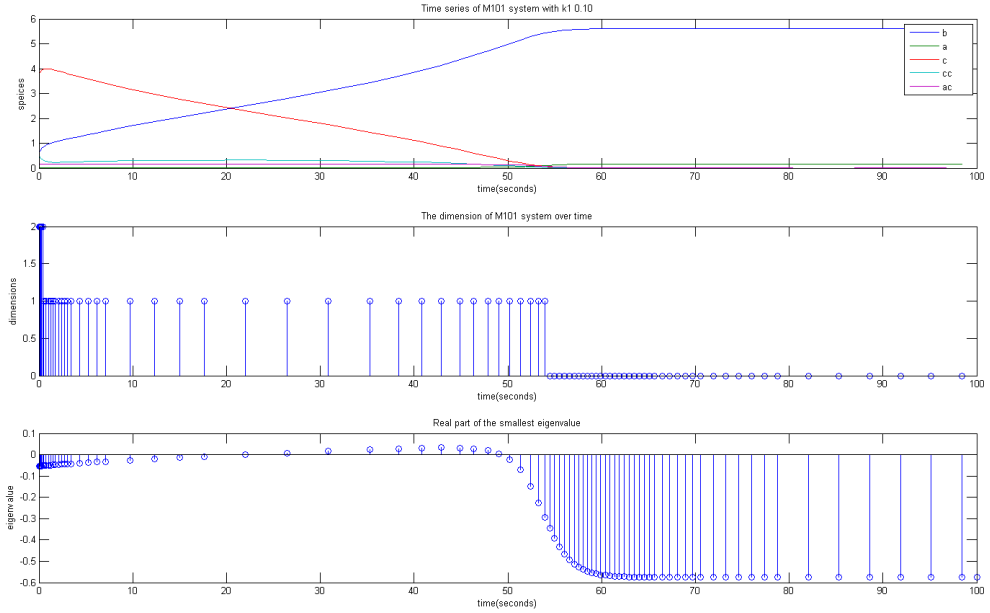


Figure 4.4: A trajectory of M101 showing a state transition from high c to low c

irreversible transitions between stable steady states.

The generated trajectories are filtered to find transitions co-occurred with violation of Routh-Hurwitz stability criterion. ICSA explained in Chapter 2 is performed for this filtering process. As has been discussed in Chapter 2, ICSA finds negative terms in coefficients of characteristic polynomial and relates the found terms with circuits in the input CRN. ICSA is performed with SBToolbox2[35] with MATLAB Symbolic Toolbox. SBML model of a input CRN is fed into SBToolbox2 to calculate a symbolic expression of Jacobian. MATLAB Symbolic Toolbox are used for major tasks of ICSA such as computing the characteristic polynomial of the Jacobian and finding negative terms in the symbolic expression of the coefficients of the characteristic polynomial.

The algebraic terms of the calculated ICSs are evaluated over the generated trajectories and used for the selection. Trajectories not satisfying any of the following conditions are filtered out:
For a characteristic equation $\lambda^n + a_1\lambda^{n-1} + a_2\lambda^{n-2} + ... + a_{n-1}\lambda + a_n = 0$ of Jacobian J,
a) there exists a root of the equation whose real part is positive.
b) there exists a coefficient $a_i < 0$ $(n \leq i \leq 1)$ of the equation over the trajectory.
c) there exists a non-constant ICS belong to a_i from condition b)
Condition a) and b) are directly from Routh-Hurwitz stability criterion and c) is to make sure there exists a non constant ICS over the trajectory. A trajectory satisfying these three

conditions is called 'interesting' trajectory and an ICS meeting condition c) is defined as an active ICS of the trajectory.

Figure 4.4 shows a trajectory, change in dimension of slow manifold and the smallest eigenvalue of M101 during the state transition. The system is in the stable steady state on the upper branch of the curve at $k_1 = 0.38$ and k_1 is decreased to 0.11. The system undergoes a transition from high c state to low c state. It can be easily seen that the transition is consistent with the bifurcation diagram. The plot in the center shows the change in the number of active time scale modes. As can be seen in the plot, the number of active modes is sharply increased at the beginning of the transition and gradually collapsed as the system reaches to the steady state. The change in real part of the smallest eigenvalues is displayed in the bottom plot. There is at least one positive eigenvalue in time interval (20,50). Time scale mode associated with a positive eigenvalue is always active and there is only one active mode in the interval. From this observation, it can be said that the time scale mode responsible for the gradual decrease of c in the time interval is the one associated with the positive eigenvalue.

ICS	Constraint	Circuit	Algebraic Expression
$v_{1,c} * v_{3,b}$	a_2	$b \xrightarrow{-} c \xrightarrow{-} b$	$(k_1 * [b]) * k_3$
$v_{2,c} * v_{3,b}$	a_2	$b \xrightarrow{-} c \xrightarrow{-} b$	$k_2 * k_3$
$v_{2,a} * v_{3,b} * v_{4,c}$	a_3	$a \xrightarrow{+} b \xrightarrow{-} c \xrightarrow{-} a$	$k_2 * k_3 * (k_4 * [a])$
$v_{3,b} * v_{4,c} * v_{6,a}$	$det(H_2)$	$a \xrightarrow{+} b \xrightarrow{-} c \xrightarrow{-} a$	$k_3 * (k_4 * [a]) * k_6$

Table 4.3: Instability causing structures of M101. $v_{x,y} = \frac{\partial v_x}{\partial y}$ is the partial derivative of the reaction rate of r_x to $[y]$

Table 4.3 shows ICSs found in M101. The first column is for the found ICSs. Each ICS belongs to a constraint derived from Routh-Hurwitz stability criterion. The second column in the table shows the constraints associated with the found ICSs. As discussed ealier, an ICS corresponds to a union of disjoint circuits and such circuits are given in the third column. $a \xrightarrow{+/-} b$ in the column represents that increase in a stimulates/inhibits production of b. For example, $b \xrightarrow{-} c \xrightarrow{-} b$ denotes mutual inhibition between b and c. The last column shows algebraic expressions calculated from the ODE model of M101.

Figure 4.5 shows a trajectory of ICSs in M101 generated by evaluating the algebraic terms for the ICSs over one of the selected trajectories. Only a_2 becomes negative for the trajectory so only ICSs from $a_2 > 0$ need to be checked to see which ICS is associated with the positive eigenvalue. Figure 4.5 illustrates the changes of the smallest eigenvalue, a_2, and ICSs from a_2 for the transition trajectory. As can be seen in the plots in the top and center of the figure, the interval with a positive eigenvalue matches with the interval of negative a_2 as has been expected. The plot in the bottom of the figure shows the change in value of $v_{1,c} * v_{3,b}$

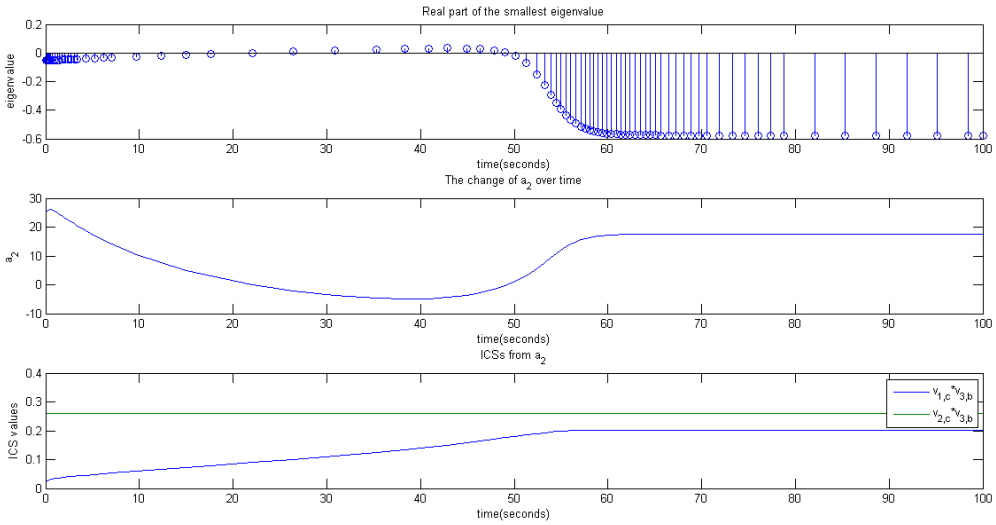


Figure 4.5: ICSs from $a_2 > 0$

and of $v_{2,c} * v_{3,b}$. Since $v_{2,c} * v_{3,b}$ is constant, the change in $v_{1,c} * v_{3,b}$ can be thought as the ICS driving the state transition. $v_{1,c} * v_{3,b}$ corresponds to circuit $b \xrightarrow{-} c \xrightarrow{-} b$. It is interesting that k_1 controls activation of this circuit and happens to be the bifurcation parameter which induce the transition.

The last processes for this step is collecting profiles of active ICSs of the selected 'interesting' trajectories from the previous process. An ICS corresponds to a set of disjoint circuits as has been discussed before. For a collection of CRNs with a common set of species, active ICSs for 'interesting' trajectories of the CRNs can be mapped to their matching circuits. Let C be the set of circuits and D be a sequence of all elements in C. Then, an ICS can be represented with a $|C|$ dimensional binary vector whose element in position i indicates whether or not the ICS contains the circuit in position i in sequence D.

Since any set of matching circuits for an 'interesting' trajectory is a subset of C, active ICSs for all 'interesting' trajectories can be compiled into a n by $|C|$ matrix whose row vectors are binary indicator vectors for instances of n 'interesting' trajectories. Such a matrix for all instances of 'interesting' trajectories is filled in this process and passed to Step 2. of whole process.

4.2.3 Analysis of ICS activity profiles

Step 2 of the process is to find any interesting patterns or rules in data from the profile collection step. One of the objectives in this step is to find conditions for a CRN to have

an 'interesting' trajectory. Decision rules for a model with a reaction signature to have such trajectories is searched using PART[14]. More complex algorithms such as those used in redescription mining [33, 30, 46] can be employed here but for the purposes of this dissertation we focus on rules such as can be captured in the form of sequential decision lists.

PART is an algorithm for learning decision lists from labeled training data. It builds a C4.5 decision tree and extracts a path with best coverage in the tree. The data points covered by the path are removed from the training data and the same process is repeated on the remaining data until whole data points are covered. The extracted paths in each iteration are output decision lists for the training data. In this case, each data point is a CRN with a specific reaction signature and the attributes for the classification are reaction tokens in the signature. The training data is labeled depending on whether or not there exits any 'interesting' trajectory found for all models with the reaction signature.

AabX	Fabc	AacX	CabX	...	Interesting
0	0	0	0	...	Yes
1	1	0	0	...	No
1	1	1	0	...	Yes
...	...	...	...	...	...

Table 4.4: Input data format of PART classifier in Weka[17]

We use the implementation of PART algorithm from Weka[17]. The attributes and class tags are converted into input data format of Weka. Table 4.4 illustrates the input data format. Each row of the table is corresponding a CRN of a reaction signature. All columns but the last column is a binary vector indicating whether or not the string token in the table header is in the reaction signature. The last column shows which class the CRN belongs to.

Another objective of this step is to analyze active ICSs for systems with 'interesting' trajectories. Such distributions can be obtained from the ICS profile matrix from the profile collection step. Distributions of active ICSs are manually inspected to find any substructure retained in the models with interesting trajectory.

4.3 Limitations and Possible Solutions

As has been discussed in Chapter 2, symbolic expressions of the coefficients of a characteristic equation needs to be calculated for ICSA. Bocher's formula introduced in the same chapter can be used for the purpose. Naive implementation of Bocher's formula involves calculation of symbolic expression for trace of J^i where i is the maximum order of interactions to be considered for ICSA. The calculation is computationally not feasible even for small i in case of J being a dense matrix.

One of the possible solutions is to utilize the time scale separation to reduce J. QE and QSS conditions can be found using classical and modified version of ILDM [37]. The found condition might be useful for the reduction of J thus decreasing computational cost of ICSA.

4.4 Experimental Results

Bistable systems including |DabX|Jbca| in their signature and propensity $>$.01 are searched in DOCSS. SBML files of the searched systems are converted into input files for continuation package, ODE solver, and structural analysis.

4.4.1 Conditions of a system having 'interesting' trajectory

The decision lists for a bistable system to have a 'interesting' trajectory are found with PART algorithm on the searched bistable configurations as has been discussed earlier. 6,230 bistable configurations are tested. Limit point bifurcations were found in 2,061 configurations of the configurations. 3,0540 'interesting' trajectories are collected for the configurations. The number of distinct string tokens of reactions signatures of the input configurations is 29 so the binary vector for the attributes has 29 dimensions. The number of configurations in 'interesting'/'not interesting' classes were originally 1,149/29,391. Since severe imbalance in data makes the classifier learned trivial, the data is preprocessed using the SMOTE[13] filter in Weka with 2,000% oversampling for 'interesting' instances to balance 'interesting'/'not interesting' instances ratio. SMOTE is an oversampling method to generate synthetic samples from k-nearest neighbors.

L_1 is a decision list for configurations with 'interesting' trajectories learned with PART:
L_1: Fabc $\bigwedge$ ¬DacX $\bigwedge$ ¬CacX $\bigwedge$ ¬DbaX $\bigwedge$ ¬CabX $\bigwedge$ ¬Jbac $\bigwedge$ ¬DbcX $\bigwedge$ ¬Jcba $\bigwedge$ ¬Jcab $\bigwedge$ ¬DcbX $\bigwedge$ ¬CcaX
The decision list is a conjunction of logical expressions of having or not having a string token in their reaction signature. For example, Fabc and ¬DacX in L_1 represents conditions of having Fabc and not having DacX in their reaction signature. Table 4.5 shows such conditions in L_1 and their matching reactions.

L_2, L_3,L_4,L_5, and L_6 are decision lists for configurations with no 'interesting' trajectory:
L_2: ¬AabX $\bigwedge$ ¬Fbac
L_3: ¬Fabc $\bigwedge$ ¬DbcX
L_4: ¬DbaX $\bigwedge$ ¬CabX $\bigwedge$ ¬Jbac $\bigwedge$ ¬Fbac
L_5: Fbac $\bigwedge$ ¬AabX $\bigwedge$ ¬DcbX
L_6: ¬Fbac

Table 4.6 shows number of bistable configurations covered by each decision list. As can be seen in the table L_1 and L_2 covers most of configurations with a 'interesting'/not 'interesting'

Conditions	matching reaction
Fabc	$2a \longleftrightarrow b + c$
¬DacX	$a \xrightarrow{c} c$
¬CacX	$a \xrightarrow{a} c$
¬DbaX	$b \xrightarrow{a} a$
¬CabX	$a \xrightarrow{a} b$
¬Jbac	$b \xrightarrow{a} c$
¬DbcX	$b \xrightarrow{c} c$
¬Jcba	$c \xrightarrow{b} a$
¬Jcab	$c \xrightarrow{a} b$
¬DcbX	$c \xrightarrow{b} b$
¬CcaX	$c \xrightarrow{c} a$

Table 4.5: Conditions of configurations with a 'interesting' trajectory

List	# of correctly covered	# of incorrectly covered
L_1	23,374	247
L_2	24,262	259
L_3	2,288	54
L_4	2,340	351
L_5	788	182
L_6	260	0

Table 4.6: Number of covered CRNs by the decision lists

trajectory.

The found rules are evaluated by 10 fold cross validation. Precision/recall and the confusion matrix of the classifier can be seen in Table 4.7. As has been demonstrated in the table, decision rules of whether a bistable configuration with DabX and Jbca in its signature has 'interesting' trajectory or not are found with high precision/recall. From these results, we can see that we are able to identify rules for a bistable configuration to have switching transitions that co-occur with violation of Routh-Hurwitz stability criteria.

a)

Precision	Recall
0.978	0.978

b)

		Actual class	
		Yes	No
Predicted class	Yes	29,142	249
	No	918	23,211

Table 4.7: Evaluation metrics of the classifier: a)Precision/recall, b)Confusion matrix

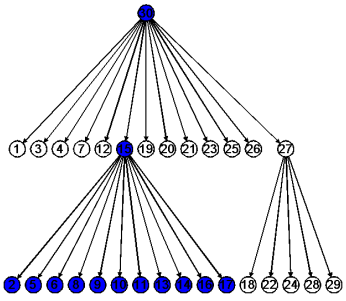


Figure 4.6: DAG of reaction signatures having a 'interesting' trajectory

4.4.2 Analysis of ICS activity profile on configurations with 'interesting' trajectories

Bistable configurations having a 'interesting' trajectory have 30 distinct reaction signatures. For a given set of reaction signature R, subset relation can be defined as: $\{(A,B) \in R \times R \mid$ every reaction token of A is also a reaction tokens of $B\}$.

The subset relation for a set of the reaction signatures of bistable configurations with a 'interesting' trajectory is transitively reduced and visualized in Figure 4.6. A node in the tree represents a reaction signature and an edge $A \rightarrow B$ denotes an element of subset relation (A,B). The DAG forms a tree as can be seen in the figure.

The result on subtree colored in blue is explained in detail in this section. As has been discussed in Approach section, the profile matrix is collected from all 'interesting' trajectories' found for the subtree. The minimal set of disjoint circuits covering all active ICSs with reaction signatures in the subtree has 39 elements so dimension of column space of the profile matix is 39. The distribution of circuits for a node in the tree can be obtained by summation of row vectors corresponding the node in the profile matrix. Figure 4.7 illustrates distributions of circuits in active ICSs for the subtree. Each subplot visualizes the distribution of circuits for a node in the subtree. X-axis is column indexes of the profile matix and y-axis is the number of 'interesting' trajectories with the circuit found for the node. For example, the upper left subplot in the figure shows the distribution of circuits

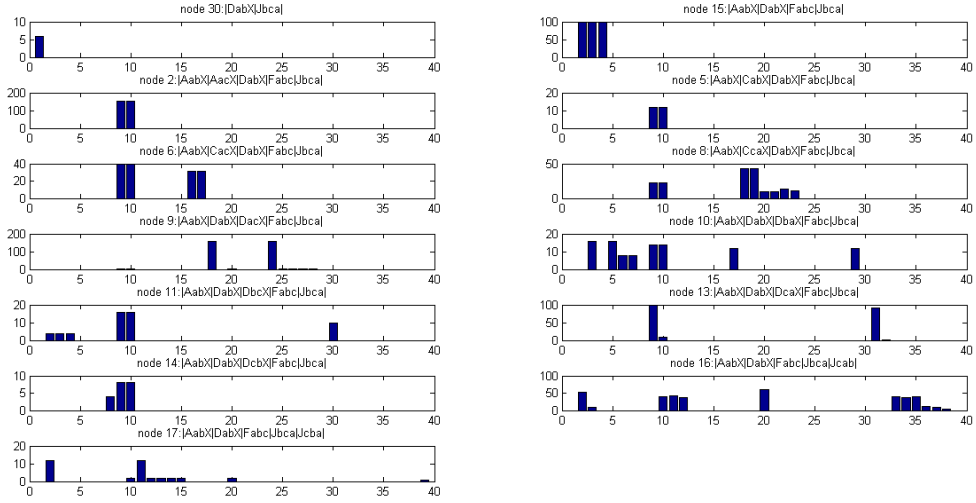


Figure 4.7: Distributions of circuits in active ICS for the subtree

in active ICSs for the trajectories of CRNs with |DabX|Jbca|. The distributions show that there are circuits retained in the subtree such circuit 9 and 10.

Table 4.8 shows a list of circuits found for nodes in the subtree. The first and second columns of the table are for node id and reaction signature of a node. The third column is for set difference between reaction tokens for a node and its parent and the last column is filed with ids of the circuits found in active ICSs for the node. A string token in parenthesis next to a circuit id is string representation of the circuit. As can be seen in the table, no circuit is shared between root node (node 30) and the rest of the nodes in the tree. There is also no node with 3 reactions having 'interesting' trajectory in the subtree. One of observations that can be made from tis table is that circuits 9(a->b_DabX-|a) and 4(a->b_DabX-|a), which are contained in the root node (|DabX|Jbca|), also cover the majority of the nodes in the subtree. Circuit 10 (b-|c-|b) also covers most of the nodes but it is ambiguous where it comes from. This shows that there are retained substructures shared within a lineage of CRNs. Node 6 is also interesting. Circuit 16 (a-|a_CacX-|a) is from newly added reaction CacX. This is an example of a situation where adding a reaction results in a new active ICS. In overall, our results demonstrate that ICS activity profiles can be used to find retained substructures within bistable configurations and to track how such substructures are affected by addition of new reactions.

node ID	signature	added reactions	circuit
30	|DabX|Jbca|	-	1(a-|b-|c->a)
15	|AabX|DabX|Fabc|Jbca|	AabX, Fabc	2,3,4 (b-|b, c-|c, a->b_DabX-|a)
2	|AabX|AacX|DabX|Fabc|Jbca|	AacX	9,10 (a-|b_DabX->a, b-|c-|b)
5	|AabX|CabX|DabX|Fabc|Jbca|	CabX	9,10
6	|AabX|CacX|DabX|Fabc|Jbca|	CacX	9,10,16,17(, , a-|a_CacX-|a, b->c-|b)
8	|AabX|CcaX|DabX|Fabc|Jbca|	CcaX	9, 10, 18, 19, 20, 21, 22, 23
9	|AabX|DabX|DacX|Fabc|Jbca|	DacX	9, 10, 18, 20, 24, 25, 26, 27, 28
10	|AabX|DabX|DbaX|Fabc|Jbca|	DbaX	3, 5, 6, 7, 9, 10, 17, 29
11	|AabX|DabX|DbcX|Fabc|Jbca|	DbcX	2, 3, 4, 9, 10, 30
13	|AabX|DabX|DcaX|Fabc|Jbca|	DcaX	9, 10, 31, 32
14	|AabX|DabX|DcbX|Fabc|Jbca|	DcbX	8, 9, 10

Table 4.8: Circuits of active ICSs for nodes in the subtree

4.5 Discussion

We have presented a streamlined way to discover and find patterns in core structures of a complex bistable chemical reaction system by combining numerical methods with data mining approaches. Our algorithm can contribute to large scale simulation based studies on patterns of switching-inducing core structure of bistable chemical reaction systems.

Chapter 5

Conclusions

Mechanisms of key biological processes are being revealed in detail thanks to the recent advance in system biology area. Such processes are often modeled with CRNs, reproducing up-to-date quantitative and qualitative behaviors of the modeled process as close as possible. Modeling of a biological process is traditionally performed by a domain expert utilizing data collected from various sources. As we aim to capture more complex details of biological machinery, the models are getting increasingly larger, thus posing challenges to the modeling process itself and for users to obtain intuitive understandings of the CRNs.

This thesis addressed two major research problems to address these challenges. The first problem pertains to automatic construction of a CRN model from time course data. Its importance lies not only in automation of the modeling process but also in being an aid to finding a simple model reproducing the data close enough. We have shown how such reconstruction is possible with systematic planning of perturbation experiments on the ODE model of a CRN.

The second problem pertains to network comprehension. Even a seemingly simple CRN model can show unexpected bifurcations in certain conditions and computer simulation has been an important tool of bifurcation analysis. Bistability is one of the important motifs to model key functions in biological process such as decision making. Switching mechanisms in such key functions are often modeled with saddle node bifurcation in a bistable system and there have been many efforts on finding simple substructures of a CRN responsible for the switching mechanism.

In this thesis, a type of such switching cores called instability causing structures (ICSs) have been studied using computer simulations. An automated process to find instability causing structures from a CRN has been proposed and applied over a large database of bistable CRNs. It is shown that there exist highly convincing inclusion/exclusion rules of reactions determining a bistable CRN with active ICSs and saddle node bifurcations. Such CRNs are found to form a single tree structure defined with subset relations in its reactions. The found

ICSs are compared across the CRNs in the tree and ICSs retained in most of the found tree.

We now outline some directions for future research.

The approach for network reconstruction has limitations in identifying multiple reactions between the same set of molecules and in the assumptions of rate laws that it makes. Adoption of numerical optimization techniques on parametized models using sensitivity profiles can be one of the approaches to tackle these limitations. Another related problem is reducing the cost of the reconstruction process by efficient planning of the buffering experiments. The current approach assumes all buffering experiments have the same cost but the cost of a buffering experiment can be highly variable depending on the type of buffered species and the amount of the species. One of the possible solutions to be explored is to organize buffering experiment hierarchically. First, buffering experiments are performed for all pairs of high cost species with all other species in the buffering set. By analyzing the experimental results, the species in the CRN can be divided into two groups having no direct/indirect interactions between the groups. Markov blankets of species in each group should be either the low cost species in the buffering set or species within the same group; i.e., buffering high cost species in the other group is not necessary to find Markov blankets of species in the current group, thus reducing the number of experiments with high cost species in their buffering sets. Sensitivity of the reconstruction process to noise in the data is another problem that can be explored.

We now turn to future directions for the bistable core finding problem. The current approach might not scale well due to explosion of combinations of interactions to consider for the search. The activity of ICSs is directly related to time scale modes as has been discussed, and time scale separation can be used to prune candidate circuits for the ICS search. The inclusion/exclusion rules found consist of reactions between molecules of specific symbols and might not be easy to apply for an arbitrary biological process model, because we will need to explore all possible instantations of these symbols. One of directions for future research can be adaptation of graph mining techniques to bridge this gap and to find more intuitive conditions. A study on how combination of the found bistable switches affects the activity profile of ICSs of each switch is also likely to be interesting.

Finally, we can generalize the type of phenomena being studied from bistability to oscillations, another key motif used in modeling a biological process. The adopted approach is also applicable to motifs of oscillation with some modification. A large scale simulation study on a database of oscillators can be undertaken.

In overall, this thesis portends well for the combination of data mining and numerical computing methods for understanding the behavior of large, complex, biochemical systems. These methods inform and complement each other and can become a valuable resource to not just computer scientists but to a range of inter-disciplinary practitioners including chemists, systems biologists, evolutionary biologists, mathematicians, and modelers, all of whom study cellular decision making in various guises.

Bibliography

[1] N. Allen, L. Calzone, K. Chen, A. Ciliberto, N. Ramakrishnan, C. Shaffer, J. Sible, J. Tyson, M. Vass, L. Watson, and J. Zwolak. Modeling Regulatory Networks at Virginia Tech. *OMICS A Journal of Integrative Biology*, Vol. 7(3):pages 285–300, Fall 2003.

[2] N. Allen, C. Shaffer, N. Ramakrishnan, M. Vass, and L. Watson. Improving the Development Process for Eukaryotic Cell Cycle Models with a Modeling Support Environment. *Simulation*, Vol. 79(12):674–688, Dec 2003.

[3] D. Angeli, J. E. Ferrell, and E. D. Sontag. Detection of multistability, bifurcations, and hysteresis in a large class of biological positive-feedback systems. *Proc Natl Acad Sci U S A*, 101(7):1822–1827, 2004.

[4] A. Arkin, P. Shen, and J. Ross. A Test Case of Correlation Metric Construction of a Reaction Pathway from Measurements. *Science*, Vol. 277(5330):1275–1279, Aug 1997.

[5] U. Bhalla. Understanding Complex Signaling Networks through Models and Metaphors. *Progress in Biophysics and Molecular Biology*, Vol. 81(1):45–65, Jan 2003.

[6] F. Bromberg, D. Margaritis, and V. Honavar. Efficient Markov Network Structure Discovery using Independence Tests. In *Proceedings of the Sixth SIAM International Conference on Data Mining*. SIAM Press, 2006.

[7] P. Brown, A. Hindmarsh, and L. Petzold. Using Krylov Methods in the Solution of Large-Scale Differential-Algebraic Systems. *SIAM Journal of Scientific Computing*, Vol. 15:1467–1488, 1994.

[8] K. Chen, L. Calzone, F. Csikasz-Nagy, F. Cross, B. Novak, and J. Tyson. Integrative Analysis of Cell Cycle Control in Budding Yeast. *Molecular Biology of the Cell*, Vol. 15:3841–3862, Aug 2004.

[9] S. Cohen and A. Hindmarsh. CVODE, A Stiff/Nonstiff ODE Solver in C. *Computers in Physics*, Vol. 10(2):138–143, Mar-Apr 1996.

[10] H. Conzelmann, J. Saez-Rodriguez, T. Sauter, E. Bullinger, F. Allgower, and E. Gilles. Reduction of mathematical models of signal transduction networks: simulation-based approach applied to egf receptor signalling. *Syst Biol (Stevenage)*, 1:159 – 169, 2004.

[11] A. Csikasz-Nagy, D. Battogtokh, K. Chen, B. Novak, and J. Tyson. Analysis of a generic model of eukaryotic cell cycle regulation. *Biophys. J.*, Vol. 90:4361–4379, 2006.

[12] A. Dhooge, W. Govaerts, and Y. A. Kuznetsov. Matcont: A matlab package for numerical bifurcation analysis of odes. *ACM Trans. Math. Softw.*, 29(2):141–164, June 2003.

[13] N. V. C. et. al. Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, 16:321–357, 2002.

[14] E. Frank and I. H. Witten. Generating accurate rule sets without global optimization. In *ICML*, pages 144–151, 1998.

[15] C. Furusawa, T. Suzuki, A. Kashiwagi, T. Yomo, and K. Kaneko. Ubiquity of log-normal distributions in intra-cellular reaction dynamics. *BIOPHYSICS*, 1:25–31, 2005.

[16] A. Goldbeter and D. Koshland. An amplified sensitivity arising from covalent modification in biological systems. *Proceedings of The National Academy of Sciences*, 78:6840–6844, 1981.

[17] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten. The weka data mining software: an update. *SIGKDD Explor. Newsl.*, 11(1):10–18, Nov. 2009.

[18] M. L. Huma and S. H. Muggleton. *Elements of Computational Systems Biology*. Wiley, February 2010.

[19] A. Hurwitz. Ueber die bedingungen, unter welchen eine gleichung nur wurzeln mit negativen reellen theilen besitzt. *Springer Berlin / Heidelberg*, 42(2):273, June 1895.

[20] A. Karnaukhov and E. Karnaukhova. System Identification in Biophysics: A New Method based on Minimizing Square Residuals. *Biofizika*, Vol. 49(1):88–97, 2004.

[21] A. Karnaukhov, E. Karnaukhova, and J. Williamson. Numerical Matrices Method for Nonlinear System Identification and Description of Dynamics of Biochemical Reaction Networks. *Biophysical Journal*, Vol. 92:3459–3473, 2007.

[22] S. Klamt and E. Gilles. Minimal Cut Sets in Biochemical Reaction Networks. *Bioinformatics*, Vol. 20(2):226–234, 2004.

[23] E. Klipp, R. Herwig, A. Kowald, C. Wierling, and H. Lehrach. *Systems Biology in Practice*. Wiley-VCH, May 2005.

[24] S. Lauritzen. *Graphical Models*. Oxford University Press, 1996.

[25] A. Lovrics, A. Csikasz-Nagy, I. Zsely, J. Zador, T. Turanyi, and B. Novak. Time scale and dimension analysis of a budding yeast cell cycle model. *BMC Bioinformatics*, 7(1):494, 2006.

[26] U. Maas and S. Pope. Simplifying chemical kinetics- intrinsic low-dimensional manifolds in composition space. *Combustion and Flame*, 88:239, 1992.

[27] W. Marwan, A. Wagler, and R. Weismantel. A Mathematical Approach to Solve the Network Reconstruction Problem. *Mathematical Methods in Operations Research*, Vol. 67:117–132, 2008.

[28] M. Maurya, S. Bornheimer, V. Venkatasubramanian, and S. Subramaniam. Reduced-order Modelling of Biochemical Networks: Application to the GTpase-Cycle Signalling Module. *IEE Systems Biology*, Vol. 152(4):229–242, Dec 2005.

[29] M. S. Okino and M. L. Mavrovouniotis. Simplification of mathematical models of chemical reaction systems. *Chemical Reviews*, 98(2):391–408, 1998.

[30] L. Parida and N. Ramakrishnan. Redescription Mining: Structure Theory and Algorithms. In *Proceedings of the Twentieth National Conference on Artificial Intelligence*, pages 837–844, 2005.

[31] N. Ramakrishnan, U. Bhalla, and J. Tyson. Computing with Proteins. *IEEE Computer*, Vol. 42(1):47–56, Jan 2009.

[32] N. Ramakrishnan and U. S. Bhalla. Memory switches in chemical reaction space. *PLoS Computational Biology*, 4(7):e1000122, 07 2008.

[33] N. Ramakrishnan, D. Kumar, B. Mishra, M. Potts, and R. Helm. Turning CARTwheels: An Alternating Algorithm for Mining Redescriptions. In *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 266–275, 2004.

[34] J. Ross, I. Schreiber, and M. Vlad. *Determination of Complex Reaction Mechanisms: Analysis of Chemical, Biological, and Genetic Networks*. Oxford University Press, Nov 2005.

[35] H. Schmidt and M. Jirstrand. Systems biology toolbox for matlab: a computational platform for research in systems biology. *Bioinformatics*, 22(4):514–515, February 2006.

[36] W. Sha, J. Moore, K. Chen, A. Lassaletta, C.-S. Yi, J. Tyson, and J. Sible. Hysteresis drives Cell-cycle Transitions in *Xenopus laevis* Egg Extracts. *PNAS*, Vol. 100(3):975–980, Feb 2003.

[37] I. Surovtsova, N. Simus, T. Lorenz, A. König, S. Sahle, and U. Kummer. Accessible methods for the dynamic time-scale decomposition of biochemical systems. *Bioinformatics*, 25:2816–2823, November 2009.

[38] R. Thomas and M. Kaufman. Frontier diagrams: Partition of phase space according to the signs of the eigenvalues or to the sign pattern of the circuits. *Int. J. of Bifurcation & Chaos*, 15(11), 2005.

[39] A. S. Tomlin, L. Whitehouse, R. Lowe, and M. J. Pilling. Low-dimensional manifolds in tropospheric chemical systems. *Faraday Discussions*, 120:125–146, 2002.

[40] J. Tyson. Modeling the Cell Division Cycle: cdc2 and Cyclin Interactions. *PNAS*, Vol. 88(16):7328–7332, Aug 1991.

[41] J. J. Tyson, K. C. Chen, and B. Novak. Sniffers, buzzers, toggles and blinkers: dynamics of regulatory and signaling pathways in the cell. *Current Opinion in Cell Biology*, 15(2):221–231, 2003.

[42] M. Vass, N. Allen, C. Shaffer, N. Ramakrishnan, L. Watson, and J. Tyson. The JigCell Model Builder and Run Manager. *Bioinformatics*, Vol. 20(18):3680–3681, Dec 2004.

[43] C. Wiggins and I. Nemenman. Process Pathway Inference via Time Series Analysis. *Experimental Mechanics*, Vol. 43(3):361–370, Sep 2003.

[44] T. Wilhelm. Analysis of structures causing instabilities. *Phys. Rev. E*, 76(1):011911, Jul 2007.

[45] D. M. Wolf and A. P. Arkin. Motifs, modules and games in bacteria. *Current Opinion in Microbiology*, 6(2):125–134, 2003.

[46] L. Zhao, M. Zaki, and N. Ramakrishnan. BLOSOM: A Framework for Mining Boolean Expressions. In *Proceedings of the Twelfth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'06)*, pages 827–832, 2006.