Probabilistic Approximation and Analysis Techniques for Bio-Pathway Models

Liu Bing

(*B.Comp.(Hons.), NUS*)

A Thesis submitted for the degree of

Doctor of Philosophy

NUS Graduate School for Integrative Sciences and Engineering

National University of Singapore

2010

Acknowledgements

I would like to express my sincerest gratitude to everybody who helped me throughout my time at NUS.

First and foremost, I am deeply grateful to my supervisors, Professor P.S. Thiagarajan and Associate Professor David Hsu. At the beginning, their recommendations helped me to successfully join NGS and to start my academic journey. Over the past four years, I have benefited tremendously from their excellent guidance, persistent support, and invaluable advices. Working with them was extremely pleasant. I have learnt a lot from them in many aspects of doing research. Their enthusiasm, dedication and preciseness have deeply influenced me as being a researcher. In addition, I appreciate their financial support during the period of my thesis writing.

Part of this thesis is a joint work with Professor Ding Jeak Lings group from Department of Biological Sciences. I am grateful to Prof Ding for her constant guidance and patience as well as her impressive inputs to our paper. I also thank all of the rest collaborators including Associate Professor Ho Bow from Department of Department of Pathology, Doctors Benjamin Leong and Sunil Sethi from National University Hospital, and Professor Anna Blom from Lund University for their valuable suggestions and assistance in paper writing. Special thanks to Zhang Jing, who has been closely working with me for two years on this project and has contributed numerous wet-lab experimental data.

I would also like to thank our current collaborators Associate Professor Wong Weng Fai from Department of Computer Science, and Associate Professor Marie-Veronique Clement from Department of Biochemistry. Thank them for the fruitful discussions that might lead to extensions and applications of this work.

I thank Professor Shazib Perviaz, who is a member of my thesis advisory committee, for his constant support as well as the suggestions on my quantification exam. I thank Professor Wong Limsoon, the coordinator of our lab, for providing me the research facilities. I am also grateful to Associate Professor Sung Wing Kin for his help on my application for the research assistantship.

I will always appreciate the friendship and support of our current and former group members: Geoffery Koh, Lisa Tucker-Kellogg, Yang Shaofa, Sucheendra Palaniappan, Joshua Chin Yen Song, Wang Junjie, Gireedhar Venkatachalam, Abhinav Dubey, Benjamin Gyori, Akshay Sundararaman, and many others. Thank them for the open, collaborative and friendly environment as well as the countless useful discussions. Special thank to Geoffery who is always a role model to me. I have learnt a lot from him. I thank Lisa for the useful discussions and suggestions. I also thank Shaofa for his advices on thesis writing.

I also want to thank my lab-mates, class-mates and friends: Dong Difeng, Koh Chuan Hook, Chiang Tsong-Han, Chen Jin; Ren Jie, Zheng Yantao, Zhao Pan, Sun Wei, Wu Zhaoxuan, Li Guangda, Huan Xuelu, Ming Zhaoyan, Huang Hua, Liu Chengcheng and Xu Jia; Wu Huayu, Liu Ning, Zhou Weiguang, Xue Mingqiang, Bao Zhifeng, Xu Liang, Pan Miao, Shi Yuan, Zhai Boxuan, Meng Lingsha, Yang Peipei, Liu Shuning, Liu Feng, Li Yan, Yin Lu, and many others. I would like to express my sincerest gratitude to them for being kind, friendly, and fun. My time at NUS has been wonderful because of all of them.

Finally, I want to thank my family. Thanks to my cousins, Liu Mei and Cai Xiaoming, and my uncle and auntie, Liu Yingzhu and Wang Runzhi, for their care and support in Singapore. I am deeply indebted to my parents for their unconditional love and to my wife Han Zheng for her understanding, support, and loving care. Their love is the source of the happiness in my life.

Contents

Summary

The cell is the building block of life. Understanding how cells work is a major challenge. Cellular processes are governed and coordinated by a multitude of biological pathways, each of which can be viewed as a complex network of biochemical reactions involving biomolecules (proteins, metabolite, RNAs). Thus it is necessary to have a system-level understating of cellular functions and behavior and to so, one must develop quantitative models.

Currently, a widely used means of modeling biological pathways is a system of ordinary differential equations (ODEs). Since biological pathways are often complex and involve a large number of reactions, the corresponding ODE systems will not admit closed form solutions. Hence to analyze the pathway dynamics one will have to use numerical simulations. However, the number of simulations required to carry out model calibration and analysis tasks can become very large due to the following facts: Models often contain many unknown parameters (rate constants in the differential equations and initial concentration levels). Estimating their values will require a large number of simulations. This also happens when performing tasks such as global sensitivity analysis that involve sampling the high-dimensional value space induced by model parameters. Further, the experimental data used for training and testing the model are often cell population-based and have limited precision. Consequently, to simulate the model and compare with such data, one must resort to Monte Carlo methods to ensure that sufficiently many values from the distribution of model parameters are being sampled.

A major contribution of this thesis is to develop a computational approach by which one can approximate the pathway dynamics defined by a system of ODEs as a dynamic Bayesian network. Using this approximation, one can then efficiently carry out model calibration and analysis tasks. Broadly speaking, our approach consists of the following steps: (i) discretize the value space and the time domain; (ii) sample the initial states of the system according to an assumed prior distribution; (iii) generate a trajectory for each sampled initial state and view the resulting set of trajectories as an approximation of the dynamics defined by the ODEs system; (iv) store the generated set

of trajectories compactly as a dynamic Bayesian network and use Bayesian inference techniques to perform analysis. This method has several advantages. Firstly, the discretized nature of the approximation helps to bridge the gap between the accuracy of the results obtained by ODE simulation and the limited precision of experimental data used for calibration and validation. Secondly and more importantly, after investing in this one-time construction cost, many interesting pathway properties can be analyzed efficiently through standard Bayesian inference techniques instead of resorting to a large number of ODE simulations.

We have demonstrated the applicability of our technique with the help of three case studies. First, we tested our method on an EGF-NGF signaling pathway model (Brown et al., 2004). We constructed the DBN approximation and used synthetic data to perform parameter estimation and global sensitivity analysis. The results show improved performance easily amortizing the cost of constructing the approximation. It also is sufficiently accurate given the lack of precision and noise in the experimental data. We further demonstrated this in the second case study using a segmentation clock pathway model taken from Goldbeter and Pourquie (2008).

In the third case study, we built and analyzed a pathway model of the complement system consisting of the lectin and classical pathways in collaboration with biologists and clinicians (Liu et al., 2010). Using our approximation technique, we efficiently trained the DBN model on *in vivo* experimental data and explored the key network features. Our combined computational and experimental study showed that the antimicrobial response is sensitive to changes in pH and calcium levels, which determines the strength of the crosstalk between two receptors called CRP and L-ficolin. Our study also revealed differential regulatory effects of the inhibitor C4BP. While C4BP delays but does not attenuate the classical pathway, it attenuates but does not delay the lectin pathway. Further, we found that the major inhibitory role of C4BP is to facilitate the decay of C3 convertase. These results elucidate the regulatory mechanisms of the complement system and potentially contribute to the development of complement-based immunomodulation therapies.

List of Figures

List of Tables

Chapter 1

Introduction

Cells are the basic units of life. Understanding how cells function is one of the greatest challenges facing science. The rewards of success will range from better medical therapies to new generation of biofuels. Over the past decades, numerous experimental techniques, such as microscopy, polymerase chain reaction (PCR), western blot, flow cytometry and fluorescence resonance energy transfer (FRET), have been developed to help biologists to investigate how cells work. Consequently, biology has made amazing advances on characterizing components inside the cell as well as identifying their interactions. These components are often referred as *biomolecules*, including large molecules such as proteins, DNA, RNA, and polysaccharides, as well as small molecules such as metabolites, sugars, lipids, vitamins, and hormones. The cell is like a hugely complex machine consisting of millions of such basic parts, which are interacting with each other and carrying out diverse cellular functions.

Conventional biology research, which focuses on identifying components and interactions inside the cell, culminates in the emerging of a variety of fields of studies with the suffix *-omics*, such as genomics, proteomics, metabolomics, lipidomics, and interactomics. These fields aim to describe and integrate complete sets of knowledge about biomolecules, resulting in a range of biological databases including gene databases

such as Entrez[1] and GeneCards[2], protein databases such as UniProt[3] and PDB[4], as well as the protein-protein interaction databases such as BioGRID[5] and BIND[6]. Hence, roughly speaking, we already have a general picture of the basic constituents of the cell. However, it is still far from an in-depth understanding of cellular processes, because biomolecules do not function alone but exist in highly regulated complex assemblies and networks. The next step in this line of research is to develop a systematic view of how cells work, how cellular processes are regulated, and how cells response to their changing internal and external environments.

This has motivated the emerging domain of *systems biology* that seeks to understand how the individual biomolecules interact and evolve in time and space to realize the various cellular functions. Systems biology integrates many different disciplines such as biology, mathematics, physics, chemistry, computer science, and engineering. A long-term vision of this field is to put all the relevant biological processes together and build a model that can simulate the whole cell or even an entire organism. Such models will have a substantial impact on our health care, food supplies and many other issues that are essential to our survival. It will not only lead to a better understanding of physiological mechanisms and human diseases, but also bring about more efficient drug development and validation processes. Furthermore, with the help of models, we may also engineer cells to have desired properties for biotechnological production of food, fuel and materials.

[1] `http://www.ncbi.nlm.nih.gov/gquery/gquery.fcgi`
[2] `http://www.genecards.org/`
[3] `http://www.uniprot.org/`
[4] `http://www.rcsb.org/pdb/`
[5] `http://www.thebiogrid.org/`
[6] `http://www.bind.ca/`

1.1 Context and Motivation

To achieve the long-term vision of systems biology, one must describe fundamental intra- and intercellular processes. The cellular processes are driven by networks of biochemical reactions, which have been termed *biological pathways*. This thesis focuses on modeling and analyzing the dynamics of biological pathways. Among the current modeling formalisms, a system of ordinary differential equations (ODEs) is the most widely used one to model pathway dynamics (Aldridge et al., 2006; Materi and Wishart, 2007). In the past few decades, many ODE models have been developed to study pathways governing various cellular functions ranging from cell cycle to cell death (Marlovits et al., 1998; Legewie et al., 2006). Due to the popularity of ODE-based modeling, standard markup languages such as SBML (Hucka et al., 2003) have been proposed for efficient model exchange and reuse. Hundreds of software systems were developed for editing, simulating and storing models. For instance, the BioModels database (Le Novere et al., 2006; Li et al., 2010) archives more than 200 published ODE models covering many of the known biological pathways.

ODE models enable many kinds of model analysis, such as sensitivity, perturbation, and population-based analysis that can be performed by solving the ODEs with different initial conditions and parameters. For instance, Spencer et al. (2009) discovered that the difference in initial concentrations of proteins regulating apoptosis signaling pathways is the primary cause of the cell-to-cell variability in the timing and probability of cell death, which may explain why only a fraction of tumor cells will be killed after exposure to chemotherapy. Another striking example is by Lee et al. (2007), who used ODE models to significantly increase the productivity of L-threonine, an amino acid that has been widely used in industries of cosmetics and pharmacy.

The ODE-based modeling has become a major approach in systems biology. However, to gain success in practical applications, there are several challenges to be addressed.

- **Large-scale pathways.** Biological pathways are often complex and involve a large number of biochemical reactions (Weng et al., 1999; Lauffenburger, 2000). For example, the ErbB signaling pathway model built by Chen et al. (2009) consisting of 828 reactions among 499 species. Hence the corresponding systems of ODEs will not admit closed form solutions. Instead, one will have to use numerical integration methods such as Runge-Kutta to perform model simulations as well as analysis. The challenge here is that numerically simulating high dimensional ODE systems will be computational intensive.

- **Experimental data.** Experimental data will be needed for the model development. Assuming parameter values are known, analysis will consist of comparing simulated behavior with experimental data. However, the data generated will only have very limited precision. Specifically, the initial concentration levels of the various proteins and rate constants will often be available only as *intervals* of values. Further, experimental data in terms of the concentration levels of a few proteins at a small number of time points will also be available only in terms of intervals of values. In addition, the data will often be gathered using a population of cells. Hence the data will represent the average concentration levels of proteins in many different cells. Consequently, when numerically simulating the ODE model, one must resort to Monte Carlo methods to ensure that sufficiently many values from the relevant intervals are being sampled. As a result, generating a single prediction to compare with the experimental data will require doing a large number of simulations.

- **Parameter estimation.** The execution of simulation requires the values of model parameters to be known. Large pathway models often possess many unknown parameters which have to be estimated from the training data. A common approach to parameter estimation is via optimizing the agreement between the

model prediction and the training data. Since there are many unknown parameters, the induced search space will be high-dimensional and contain many local minima. Hence one will have to use global methods such as evolutionary strategies. In order to find a good solution, global methods often evaluate many combinations of parameter values. An evaluation is done by simulating the whole system and computing the error between the model prediction and the experimental data. As a result, parameter estimation will require also doing a large number of simulations. Further, if the population data with limited precision, as mentioned above, is used as training data, even more simulations will be needed.

- **Model analysis.** Many kinds of model analysis require doing a large number of simulations as well. A few examples will be reviewed in Section 2.5, including global sensitivity analysis, perturbation optimization and population-based analysis. Specifically, the global sensitivity analysis assesses the overall effects of parameters on the model output by simultaneously perturbing all the parameters within a parameter space. It often follows a Monte Carlo scheme: simulate the system for a large number of combinations of parameter values and derive the global sensitivities by statistically analyzing the simulation results. Perturbation optimization aims to find the best perturbation to fulfill certain design goals such as maximizing the production of a biochemical substance, while minimizing the formation of undesirable byproducts. Due to the combinatorial nature of the problem, the solution spaces of large models will contain a huge number of candidate perturbations. Consequently, similar to parameter estimation, finding the best perturbation will require doing a large number of simulations.

1.2 Our Approach and Contributions

ODE models are prevalent for modeling biological pathways. However, as pointed out above, carrying out model calibration and analysis on large pathways will require a large number of simulations, which is very computational expensive. This motivates our main goal, namely, to approximate the dynamics of systems of ODEs modeling biological pathways.

In this thesis, we propose an approach by which one can approximate the ODE dynamics as a dynamic Bayesian network (DBN) (Murphy, 2002). As a result, tasks such as parameter estimation and global sensitivity analysis can be efficiently carried out through standard Bayesian inference techniques. Our techniques can be adapted to modeling formalisms such as hybrid functional Petri nets (Matsuno et al., 2003b) as well.

1.2.1 The Approximation Technique

Given a system of ODEs, we assume that the dynamics is of interest only for a finite time horizon and that the states of the system are to be observed only at a finite set of discrete time points. Next we partition the range of each variable into a finite set of intervals according to the assumed observation precision. We also discretize the range of each parameter into a finite set of intervals. The initial values as well as the parameters of the ODE system are assumed as distributions (usually uniform) over the intervals defined by the discretization. For unknown parameters, we assume they are uniformly distributed within their ranges.

After fixing the discretization and the distribution of initial states, we sample the initial states of the system (i.e. a vector which assigns an initial value for each variable and parameter) and generate a trajectory by numerical integration for each of the sampled initial states. The key idea is that a sufficiently large set of such trajectories is a good approximation of the dynamics defined by the ODEs system.

The second key idea is that this set of trajectories or rather, the statistical properties of these trajectories can be compactly stored in the form of a dynamic Bayesian network (DBN) (Murphy, 2002) by exploiting the network structure of the pathway and simple counting. As a result, by querying this DBN representation using standard inferencing techniques one can analyze, in a probabilistic and approximate fashion, the dynamics defined by the system of ODEs.

The construction process consists of two steps: (i) derive the underlying graph of the DBN approximation by exploiting the structure of the ODEs, (ii) fill up the entries of the conditional probability tables associated with the nodes of the DBN by sampling the prior distributions, performing numerical integration for each sample, discretizing generated trajectories by the predefined intervals and computing the conditional probabilities by simple counting.

Since the trajectories are grouped together through the discretization, our method bridges the gap between the accuracy of the results obtained by ODE simulation and the limited precision of experimental data used for model development. In addition, the approximation represents the dependencies between the variables more explicitly in the graph structure of the underlying DBN. More crucially, many interesting pathway properties can be analyzed efficiently through standard Bayesian inference techniques, instead of resorting to large scale numerical simulations. Here we present a few examples informally:

- **Probabilistic inference.** Given initial state as evidence, the Bayesian inference technique called the Factored Frontier algorithm (Murphy and Weiss, 2001) can be used to approximately but efficiently infer the marginal probability of each species' concentration at a given time point.

- **Parameter estimation.** Our approximation approach enables a two-stage parameter estimation method. In the first stage, we infer the marginal distributions

of the species at different points in the DBN. The mean of each marginal distribution are computed in order to compare with the time serials training data. Standard optimization methods are used for searching in the discretized parameter space. The result of this first stage is a maximum likelihood estimate of a combination of intervals of parameter values. In the second stage, by treating the resulting combination of intervals of parameter values from the first stage as the (drastically reduced) search space, one can further estimate the real values for unknown parameters. The second stage results in parameters with a finer granularity, which can be used to perform simulations and analysis requiring perturbing the initial concentrations.

- **Global sensitivity analysis.** We can use DBN approximation to perform global sensitivity analysis. Monte Carlo samples are drawn from the discretized parameter space. Simulation trajectories will be approximated by the mean of marginal distributions inferred from the DBN by supplying the selected combination of intervals of parameter values as evidence.

Admittedly, there is a one-time computational cost incurred to construct the DBN approximation. But this cost can be easily amortized by performing multiple analysis tasks using the DBN approximation. This will be demonstrated by studying two existing pathway taken from Brown et al. (2004) and Goldbeter and Pourquie (2008) and a "live" pathway called complement system in collaboration with biologists and clinicians (Liu et al., 2010).

Our work is, in spirit, related to the discretized approximations presented in Calder et al. (2006b,c); Ciocchetta et al. (2009) that are based on stochastic modeling formalisms such as PEPA (Hillston, 1996) and the modeling language PRISM (Kwiatkowska et al., 2002). In these works, the dynamics of a process-algebra-based description of a biological pathway is given in terms of a Continuous Time Markov Chain (CTMC)

which is then discretized using the notion of *levels* to ease analysis. Apart from the fact that our starting point is a system of ODEs, a crucial additional step that we take is to exploit the structure of the pathway to factor the dynamics into a DBN. We then perform analysis tasks on this more compact representation. In a similar vein, our model is more compact than the graphical model of a network of non-homogenous Markov chains studied in Nodelman et al. (2002).

For sure, our DBN approximation may be viewed as a factored Markov chain. In this sense, a crucial component of our construction mirrors the technique of factoring a Hidden Markov Model (HMM) as a DBN by decomposing a system state into its constituent variables (Russell and Norvig, 2003). This connection leads us to believe that the techniques proposed in Langmead et al. (2006a), as well as the verification techniques reported in Clarke et al. (2008); Heath et al. (2008) can be adapted to our setting. Analyzing CTMC models PEPA requires stochastic simulations that are often computationaly intensive Geisweiller et al. (2008). We note however the DBN approximation is a probabilistic graphical model and hence we do not have to resort to stochastic simulations. The inferencing algorithm we use (the Factored Frontier algorithm (Murphy and Weiss, 2001)), in one sweep, gathers information about the statistical properties of the family of trajectories encoded by the DBN approximation.

1.2.2 The Biological Contributions

The complement system is key to innate immunity and its activation is necessary for the clearance of bacteria and apoptotic cells. However, insufficient or excessive complement activation will lead to immune-related diseases. It is so far unknown how the complement activity is up- or down- regulated and what the associated pathophysiological mechanisms are. To quantitatively understand the modulatory mechanisms of the complement system, we built a computational model involving the enhancement and suppression mechanisms that regulate complement activity. Our model consists

of 42 species, 45 reactions and 85 kinetic parameters with 71 of the parameters being unknown. The ODE model is accompanied by a DBN as a probabilistic approximation of the ODE dynamics. We used the DBN approximation to perform parameter estimation and sensitivity analysis. Our combined computational and experimental study highlights the importance of infection-mediated microenvironmental perturbations, which alter the pH and calcium levels. It also reveals that the inhibitor, C4BP induces differential inhibition on the classical and lectin complement pathways and acts mainly by facilitating the decay of the C3 convertase. These predictions were validated empirically. Thus our results help to elucidate the regulatory mechanisms of the complement system and potentially contribute to the development of complement-based immunomodulation therapies.

1.3 Outline

The rest of this thesis is organized as follows.

In Chapter 2 we give an overview of the current state of pathway modeling. We present the background knowledge on biological pathways and discuss the process of pathway modeling. We then review several formalisms that are commonly used to model the pathway dynamics. We also describe some existing methods for parameter estimation. Further, we present two useful model analysis techniques.

Chapters 3-5 form the core of the work, in which we present our probabilistic approximation technique. After introducing the preliminaries in Chapter 3, we describe our method for constructing the DBN approximation in Chapter section 4. In Chapter 5, we present techniques for performing tasks such as basic inferencing, parameter estimation and global sensitivity analysis using the DBN approximation.

Chapter 6 establishes the applicability of probabilistic approximation techniques. In Section 6.1 and Section 6.2 we present two case studies on the EGF-NGF signaling pathway and the segmentation clock pathway respectively. We compare the efficiency of

our method to conventional approaches for parameter estimation and global sensitivity analysis. We also compare the performance of different sampling techniques and the accuracies of approximations constructed using different discretization schemes. In Section 6.3 we further demonstrate the usefulness of our method by an integrated computational and experimental study of the human complement system. We present our model constructed for the complement regulatory mechanisms. We also discuss the computational and experimental results as well as the biological insights we gained.

Finally, in Chapter 7, we summarize the main results and discuss the future lines of research.

1.4 Declaration

This thesis is based on the following material:

- "Probabilistic approximations of ODE-based bio-pathway dynamics", B. Liu, P.S. Thiagarajan, D. Hsu. *Theoretical Computer Science*. (accepted).
- "A computational and experimental study of the regulatory mechanisms of the complement system", B. Liu, J. Zhang, P. Y. Tan, D. Hsu, A. M. Blom, B. Leong, S. Sethi, B. Ho, J. L. Ding, P.S. Thiagarajan. *PLoS Computational Biology* (accepted).
- "Probabilistic approximations of signaling pathway dynamics", B. Liu, P.S. Thiagarajan, D. Hsu. *In Proc. of the 7th Computational Methods in Systems Biology (CMSB)*, 2009.
- "Probabilistic approximations of bio-pathway dynamics", B. Liu, D. Hsu, P.S. Thiagarajan. *In the 13th Annual International Conference on Research in Computational Molecular Biology (RECOMB) Poster Book*, 2009.

Chapter 2

Background and Related Work

In this chapter, we discuss the current state of bio-pathway modeling. After presenting the background knowledge, we review the processes of model construction, calibration, validation and analysis. We then discuss several formalisms that are used to capture pathway dynamics. Next we review some existing methods for model calibration. Finally, we present two useful model analysis techniques, namely, sensitivity analysis, and perturbation optimization.

2.1 Biological Pathways

Cellular processes are driven by networks of biochemical reactions, termed *biological pathways*. Biological pathways can be loosely classified into *signaling pathways*, *metabolic pathways*, and *gene regulatory networks*. Specifically:

- **Signaling pathways.** Signaling pathways describe how cells sense changes or stimuli in their environment, pass the received signals messages via cascades of biochemical reactions, and respond by modifying their metabolisms, transcriptional activities or cell fates. The chief actors in signaling pathways are proteins such as receptors, kinases, and transcription factors.

- **Metabolic pathways.** Metabolic pathways consist of chemical reactions involved in metabolism, through which cells acquire energy for survival and reproduction. The major players in metabolic pathways are chemical compounds such as glucose, adenosine diphosphate (ADP), and adenosine triphosphate (ATP).

- **Gene regulatory networks** The expression of a gene is highly regulated by transcription factors synthesized from other genes. Gene regulatory networks often abstract the reactions involved in the processes of DNA transcription, RNA translation, and post translation modification of proteins and depict the indirect regulatory relationship among genes in the cell.

The three classes of biological pathways describe different aspects of cellular processes. Cells rely on their tight cooperation to achieve proper functioning. In this thesis, we focus mainly on signaling pathways, though our techniques can be applied to metabolic pathways and gene regulatory networks as well.

Cellular processes are dynamic. In other words, the number of biomolecules such as protein concentrations, metabolite concentrations, and gene expression levels are changing over time. Hence the biological pathways can be viewed as dynamical systems, whose state is defined as a snapshot the quantity of involved species at a time point. The *dynamics* of biological pathways are crucial for cellular functions. A remarkable example is the biological pathway controlling the circadian rhythm (biological clock). The built-in circadian rhythm in our body regulates the daily cycles of many physiological processes such as the sleep-wake cycle and feeding rhythms (Bell-Pedersen et al., 2005). It arises from the oscillatory expression of a number of genes. The time profile of expression level of some related genes are shown in Figure 2.1. It can be observed that the periods of the oscillations roughly equal to 24 hours. The oscillations of gene expression are governed by the underlying signaling pathways. Figure 2.2 depicts the *Drosophila* circadian rhythm pathway proposed by Matsuno et al. (2003a). The oscillator is composed of interlocking feedback loops that regulate the concentrations of

transcription factors. These transcription factors further control the expression of many other genes, as the output of the oscillator, resulting in behavioral and physiological rhythms.

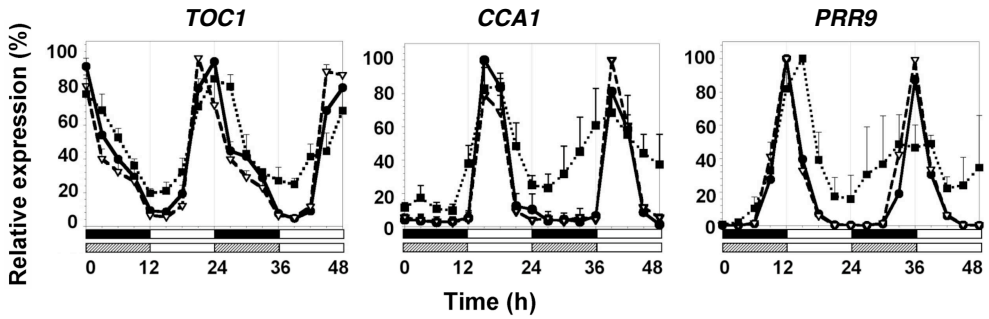


Figure 2.1: The expression of circadian rhythm related genes. This figure is reproduced from James et al. (2008).

There are hundreds of biological pathways governing various cellular processes ranging from cell cycle to cell death. Some of the heavily studied signaling pathways are summarized in Figure 2.3 (Lodish, 2003). For instance, apoptosis pathways induce the programmed cell death (Spencer et al., 2009). EGF/NGF signaling pathway determines the cell differentiation or cell proliferation (Kholodenko, 2007). Wnt signaling pathway governs the expression of developmental genes (Logan and Nusse, 2004). NF-κB pathway regulates inflammatory responses (Egan and Toruner, 2006). Similar to circadian rhythm pathway, these pathways often consist of many species and multiple feedback loops. Consequently, it is very difficult to predict the dynamical behavior of the system based on intuition. Hence one will have to resort to computational modeling.

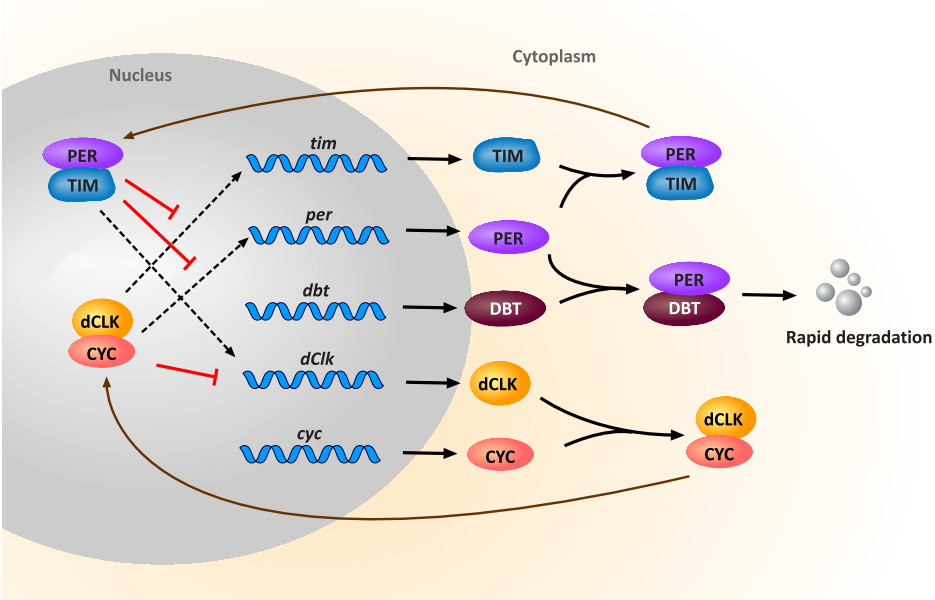


Figure 2.2: The *Drosophila* circadian rhythm pathway model. This figure is reproduced from Matsuno et al. (2003a).

2.2 Pathway Modeling

To study the complex dynamics of biological pathways, a variety of computational models have been proposed in recent years, ranging from qualitative models that focus on the generic properties of biological networks (Papin and Palsson, 2004; Helikar et al., 2008) to quantitative models that can simulate the time course of biological pathways under various conditions (Vaseghi et al., 2001). The choice of a modeling formalism depends on the goals of the modeling effort as well as the biological context. For instance, the Boolean network is a frequently used qualitative formalism (Fisher et al., 2007; Thakar et al., 2007), while typical quantitative formalisms are ordinary differential equations (ODEs) (Aldridge et al., 2006), Petri nets (Matsuno et al., 2003b), performance evaluation process algebra (PEPA) (Hillston, 1996), PRSIM (Kwiatkowska et al., 2002), and κ (Danos et al., 2007). On what follows, we focus mainly on the quantitate model.

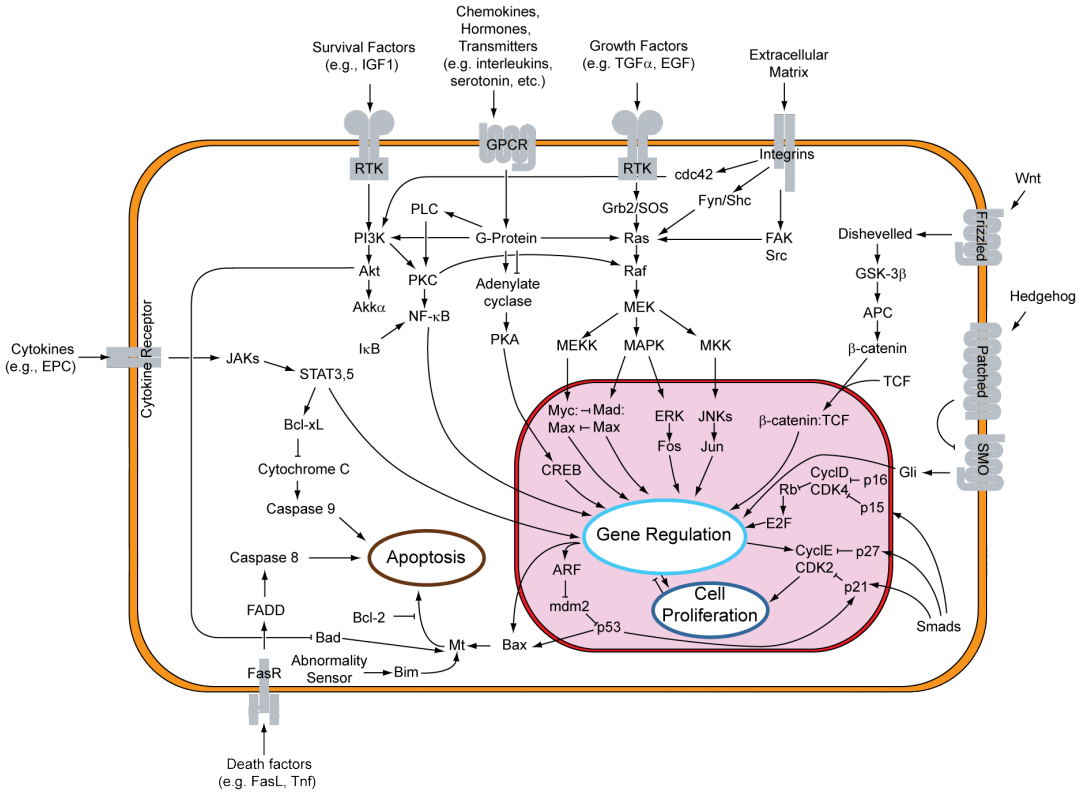


Figure 2.3: Overview of some of the important signaling pathways (Lodish, 2003)

Regardless of the the type of quantitative model used, a typical computational modeling effort involves the following steps:

1. **Model construction.** Decide the model scope and build the model structure by capturing the current knowledge of the pathway.

2. **Model calibration.** Divide the available experimental observations of the pathway dynamics into two parts -training data and test data- and calibrate the model parameters so that model predictions are able to reproduce the observations in the training data.

3. **Model validation.** Test the capability of a calibrated model by evaluating the fitness of model predictions to the test data. (The test and training data can be of different kinds. The key point is that the model must be validated using data that was not used for training it.)

4. **Model anlaysis.** Perform various kinds of analyses on the validated model in order to gain biological insights, reveal the network properties, and generate hypotheses.

In Step 1, an initial model can be constructed based on the literature as well as the pathway databases such as Reactome (Joshi-Tope et al., 2005). In this step one often requires the guidance of biologists. In Steps 2 and 3, the experimental data can include both quantitative and qualitative measurements. However, quantitative measurements of the time serials of species concentration are preferred for Step 2, as they may provide more constraints to the model. The calibration process of Step 2 is also known as parameter estimation, which will be discussed in detail in Section 2.4.

If the model predictions fit the training data in Step 2 and can be validated by test data in Step 3, we trust the model to be reasonably reliable and use it as a basis for analysis in Step 4. Simulation is a useful tool for performing model analysis. Through

simulations, one can observe the time profile of species or system behavior that have not been measured, or even can not be measured via current technology. Further, one can simulate the system under different conditions by modifying the model structure, initial condition or kinetic parameters. In this manner, one can carry out "what if?" experiments suggested by biologists through local modifications of the model. One can also apply techniques such as sensitivity, perturbation and population-based analysis etc. The corresponding wet-lab experiments will be, in general, very time consuming and expensive. They might not even be possible due to the unavailability of the needed bio-markers. In this sense, the model and its analysis techniques can serve as an additional tool, which biologists can use to perform extensive *in silico* experiments quickly and cheaply, in order to advance biological knowledge.

It is worth noting that, in practice, the process of model development may not simply follow a linear order of the above steps but often involve a cyclic workflow. For Step 2, if one is unable to find proper parameters so that the fitness between model predictions generated using the estimated parameters and training data is acceptable, one will have to go back to Step 1 and refine the model structure by adding further structural details which had been left out. Similarly, for Step 3, if the model cannot be validated, one could go back to Step 1 and improve the model. In addition, one could also try to acquire more experimental data concerning the structure and dynamics. But what if we still can not pass Step 2 and Step 3, when we already exhausted the resources? Interestingly, the failure in Step 2 or Step 3 might become a seedbed for generating hypotheses. By analyzing the mismatch between model prediction and the data, one may propose missing links, cross-talks, feedback loops, etc. of the pathway, which can guide biologists in their further investigations.

2.3 Modeling Formalisms

In this section, we present some of the well-established quantitative models for capturing and analyzing pathway dynamics.

2.3.1 Ordinary Differential Equations

Modeling biological pathway dynamics with ordinary differential equations (ODEs) is a major approach in current systems biology research (Materi and Wishart, 2007). The idea is to describe biochemical reactions such as biomolecular association and enzyme catalytic modification, using equations derived from physicochemical theories (Aldridge et al., 2006).

In the context of biological pathway modeling, one often uses t to denote time and x to denote the concentration level of individual biomolecular species. As a result, the function $x(t)$ will depict the time profile of species x while its derivative $\frac{dx}{dt}$ will represent the rate of change of x.

A biological pathway usually involves many species and can be viewed as a network of biochemical reactions. The rate of change of the concentration of each species in the network will be determined by the rates of reactions that produce or consume this species. Based on suitable assumptions, physical and chemical laws (such as mass action law, Michaelis-Menten law and power law) can be applied to calculating the reaction rates from the concentrations of their participating species. For example, under assumption the species are spatially homogeneous, the mass action law (Guldberg and Waage, 1879) states that the rate of a reaction is proportional to the concentrations of reacting species. Let's consider a reversible binding process of two species shown as follows:

$$A + B \underset{v_2}{\overset{v_1}{\rightleftharpoons}} AB \tag{2.1}$$

where A and B are substrates, AB denotes the formed complex, and v_1 and v_2 represents

the association rate and dissociation rate respectively. By the mass action law, we have:

$$v_1 = k_1 \cdot A \cdot B$$

$$v_2 = k_2 \cdot AB$$

where k_1 and k_2 are so-called rate constants.

The choice of a kinetics law depends on the nature of the reaction to be described. For example, the enzyme catalyzed reactions such as protein phosphorylation are often modeled using Michaelis-Menten equations. Equation 2.2 shows the reaction scheme of a typical enzyme catalyzed reaction.

$$S + E \xrightarrow{v} P + E \tag{2.2}$$

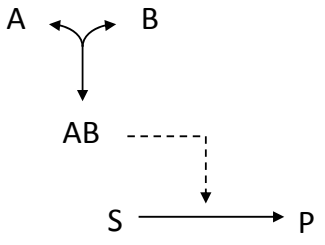
where S denotes substrate, E denotes enzyme, P denotes product and v denotes the reaction rate. By assuming that $S \gg E$, v can be expressed by the Michaelis-Menten equation as follows:

$$v = \frac{k \cdot S \cdot E}{K_m + S} \tag{2.3}$$

where k and K_m are constants.

Once we write down rate equations for all reactions in a network, the rate of change of each species can then be derived by summing all reaction rates that produce this species and subtracting all reaction rates that consume this species. As reaction rates are calculated from the concentrations of species and kinetic constants, the rate of change of a species x_i can be written as a function f_i, typically nonlinear, involving variables from $\{x_1, x_2, \ldots, x_n\}$ and parameters (rate constants) from $\{p_1, p_2, \ldots, p_m\}$. Consequently, a biological pathway can be modeled as a system of ODEs of the form:

$$\frac{dx_i}{dt} = f_i(\mathbf{x}(t), \mathbf{p}) \tag{2.4}$$



$$\frac{dA}{dt} = -k_1 \cdot A \cdot B + k_2 \cdot AB$$
$$\frac{dB}{dt} = -k_1 \cdot A \cdot B + k_2 \cdot AB$$
$$\frac{dAB}{dt} = k_1 \cdot A \cdot B - k_2 \cdot AB$$
$$\frac{dS}{dt} = -\frac{k \cdot S \cdot AB}{K_m + S}$$
$$\frac{dP}{dt} = \frac{k \cdot S \cdot AB}{K_m + S}$$

Figure 2.4: The ODE model of a small pathway.

where the vector $\mathbf{x}(t)$ represents the concentrations of species at time t, and the vector $\mathbf{p}$ refer to the rate constants of the reactions.

Example Consider a small pathway which links the assembly process described in Equation and the catalysis process described in Equation 2.2 by setting AB to be E (see Figure 2.4, left panel). The ODE model of this pathway is shown in the right panel of Figure 2.4.

Given the initial values of the variables and parameters (initial condition) and suitable continuity assumptions, a system of ODEs will have a unique solution specifying how the system will evolve over time (Hirsch et al., 2004). Hence models defined with ODEs can be used to produce predictions of system behavior by solving this *initial value problem*. However, the ODE systems describing biological pathway dynamics are usually high-dimensional and nonlinear. Hence they will not admit closed form solutions. Instead, one will have to resort to numerical integration methods to get approximate solutions. For example, *finite difference methods* numerically approximate the solutions of differential equations. The idea can be illustrated as follows. By definition we have

$$x'(t) = \lim_{\delta \to 0} \frac{x(t+\delta) - x(t)}{\delta}, \tag{2.5}$$

then a reasonable approximation of the derivative would be

$$x'(t) \approx \frac{x(t+\delta) - x(t)}{\delta} \tag{2.6}$$

for a sufficient small δ. Since $x'(t)$ is known, by giving initial condition $x(0)$, we can iteratively compute $x(t)$ for any t as follows:

$$x(t+\delta) = x(t) + \delta \cdot x'(t) \tag{2.7}$$

This is the so-called *Euler's Method*. To achieve high accuracy, it requires δ to be very small. Accordingly, for a fixed T, the maximal time point of interest, the required number of simulation steps T/δ will be a large number. As a result, solving large ODE system will be computationally intensive. In the past decades, many advanced ODE solvers have been developed to improve the performance of numerical integration. Different solvers are usually specialized for better performance on some classes of ODEs. To deal with the ODE systems of biological pathway models, methods such as Runge-Kutta (Hindmarsh, 1983) and LSODA (Petzold, 1983) have been used. For example, let $x'(t) = f(x(t))$ the formula of the fourth order Runge-Kutta (RK4) will be as follows:

$$\begin{aligned}
A_1 &= f(x(t)) \\
A_2 &= f(x(t) + \frac{1}{2} \cdot \delta \cdot A_1) \\
A_3 &= f(x(t) + \frac{1}{2} \cdot \delta \cdot A_2) \\
A_4 &= f(x(t) + \delta \cdot A_1) \\
x(t+\delta) &= x(t) + \frac{1}{6} \cdot \delta \cdot (A_1 + 2A_2 + 2A_3 + A_4)
\end{aligned}$$

However, it remains computationally expensive for solving large or stiff[1] ODE systems,

[1] A system of ODEs is said to be stiff if explicit numerical methods such as Runge-Kutta require very small step size to achieve the desired accuracy.

which are often unfortunately induced by biological pathway models.

Simplifications

To reduce the complexity of ODE-based pathway models, simplification methods have been proposed based on certain assumptions. First of all, during the model design process, assumptions can be made about the model scope. Species will be included in the model only if they are necessary for the target analysis. It is important to determine the degree of details so that the model constructed contains as few species and parameters as possible, while meeting the design goals. For example, nuclear localization of the transcriptional activator Nuclear factor κB (NF-κB) is controlled in mammalian cells by NF-κB inhibitor protein IκB, which has three isoforms: IκBα, IκBβ, and IκBϵ. Hoffmann et al. (2002) found that IκBα is responsible for strong negative feedback that allows for a fast turn-off of the NF-κB response, whereas IκBβ and IκBϵ function to reduce the system's oscillatory potential and stabilize NF-κB responses during longer stimulations (Hoffmann et al., 2002). Thus, their model includes all the three isoforms with corresponding reactions in order to understand their different roles. On other hand, in the model built by Cho et al. (2003), the three isoforms are treated as one protein since they only aim to analyze the sensitivity of parameters in TNFα-mediated NF-κB pathway and this will not be effected by the variation of IκB isoforms.

Secondly, one can simplify the ODE model by abstractions. In fact, the Michaelis-Metnten equation is obtained by abstracting mass action kinetics. By assuming that the concentration of substrate is much larger than the concentration of enzyme, it eliminates the unnecessary intermediate products and replace the original parameters that are hard to measure by fewer measurable ones (Klipp et al., 2005). The idea of Michaelis-Menten approximation has been extended by Schmidt et al. (2008) to deal with all rate expressions that can be written as a fraction between two polynomials. For instance, after applying their algorithm, complex rate equations such as the one

appear in (Teusink, 2000):

$$v_{original} = \frac{V(\frac{[F16bP]}{K_{F16bP}} - \frac{[DHAP][GAP]}{K_{F16bP}K_{Keq}})}{1 + \frac{[F16bP]}{K_{F16bP}} + \frac{[DHAP]}{K_{DHAP}} + \frac{[GAP]}{K_{GAP}} + \frac{[F16bP][GAP]}{K_{F16bP}K_{GAP}} + \frac{[DHAP][GAP]}{K_{DHAP}K_{GAP}}} \tag{2.8}$$

can be simplified as:

$$v_{simplified} = \frac{K_2[F16bP]}{1 + K_1[F16bP]}. \tag{2.9}$$

Applications

In the recent years, ODE based modeling has played a dominant role in systems biology. Numerous insights have been gained through simulating and analyzing ODE models. For example, Gallego et al. (2006) found that *tau* has an opposite role to what we believed in circadian rhythms. Sasagawa et al. (2005) showed that transient ERK activation depends on rapid increases of EGF and NGF but not on their final concentrations, whereas substained ERK activation depends on the final concentration of NGF but not on the temporal rate of increase. Spencer et al. (2009) discovered that differences in the levels of proteins regulating receptor-mediated apoptosis are the primary causes of cell-to-cell variability in the timing and probability of death in human cell lines. Basak et al. (2007) showed that mutant cells with altered balances between canonical and noncanonical IkB proteins may exhibit inappropriate inflammatory gene expression in response to developmental signals. With help of ODE models, all the above example studies generated very interesting and important hypotheses, which were confirmed or supported by further verification wet-lab experiments.

2.3.2 Petri Nets

Petri nets, originally proposed by Carl Adam Petri in 1962 (Petri, 1962), is a mathematical model for the representation and analysis of concurrent processes. It graphically depicts the structure of a concurrent system as a directed bipartite graph with annota-

tions. A Petri net consists of three primitive elements - places, transitions and directed arcs. In the context of bio-pathway modeling, places often denote species while transitions represent the biochemical reactions. The places are connected to the transitions (and vice versa) via directed arcs to form a network.

In the graphical representation, places are drawn as circles, transitions are denoted by bars or boxes, and arcs are labeled with their weights (positive integers), where a k-weighted can be interpreted as the set of k parallel arcs. The input places of a transition are the places from which an arc runs to it; its output places are those to which an arc runs from it.

Places may contain any nonnegative number of tokens, which are represented as block dots inside the corresponding place. A distribution of tokens over the places of a net is called a *marking*. Transitions can fire (i.e. execute) if they are enabled, which means there are enough tokens in every input place. When a transition fires, it consumes a number of tokens from each of its input places, and produces a numbers of tokens on each of its output places.

Example Figure 2.5 shows a Petri net model of the enzyme catalysis system. In this example, the places E, S, P denote the enzyme, product and substrate respectively. The transition T represents the enzyme catalyzed reaction. The number of tokens depicts the concentration level of a species. The initial marking is shown in the left panel of Figure 2.5. Transition T is enabled. After firing T once, the resulting marking is shown in the right panel of Figure 2.5.

Petri nets support a number of qualitative analysis for checking the topological properties of the network. To enable quantitative simulation and analysis, various types of Petri nets have been proposed by extending the original Petri net, such as *timed Petri nets*, *stochastic Petri nets*, *hybrid Petri nets*, and *functional Petri nets* (Reisig and Rozenberg, 1998). Many of them have been deployed for simulating the

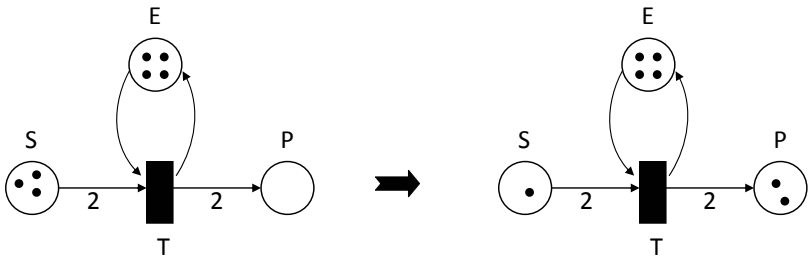


Figure 2.5: A Petri net example of the enzyme catalysis system.

dynamics of biological pathways. For instance, Ruths et al. (2008) studied a MAPK and AKT signaling network downstream from EGFR in two breast tumor cell lines using stochastic Petri net. Bonzanni et al. (2009) used a coarse-grained quantitative Petri net to mimic the multicelluar process of *Caenorhabditis elegans* vulval development. Additional Petri net models of biological pathways can be found in Chen and Hofestaedt (2003), Voss et al. (2003), Heiner et al. (2003), Koch et al. (2005), and Lee et al. (2006).

The Petri net-based approaches used in systems biology has been reviewed in Koch et al. (2010). Among various types of Petri nets, the *Hybrid Functional Petri net* (HFPN) (Matsuno et al., 2003c) is an useful approach that can capture both the discrete and continuous features of pathway dynamics. This variant has been implemented in a software tool called Cell Illustrator (Doi et al., 2003; Nagasaki et al., 2010), which has been used to model and analyze a number of biological pathways (Tasaki et al., 2010; Do et al., 2010; Li et al., 2009; Sato et al., 2009).

The HPFN inherits the notations of the hybrid Petri net (David and Alla, 1987) and the functional Petri net (Valk, 1978) and adds more functionality. As it can deal with both discrete and continuous components, two kinds of places and transitions are used (the graphical notation are shown in Figure 2.6).

A discrete place is the same as a place in Petri net, i.e. it can only hold integer

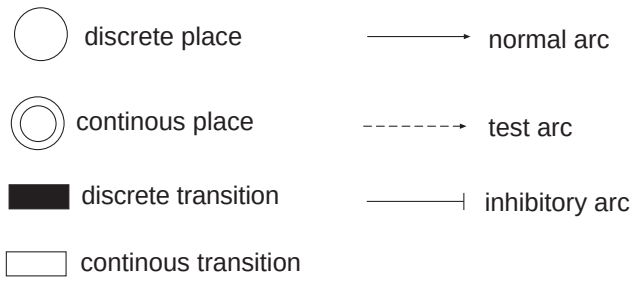


Figure 2.6: HFPN notations.

number of tokens. In other hand, a continuous place can hold non-negative real numbers as its content. For transitions, a discrete transition can only fire when its firing conditions are satisfied for certain duration of time, denoted by a *delay function*. In contrast, a continuous transition fires continuously in and its firing speed is given as a *firing function* of values at particular places in the model. The firing speed describes the consumption rate of its input places and the production rate of its output places.

In addition, there are two more kinds of arcs - the *inhibitory arc* and the *test arc* (Figure 2.6). An inhibitory arc with weight r enables the transition to fire only if the content of the place at the source of the arc is less than or equal to r. A test arc, behaves like a normal arc, except that it does not consume any content of the place at the source of the arc when it fires. Furthermore, there are also some restrictions for connection. For example, a discrete place cannot connect to a discrete place via a continuous transition. Test and inhibitory arcs are restricted to only connect incoming places to transitions as they both involve satisfying a precondition.

Example A HFPN model of the enzyme catalysis system is shown in Figure 2.7. In this example, the markings of the continuous places E, P, and S denote concentrations of the enzyme, product and substrate. The formula of the transition T specifies the rate equation of the enzyme catalyzed reaction. Let $k = 0.01$, $K_m = 10$, after firing T

for a time step $\delta = 1$, the resulting marking is shown in the right panel of Figure 2.7.

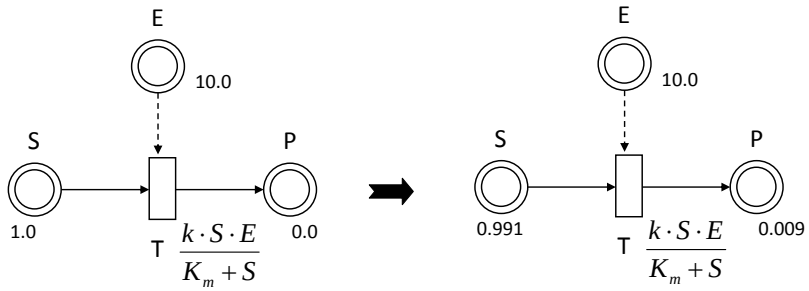


Figure 2.7: A Petri net example of the enzyme catalysis system.

Notice that the execution process of this HFPN is equivalent to solving the following ODE system using the Euler's method.

$$\frac{dS}{dt} = -\frac{k \cdot S \cdot E}{K_m + S}$$
$$\frac{dP}{dt} = \frac{k \cdot S \cdot E}{K_m + S}$$
$$\frac{dE}{dt} = 0$$

In this manner, any ODE-based pathway model can be translated into a HFPN model, which only contains continuous places and transitions as well as the arcs. Hence, the HFPN can be viewed as an extension of the ODE formulism with discrete aspects.

2.3.3 Stochastic Models

Deterministic models such as ODE and HFPN assume that the concentrations of involved species are sufficiently high and the molecules are uniformly distributed in cellular compartments. However, when the concentrations of species are low (e.g. dozens or hundreds), the variability of reaction processes will increase and may significantly influence the systems behavior. For example, the development of phage λ infected *E.*

coli cells (Arkin et al., 1998) is determined by a switch point. Two proteins with low concentration levels competitively control this switch. As a result, the developmental outcome is probabilistic and cannot be captured by conventional deterministic models. In such cases, stochastic modeling will be required.

In stochastic modeling, one often described the state of the system by a vector $X(t) = (X_1(t), X_2(t), \ldots, X_N(t))$, where $X_i(t)$ is a nonnegative integer which expresses the number of molecules of species i at time t. Starting from an initial state $X(0) = x_0$, $X(t)$ can evolve its value when a reaction takes place, which is a stochastic event.

By modeling the probabilities of occurrences of reactions, the Chemical Master Equation (CME) can be used to capture the evolution of $X(t)$ (de Jong, 2002). However, its size grows exponentially as the number of species increase and does not have analytical solutions. In order to efficiently simulate CME, Gillespie (1977) developed a stochastic simulation algorithm. Instead of solving for the individual state transition probabilities, the Gillespie's algorithm generates trajectories of $X(t)$. The statistical properties of the ensemble of the trajectories generated by the algorithm can yield - in principle- accurate information about the global stochastic dynamics as predicted by the CME. Since the Gillespie's algorithm is computationally expensive in terms of time, many improvements have been proposed such as the τ-leaping approximation (Wilkinson, 2006).

Note that if the value of $X_i(t)$ represent a discret concentration level, $X(t)$ can be viewed as a Continuous Time Markov Chain (CTMC) (Ross, 2002). Hence the idea of Gillespie's algorithm can also been adapted by many stochastic modeling formalisms such as PEPA (Hillston, 1996), PRISM (Kwiatkowska et al., 2002), and κ (Danos et al., 2007), which are modeling languages describing the system's dynamics in terms of a CTMC.

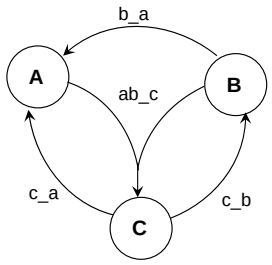
PEPA

PEPA (Hillston, 1996) is a stochastic process algebra originally designed to modeling computer and communication systems. Recently, it has also been applied to modeling biological pathways (Calder et al., 2006b,c; Ciocchetta et al., 2009). The PEPA language have five combinators, *prefix*, *choice*, *cooperation*, *hiding* and *constant*.

- **Prefix** $(\alpha, r).P$ implies that after the component has performed activity α at rate r, it behaves as componet P.
- **Choice** $P_1 + P_2$ sets up a competition between two possible alternatives.
- **Cooperation** $P_1 \underset{L}{\bowtie} P_2$ describes the synchronization of P_1 and P_2 over the activities in the cooperation set L.
- **Hiding** P/L is a component behaves like P except that any activities of types within L are hidden.
- **Constant** $A \stackrel{def}{=} P$ is a component whose meaning is given by a defining equation.

Example Figure 2.8 shows a PEPA model of a small network presented in Calder et al. (2006a). Species A, B, and C are associated with distinct PEPA components. The concentrations of species are discretized into high (H) and low (L) levels. A stochastic rate is associated with each event in this process algebra.

A PEPA model can be mapped to a CTMC and can be simulated and analyzed using stochastic simulation tools such as Dizzy (Ramsey et al., 2005). If we use numbers of molecules instead of discrete concentration levels, a PEPA model can be mapped to a CME that can be simulated using Gillespie's algorithm. Interestingly, Geisweiller et al. (2008) showed that an ODE model can also be derived from a PEPA representation. Recently, an extension of PEPA called Bio-PEPA has been proposed in order to handle more features of biological systems. Bio-PEPA is promising to support different kinds



$$A_L \stackrel{def}{=} (b\_a, \beta).A_H + (c\_a, \gamma).A_H$$
$$A_H \stackrel{def}{=} (ab\_c, \alpha).A_L$$
$$B_L \stackrel{def}{=} (c\_b, \delta).B_H$$
$$B_H \stackrel{def}{=} (ab\_c, \alpha).B_L + (b\_a, \beta).B_L$$
$$C_L \stackrel{def}{=} (ab\_c, \alpha).C_H$$
$$C_H \stackrel{def}{=} (c\_a, \gamma).C_L + (c\_b, \delta).C_L$$
$$(A_H \underset{\{ab\_c, b\_a\}}{\bowtie} B_H) \underset{\{ab\_c, c\_a, c\_b\}}{\bowtie} C_L$$

Figure 2.8: A PEPA example of a small biopathway (Calder et al., 2006a).

of analysis, including stochastic simulation, ODE-based analysis, and PRISM-based model checking.

PRISM

Probabilistic modeling checking is a formal verification technique for analyzing the properties of stochastic systems (Kwiatkowska et al., 2007). PRISM (Kwiatkowska et al., 2002) is the state of the art tool for carrying out probabilistic model checking on CTMC models and has been applied to systems from various domains. Recently, it has been used to analyze biological pathways (Kwiatkowska and Heath, 2009) such as the ERK (Calder et al., 2005), and FGF signaling pathways (Kwiatkowska et al., 2006; Heath et al., 2008).

The PRISM modeling language describes stochastic systems using *variables* and *modules*. In the context of biopathway modeling, the values of variables are nonnegative integer representing the discrete concentration levels of species. A module contains a number of variables and specifies then updating rules for them. Each rule describes how the values of variables involved in a biochemical reaction are updated under particular conditions. Each update is also assigned a rate describing the probability of occurring.

Example Figure 2.9 shows an example of the PRISM model of the reversible binding process presented in equation 2.1.

```
ctmc

const double k1 = 0.1;
const double k2 = 0.01;

module A
     A : [0..5] init 5;
     [bind] (A>0) -> A* k1 : (A' = A - 1);
endmodlue

module A
     A : [0..5] init 5;
     [bind] (B>0) -> B* k1 : (B' = B - 1);
endmodlue

module AB
     AB : [0..5] init 0;
     [bind] (AB < 5) -> k2 : (AB' = AB + 1);
endmodlue

module RATES
     [bind] true -> k1 : true;
     [bind] true -> k2 : true;
endmodlue
```

Figure 2.9: A PRISM example of the binding process $A + B \rightleftharpoons AB$.

The main feature of PRISM-based modeling is that many interesting and complex properties of the system can be verified via probabilistic model checking. PRISM allows properties to be specified using various temporal logics such as Linear Temporal Logic (LTL) (Pnueli, 1977), Probabilistic Continuous Temporal Logic (PCTL) (Hansson and Jonsson, 1994) and Continuous Stochastic Logic (CSL) (Aziz et al., 2000). For instance, a property can be written as the following logical formula:

$$(A < 2) \Rightarrow \mathbf{P}_{>0.2}[true\mathbf{U}^{[0,4]}(AB = 3)] \tag{2.10}$$

This property can be read as "if protein A's concentration level is lower than 2, then the probability of the complex AB's concentration level being 3 within the next 4 seconds is greater than 0.2".

A common limitation of the current stochastic models is scalability. As stochastic simulations are computational intensive, the computations may become intractable when analyzing large pathways. For instance, it has been reported by Calder et al. (2005) that modeling checking an PRISM model of the ERK pathway, which consists of only 11 species, with additional inhibition reactions, required the computational power of a grid of over 90 computers.

2.4 Model Calibration

As discussed in previous section, many of the quantitative formalisms will induce a large number of parameters. Usually, only a few of them are available in literature or can be directly measured experimentally. Most of their values will be unknown. Thus, one often has to estimate the values of unknown parameters from experimental data. In this section, we focus on model calibration in the context of deterministic formalisms such as ODEs and Petri nets, since stochastic models often assume parameters are known and very little has been done for calibrating them.

The goal of model calibration is to estimate unknown parameter so that the model can reproduce the experimental observations. Hence a common approach of parameter estimation is to optimize the agreement between the model prediction and available experimental data. In this manner, parameter estimation can be formulated as an optimization problem with differential algebraic constraints. Typically, the goodness-of-fit of a parameter combination is evaluated by the following objective function, which measures the weighted sum of square error between model prediction and experimental data:

$$f_{obj}(\mathbf{p}) = \sum_{i,j} \omega_i (x_{i,j} - y_{i,j}(\mathbf{p}))^2 \tag{2.11}$$

where $\mathbf{p}$ is the parameter set being tested, $x_{i,j}$ is the experimental observation of

the concentration of species x_i at time point j, $y_{i,j}(\mathbf{p})$ is the corresponding prediction generated using $\mathbf{p}$, and ω_i is the normalization factor for x_i which is usually the inverse of the maximum value of x_i.

In order to find the parameter set $\mathbf{p}_{opt}$ that has the minimum objective value, a common scheme of optimization algorithms is to repeatedly execute two steps: (1) make guesses regarding the values of the parameters; (2) evaluate the goodness-of-fit of the guesses. For step (1), guesses may be generated randomly in the first round but later guesses are usually made based on the results of previous rounds. For step (2), to get the value of $y_{i,j}$ in equation 2.11, one will have to simulate the ODE system upto the maximum time point of the experimental observations. Obtaining the optimal solution often requires repeated executions of these two steps. Thus, the parameter estimation process will often be computationally intensive.

To improve the performance of parameter estimation, a critical issue to be addressed is how to make "clever" guesses based on guesses that have been evaluated. In other words, how to traverse the solution space so that the optimal solution can be found as fast as possible? The traversing process is also known as *searching*, which is the major distinguishing feature of the parameter estimation algorithms. For instance, to determine the next point in the solution space to search, the Steepest Descent (Fogel et al., 1992) method will follow the direction of steepest descent on the hypersurface of the objective function. The Levenberg-Marquardt (Levenberg, 2; Marquardt, 1963) method combines this heuristic with the Newton methods. The Hooke and Jeeves (HJ) method (Hooke and Jeeves, 1961; Swann, 1972) will remember the descent direction of previous searches and suggest a new direction to search. These methods are classified as the *local methods*. In practice, they converge quite fast. However, they suffer the local minima problem (Moles et al., 2003) and often return suboptimal solutions with bad quality.

On other hand, global methods in principle guarantee optimal solutions. Many

global methods have been proposed based on a variety of heuristics inspired by nature. For example, algorithms such as Genetic Algorithm (GA) (Back et al., 1997; Mitchell, 1995) and Evolutionary Strategy (ES) try to mimic evolution which is driven by reproduction and selection. The idea of ES is illustrated in Algorithm 1. Particle Swarm Optimization (PSO) method develped by Kennedy and Eberhart (1995) is inspired by a flock of birds or a school of fish searching for food. Benchmarking tests of the performance of global methods on biological pathway models have been done by Moles et al. (2003) and Fomekong-Nanfack et al. (2007). They separately showed that a variation of ES called Stochastic Ranking Evolutionary Strategy (SRES) (Runarsson and Yao, 2000) outperform other commonly used global methods. Some recent works attempted to improve SRES by either transforming the search space (Kleinstein et al., 2006) or incorporating more heuristics such as Fisher information matrix analysis (Rodriguez-Fernandez et al., 2006a,b). Although the resulting algorithms outperform others in general cases, they might still fail to produce good results within acceptable time when dealing with large signaling networks. A pragmatic strategy one may consider is to optimize the standard parameter estimation algorithms using the network properties of the particular biological pathways being studied. Such example can be found in Birtwistle et al. (3) and Bentele et al. (2004).

As one of the difficulties of parameter estimation is due to the high dimensionality of search space, Koh et al. (2005) proposed a decompositional approach that can break down a large pathway model into smaller components by exploiting its structure. As a result, estimating parameters within each component separately is allowed and the computational cost is largely reduced. In this approach, components that share common parts may have conflicting parameter estimations, as they are computed independently. Thus, in a subsequent work (Koh et al., 2007), global consistency is achieved by applying belief propagation techniques. Notice that not all the networks can be decomposed into small components and decompositional approaches rely on other search

begin
  Initialize parent population $\mathbf{P}_\mu = \{\mathbf{p}_1, \ldots, \mathbf{p}_\mu\}$
  repeat
    for $i \leftarrow$ *1 to* λ **do**
      $\mathcal{S} \leftarrow \mathbf{P}_\mu$
      Randomly select parents $\mathbf{p}_{c1}, \mathbf{p}_{c2} \in \mathbf{P}_\mu$
      $\mathbf{p}_{new} \leftarrow$ Recombine$(\mathbf{p}_{c1}, \mathbf{p}_{c2})$
      $\mathbf{p}_{new} \leftarrow$ Mutate$(\mathbf{p}_{new})$
      $\mathcal{S} \leftarrow \mathcal{S} \cup \{\mathbf{p}_{new}\}$
      Sort$(\mathcal{S})$
      $\mathbf{P}_\mu \leftarrow$ Select first μ from $\mathcal{S}$
    end
  until Stopping Criteria ;
end

Algorithm 1: $(\mu + \lambda)$-ES

methods to deal with single component. Further, the high cost of simulations remains a major barrier. Another difficulty of parameter estimation is due to inherent uncertainty of data. To deal with noisy data, probabilistic approaches that aim to estimate the posterior distributions for parameters via Bayesian inference has been proposed (Yoshida et al., 2008; Girolami, 2008; Koh et al., 2010a). Furthermore, experimental data is often generated incrementally. When new data arrives, we may have to repeat the whole process of parameter estimation which is very time consuming. A recent work by Koh et al. (2010b) attempts to address this issue by representing the pathway parameter estimates using probabilistic graphical models. As a result, parameter estimation can be performed incrementally by integrating new experimental data into an existing model.

2.5 Model Analysis

2.5.1 Sensitivity Analysis

In general, sensitivity analysis is the study of how the variation in the *input* of a computational model affects, qualitatively or quantitatively, the *output* of the model(Saltelli,

2008). Here the input can be defined as the initial state or parameters of the model and the output can be defined to be the dynamical behavior of a network component of interest. Besides enriching our understanding, sensitivity analysis is also a powerful technique for a range of purposes that have been summarized by van Riel (2006) as follows:

- Drug target selection (Cascante et al., 2002; Rullmann et al., 2005).
- Biomarker selection (de Pillis et al., 2005).
- Experiment design (Rodriguez-Fernandez et al., 2006c; Cho et al., 2003; Gadkar et al., 2005b).
- Model reduction (Bentele et al., 2004).
- Robustness analysis (von Dassow et al., 2000; El-Samad et al., 2005).

Local sensitivity analysis

Local sensitivity analysis is a particular form of sensitivity analysis similar to *metabolic control analysis* (Salter et al., 1994). It has been widely applied on models of biological pathways ranging from metabolic pathways (van Stiphout et al., 2006) to signaling pathways (Schoeberl et al., 2002). Specifically, the sensitivity coefficient s_{ij} is defined as the normalized first order derivatives of the model output o_i with respect to the model parameter p_j:

$$s_{ij} := \frac{\partial o_i}{\partial p_j} \cdot \frac{p_j}{o_i} \equiv \frac{\partial \ln(o_i)}{\partial \ln(p_j)} \tag{2.12}$$

Here one often specify p_j to be a rate constant or the initial concentration of a species and o_i as a quantity that assets a characteristic of the system response. For instance, we can define o_i to be the transient concentration of a particular species (usually the endpoint of signal transduction) at a specific time point t (Birtwistle et al., 3). In this case, the sensitivity s_{ij} will become time dependent and can be denoted as $s_{ij}(t)$.

One may then plot $s_{ij}(t)$ and further investigate how the sensitivities evolve over time (Gunawan and Doyle, 2006). Furthermore, depending on the dynamical properties of the system being studied, many other characteristics of the output response have been used, such as: the amplitude and time of the response peak, the duration of the response (Schilling et al., 2009), the integration of the response curve (Swameye et al., 2003), the amplitude, period and phase of oscillation (Schoeberl et al., 2002; van Stiphout et al., 2006; Gunawan and Doyle, 2006), the steady-state levels (Feng and Rabitz, 2004), the deviation from the observations (Cho et al., 2003; Zi et al., 2005; Zhang and Rundell, 2006), etc.

Given a parameter p_j, the corresponding o_i can be predicted by simulating the model. Thus, centered difference approximation techniques (Gunawan et al., 2005) can be employed to compute sensitivity coefficients s_{ij} as follows:

$$s_{ij} = \frac{\partial o_i}{\partial p_j} \cdot \frac{p_j}{o_i} \approx \frac{o_i(p_j + \Delta p_j) - o_i(p_j - \Delta p_j)}{2\Delta p_j} \cdot \frac{p_j}{o_i} \tag{2.13}$$

Local sensitivity analysis assesses the effects of perturbations within a small local region around a specific point in parameter space. In other words, the computed local sensitivities rely on the actual values of model parameters. However, in practice, the values of many parameters have to be estimated form noisy and limited value. It is possible for local sensitivity analysis to draw different conclusions about the importance of the same parameter based on different sets of estimated values. Furthermore, even if all the parameters can be measured experimentally, changes in cellular environments may induce extensive variations of model parameters that might lead to different local sensitivities. Therefore, it is a good to do sensitivity analysis in a more *global* manner by exploring the effects of perturbations within a large region of parameter space.

Global sensitivity analysis

To overcome the limitations of traditional local sensitivity analysis methods, various global methods have been recently applied on biological pathway models (Cho et al., 2003; Zi et al., 2005; Bentele et al., 2004; Zhang and Rundell, 2006; Lüdtke et al., 2008; Rodriguez-Fernandez and Banga, 2008). These methods assess the overall effects of parameters on the model output by simultaneously perturbing all the parameters within a parameter space. A common Monte Carlo scheme adopted by many of them can be described as follows: (1) draw a representative number of samples from the parameter space (2) simulate the system for each sampled combination of parameters (3) derive the global sensitivities of parameters by a statistical or information theoretic analysis of the simulation results.

In step (3), the global sensitivities are measured in different ways depending on the method used. For instance, the partial rank correlation coefficient (PRCC) analysis calculates the global sensitivities from the Pearson correlation coefficients between model output and input parameters (Draper and Smith, 1981). The global sensitivities calculated by Bentele et al. (2004) is a weighted average of the local sensitivities of sampled values of parameters, where the weights are determined by a Boltzmann distribution function of the error between model simulation and experimental data. Sobol's method estimates the partial variances of the model output for input parameters and defines the global sensitivities as the ratio of the related partial variances to the overall variance of the model output (Sobol, 2001). In Multi-parametric sensitivity analysis (MPSA) (Cho et al., 2003), the sampled parameter sets are classified into two classes based on the objective value of each sample, which measures the error between experimental data and prediction generated by selected parameters. The global sensitivities are then evaluated as the Kolmogorov-Smirnov statistic Kirjavainen et al. (2008) of cumulative frequency curves of the parameter values associated with the two classes. There are also attempts of deriving global sensitivities via information

theoretic analysis. For example, Lüdtke et al. (2008) treated the pathway system as a 'communication channel' and quantified the associations between input parameters and model output by decomposing their mutual information. More methods for global sensitivity analysis have been reviewed in the book by Saltelli (2008).

Biological pathway models often contain many parameters, which lead to a high dimensional parameter space. Hence step (1) of the above scheme will require a large number of samples to explore the parameters space. Consequently, carrying out global sensitivity analysis becomes computationally extremely expensive. To get around of this, efficient sampling methods have been proposed. For instance, Latin hypercube sampling (LHS) is a sampling method requiring fewer samples while guaranteeing that individual parameter ranges are evenly covered. It has been adopted to improve MPSA (Zi et al., 2005) and PRCC (Zhang and Rundell, 2006) analysis. Instead of random sampling, heuristic sampling from optimization algorithms has been used for computing global sensitivities with certain special definition (Sahle et al., 2008). Furthermore, Zhang and Rundell (2006) proposed to reuse the computational effort put during parameter estimation to improve the performance of global sensitivity analysis.

2.5.2 Perturbation Optimization

With a comprehensive understanding of cellular mechanisms, the modern technologies enable us to have many controls over the cellular functioning and phenotype. Such controls are often accomplished by means of genetic modifications or drug treatment, which perturb properties of components or interactions in a biological network. As a result, desired cellular properties or dynamical behaviors might be achieved to facilitate the development of many applications, ranging from therapeutic strategies for diseases (Khosla and Keasling, 2003) to industrial applications of metabolic engineering (Raab et al., 2005) and synthetic biology (Andrianantoandro et al., 2006; Heinemann and Panke, 2006) such as production of various biochemical substance including proteins

(Vives et al., 2003), amino acids (Park and Lee, 2010), biofuels (Keasling and Chou, 2008), etc. For example, L-threonine is an amino acid that has been widely used in industries of cosmetics and pharmacy (Lee et al., 2007). L-threonine has been produced from bacteria such as *Escherichia coli* through biosynthetic pathways. The productivity can be improved by genetically mutating genes encoding pathway components. The goal here is to maximize the production of L-threonine, and in the meantime, to minimize the formation of undesirable byproducts. To achieve this goal, one has to answer the question "which genes shall we mutate?". Similar questions will be raised by all applications presented above. However, it is very difficult to answer due to the inherent complexity of biological networks. As the number of candidate perturbation strategies will be exponential, it is impossible to test the effect of strategies one by one to pinpoint the best strategy. Instead, one will have to resort to computational models, on which *in silico* perturbations effects can be cheaply simulated and examined, to figure out the optimal solution. We term this kind of model analysis as *perturbation optimization*.

Mathematically, perturbation optimization is a combinatorial optimization problem:

$$\begin{aligned} &\text{maximize: } f(x) \\ &\text{subject to: } \mathbf{c}(x), \end{aligned}$$

where the decision variable x denotes a perturbation, the objective function f to maximize quantifies simulation results of the model with the corresponding perturbation, and $\mathbf{c}$ is a set of constraints specifying the requirements that must be met to ensure cells survive and have proper functioning. A perturbation can be the mutation of a set of genes, which will result in the changes of initial conditions or kinetic parameters in the model. For instance, in an ODE model of metabolic pathways in *E. coli* Lee et al. (2007), deleting the *lysA* gene will induce the initial concentration of diaminopimelate decarboxylase to be zero. Furthermore, a point mutation replacing the 290th C with

T of the *ilvA* gene will decrease the activity of threonine dehydratase and result in the changes on related kinetic parameters. It is worth noting that the *E. coli* model constructed by Lee et al. (2007) consists of 979 reactions and 814 species. Due to the combinatorial nature of the problem, such large models will induce solution spaces containing a huge number of candidate perturbations. Hence the optimization procedure is often very computationally intensive as it requires a large amount of model simulations. To combat the combinatorial explosion of solution space, many optimization methods have been used in recent years. A review of several standard methods employed can be found in Banga (2008), including Linear programming (LP) (Papoutsakis, 1984), Bilevel optimization (BLO) (Burgard et al., 2003; Chang and Sahinidis, 2005; Gadkar et al., 2005a), Mixed Integer nonlinear programming (MINLP) (Vital-Lopez et al., 2006), and Dynamic optimization (DO) (Lebiedz, 2005).

Chapter 3

Preliminaries

In this chapter, we develop the notions leading to the fact that the flows (vector fields) that arise as the solution to our systems of ODEs will be measurable functions. This will secure the mathematical basis for our approximation. More information can be found in (Hirsch et al., 2004; Ammann, 1990; Durrett, 2004; Feldman, 2008).

3.1 Continuity, Probability and Measure Theory

Let $\mathbb{N}$ denote the set of non-negative integers. Assume that X and Y are metric spaces (Bryant, 1985). A function $f : X \to Y$ is said to be of class C^k, where $k \in \mathbb{N}$, if the derivatives f', f'', ..., $f^{(k)}$ exist and are continuous. Thus, the class C^0 consists of all continuous functions and the class C^1 consists of all continuously differentiable functions.

A **σ-algebra** over a set X is a nonempty collection of subsets of X that is closed under complementation and countable unions. The **Borel σ-algebra** on a topological space X, denoted as $\mathcal{B}_X$, is the minimal σ-algebra containing all the open sets of X.

A **probability space** is a triple $(\Omega, \mathcal{F}, \mathbf{P})$ consisting of a set Ω, a σ-algebra $\mathcal{F}$ over Ω, and a function $\mathbf{P} : \mathcal{F} \to [0, 1]$ such that:

(i) $\mathbf{P}(\Omega) = 1$;

(ii) if $\{A_w\}_{w \in W}$ is a countable family of pairwise disjoint sets in $\mathcal{F}$, then $\mathbf{P}(\cup_w A_w) = \sum_w \mathbf{P}(A_w)$.

Let X and Y be nonempty sets and $\mathcal{M}$ and $\mathcal{N}$ be σ-algebras of subsets of X and Y respectively. A function $f : X \to Y$ is said to be $(\mathcal{M}, \mathcal{N})$-**measurable** if

$$E \in \mathcal{N} \Rightarrow f^{-1}(E) \in \mathcal{M}. \tag{3.1}$$

The following fact is crucial for our purposes.

Proposition 3.1. (Feldman, 2008) *If X and Y are metric spaces and $f : X \to Y$ is continuous, then f is $(\mathcal{B}_X, \mathcal{B}_Y)$-measurable.*

3.2 ODEs and Flows

Through the rest of this chapter, we assume a set of ODEs

$$\dot{x_i}(t) = f_i(\mathbf{x}(t), \mathbf{p}) \tag{3.2}$$

involving the variables $\{x_1, x_2, \ldots, x_n\}$. Each variable $x_i(t)$ is a real-valued function of t with the domain of t being the set of reals. $\{p_1, p_2, \ldots, p_m\}$ is the set of real-valued parameters. We will require the ODEs to be *autonomous* in the sense t does not appear explicitly in any f_i. In our setting, we will often be interested in studying the dynamics for different combinations of values for the parameters. Hence it will be convenient to treat them also as variables. However they will be time-invariant; once their values are fixed at $t = 0$, these values will not change through the passage of time. Consequently, we will implicitly assume m additional differential equations of the form $\dot{p_j}(t) = 0$ with j ranging over $\{1, 2, \ldots, m\}$. We will often let $\mathbf{v}$ range over $\mathbb{R}^n_+$, the values space of the

variables and $\mathbf{k}$ range over $\mathbb{R}_+^m$, the values space of the parameters and $\mathbf{z}$ range over $\mathbb{R}_+^{n+m}$, the combined values space. In vector form, our system of autonomous ODEs may be represented as:

$$\mathbf{Z}' = F(\mathbf{Z}). \tag{3.3}$$

We shall assume that the ODEs will be modeling mass action or Michaelis-Menten kinetics (Klipp et al., 2005). However, our method will be applicable for many other types of reaction kinetics too.

Based on the preceding remarks, we can assume $f_i : \mathbb{R}_+^{n+m} \to \mathbb{R}_+$ to be of the form:

$$\sum_{j=1}^{r_i} c_j n_{ij} g_j, \tag{3.4}$$

where r_i is the number of reactions associated with species x_i and $c_j = -1$ ($c_j = +1$) if x_i is a reactant (product) of the jth reaction. Further, the quantities $n_{ij} \in \mathbb{Z}$ denote the stoichiometric coefficients and g_j are rational functions of the form $g_j = p_\alpha x_a x_b$ (mass action) or $g_j = p_\alpha x_a x_b/(p_\beta + x_a)$ (Michaelis-Menten) with $a, b \in \{1, 2, \ldots, n\}$ and $\alpha, \beta \in \{1, 2, \ldots, m\}$, describing the kinetic rates of the corresponding reactions. Consequently, we shall assume that g_i are differentiable and g_i' are continuous on $\mathbb{R}_+$. As a result, g_i are C^1 (continuously differentiable) functions. This leads us to $f_i \in C^1$ for each i and hence $F : \mathbb{R}_+^{n+m} \to \mathbb{R}_+^{n+m}$ can also be assumed to be a C^1 function. Furthermore, the variables representing the concentration level of a species within a single cell as well as the parameters capturing the reaction rates will take values from a bounded interval. Hence the domain of F can be restricted to a bounded region $\mathcal{D}$ of $\mathbb{R}_+^{n+m}$.

Given $\mathbf{z}_0 = (\mathbf{v}_0, \mathbf{k})$ where $\mathbf{v}_0$ specifies the initial values of the variables and $\mathbf{k}$ specifies the parameters values, the system of ODEs will have a unique solution since $F \in C^1$ (Hirsch et al., 2004). We shall denote this solution by $\mathbf{Z}(t)$ with $\mathbf{Z}(0) = \mathbf{z}_0$ and $\mathbf{Z}'(t) = F(\mathbf{Z}(t))$. We are guaranteed that $\mathbf{Z}(t)$ will be a C^0-function (Hirsch et al.,

2004).

It will be convenient to define the flow $\Phi : \mathbb{R}_+ \times \mathcal{D} \to \mathcal{D}$ of $\mathbf{Z}' = F(\mathbf{Z})$ for arbitrary initial vectors $\mathbf{z}$. It will be a C^0-function given by: $\Phi(t, \mathbf{z}) = \mathbf{Z}(t)$ with $\Phi(0, \mathbf{z}) = \mathbf{Z}(0) = \mathbf{z}$ and $\partial(\Phi(t, \mathbf{z}))/\partial t = F(\Phi(t, \mathbf{z}))$ for all t (Hirsch et al., 2004). Further, $\Phi(t, \cdot)$ will be bijective.

Since the flow Φ is C^0, i.e. continuous and $\mathcal{D} \subseteq \mathbb{R}^{n+m}$ is a metric space we are assured that $\Phi(t, \cdot)$ is $(\mathcal{B}_\mathcal{D}, \mathcal{B}_\mathcal{D})$-measurable by Proposition 3.1. In what follows, we use Φ_t to denote $\Phi(t, \cdot)$ and summarize the above observations via:

Proposition 3.2. *Suppose $\boldsymbol{Z}' = F(\boldsymbol{Z})$ is an autonomous system of ODEs with F in C^1 and with the domain of F being a bounded region $\mathcal{D}$ of $\mathbb{R}_+^{n+m}$. Then there exists a unique flow $\Phi : \mathbb{R}_+ \times \mathcal{D} \to \mathcal{D}$ for arbitrary initial vectors $\boldsymbol{z}$ satisfying: $\Phi(t, \boldsymbol{z}) = \boldsymbol{Z}(t)$ with $\Phi(0, \boldsymbol{z}) = \boldsymbol{Z}(0) = \boldsymbol{z}$ and $\partial(\Phi(t, \boldsymbol{z}))/\partial t = F(\Phi(t, \boldsymbol{z}))$ for all t. Further, $\Phi(t, \cdot)$ will be in C^0 and hence $\mathcal{B}_\mathcal{D}$-measurable. As a result, for all $t \in \mathbb{R}$:*

$$B \in \mathcal{B}_\mathcal{D} \Rightarrow \Phi_t^{-1}(B) = \{\boldsymbol{z} \in \mathcal{D} \mid \Phi(t, \boldsymbol{z}) \in B\} \in \mathcal{B}_\mathcal{D}. \tag{3.5}$$

3.3 Markov Chains

A **Markov Chain** (Norris, 1997) is a pair $(S, \{p_{ij}\})$ where $S = \{s_1, s_2, \ldots, s_{\hat{n}}\}$ is set of states and $p_{ij} \in [0, 1]$ are the transition probabilities with $\Sigma_{j=1}^{\hat{n}} p_{ij} = 1$ for every i. Thus if the system is in state s_i at t then it will be in state s_j at $t + 1$ with probability p_{ij}. Given an initial probability distribution Ψ^0 over S at $t = 0$, viewed as an $1 \times \hat{n}$-row vector, the probability distribution Ψ^k over S at $t = k$ will be given by $(\Psi^0)T^k$ where T is the $\hat{n} \times \hat{n}$ transition probability matrix with $T_{ij} = p_{ij}$.

3.4 Bayesian Networks

A **Bayesian network** (Russell and Norvig, 2003) is a finite acyclic directed graph $BN = (V, E)$ which has a finite-valued random variable X_v and a conditional probability table CPT_v associated with each node v. The entries in CPT_v will be of the form $Pr(X_v = x|X_{v_1} = x_1, X_{v_2} = x_2, \ldots, X_{v_j} = x_j)$ where $\{v_1, v_2, \ldots, v_j\}$ is the set of *parents* of v given by $Pa(v) = \{u \mid (u, v) \in E\}$. BN represents -often compactly- the joint probability distribution over the random variables $\{X_v\}_{v \in V}$ given by: $Pr(X_{v_1} = x_1, X_{v_2} = x_2, \ldots, X_{v_{\check{n}}} = x_{\check{n}}) = \prod_{i=1}^{\check{n}} Pr(X_{v_i} = x_i|X_{v_{i1}} = x_{i1}, X_{v_{i2}} = x_{i2}, \ldots, X_{v_{ij}} = x_{ij})$ with $Pa(v_i) = \{v_{i1}, v_{i2}, \ldots, v_{ij}\}$.

3.5 Dynamic Bayesian Networks

Dynamic Bayesian networks (DBNs) are Bayesian networks that model temporal evolution of systems whose (local states) are modeled as random variables (Murphy, 2002). There are many variants of dynamic Bayesian networks. We will be dealing with a restricted class of time-variant two-slice dynamic Bayesian networks. They will be of the form $(B_0, \{B^d_{\rightarrow}\}_{d=1}^{\hat{d}}, Pa)$, where B_0 defines the initial probability distributions $\{Pr(\mathbf{X}_i^0)\}$ of the random variables $\{X_i\}_{i=1}^{l}$. And $\{B^d_{\rightarrow}\}$ are two-slice temporal Bayesian networks for the time points $\{t^1, \ldots, t^{\hat{d}}\}$. The nodes of the Bayesian network $B^d_{\rightarrow}$ denoted V^d is given by $V^d = \{X_i^{d-1} \mid 1 \leq i \leq l\} \cup \{X_i^d \mid 1 \leq i \leq l\}$ (here we are identifying the nodes with the random variables associated with them). The edge relation E^d will be the subset of $\{X_i^{d-1} \mid 1 \leq i \leq l\} \times \{X_i^d \mid 1 \leq i \leq l\}$ satisfying $(X_j^{d-1}, X_i^d) \in E^d$ iff $X_j \in Pa(X_i)$. As might be expected, $Pa : \mathbf{X} \rightarrow 2^{\mathbf{X}}$ with $\mathbf{X} = \{X_i \mid 1 \leq i \leq l\}$. Each node X_i^d will also a conditional probability table CPT_i^d associated with it with entries of the form $Pr(X_i^d = x \mid X_{i1}^{d-1} = x_{i1}, \ldots X_{ij}^{d-1} = x_{ij})$, where $Pa(X_i) = \{X_{i1}, \ldots X_{ij}\}$.

Thus the way the nodes of the $(d+1)$th layer are connected to the nodes of the dth

layer will remain invariant. However CPT_i^{d+l} will be, in general, different from CPT_i^d. An example of such a dynamic Bayesian network is shown in Figure 3.1. To avoid clutter we have not shown the CPTs associated with each node. In our setting these dynamic Bayesian networks will represent an associated Markov chain in a factored form.

$Pa(X_1) = \{X_1, X_3\}$

$Pa(X_2) = \{X_2, X_1, X_4\}$

$Pa(X_3) = \{X_3\}$

$Pa(X_4) = \{X_4\}$

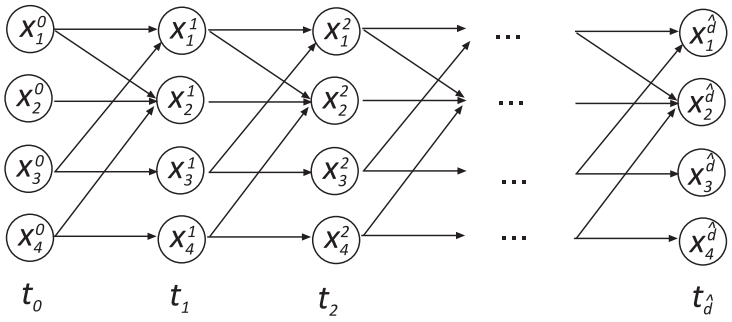


Figure 3.1: A DBN example.

Chapter 4

The Dynamic Bayesian Network Approximation

Here we present our technique for approximating the pathway dynamics defined by a systems of ODEs as a dynamic Bayesian network (DBN).

At the first step we show how the discretization of the value space and time domain leads to the derivation of a Markov chain from the ODEs dynamics. We then show how this Markov chain can be further approximated as a dynamic Bayesian network by introducing independence assumptions obtained from the network structure.

4.1 Overview

Conceptually, our approximation technique consists of two major steps. First we discretize the value spaces of the variables and parameters into a finite sets of intervals. We also discretize the time domain of interest into a finite number of time points. In addition, we assume a prior distribution of initial values over the intervals. As a result, the flow defined by our system of ODEs will induce a Markov chain $\mathcal{MC}_{ideal}$.

In the second step we further approximate $\mathcal{MC}_{ideal}$ as a dynamic Bayesian network

(DBN). This second step is motivated by a number of considerations. To start with, $\mathcal{MC}_{ideal}$ can not be computed explicitly since the ODEs systems of interest will not admit closed form solutions. Secondly, if we approximate $\mathcal{MC}_{ideal}$ directly as a Markov chain, say $\mathcal{MC}_{approx}$, then the resulting Markov chains size will be exponential in the number of variables involved in the ODEs system. To get around this we introduce independence assumptions based on the way the variables are coupled to each other in the biochemical reactions network. We then approximate $\mathcal{MC}_{ideal}$ as a dynamic Bayesian network, which in this context, may be viewed as a factored Markov chain.

We compute the conditional probability tables of the DBN by sampling the initial states according to the prior sufficiently many times and generating a trajectory for each of the sampled initial states. Then by a simple process of counting tied to the discretized value space and time domain, we obtain the dynamic Bayesian network.

This two step procedure is however just a conceptual framework. We shall construct the DBN approximation *directly* from the given system of ODEs instead of passing through a Markov chain. We now proceed with a more technical description the steps involved in constructing the DBN approximation. In doing so, we shall assume that we are given the system of ODEs $\dot{x}_i(t) = f_i(\mathbf{x}(t), \mathbf{p})$ with n variables and m rate parameters specified in the previous chapter with the associated notations and assumptions.

4.2 The Markov Chain $\mathcal{MC}_{ideal}$

Biological pathway models are usually validated by experimental data available only for a few time points with the concentrations measured at the final time point typically signifying the steady state value. Hence we assume the dynamics is of interest only for discrete time points and that too only up to a maximal time point. We denote these time points as $\{t_0, t_1, \ldots, t_{max}\}$. It is *not* necessary to uniformly discretize the time domain. However, to simplify the notations of the following sections, we fix a time step $\Delta t > 0$ and the time points of interest is assumed to be the set $\{d \cdot \Delta t\}$ with d ranging

over $\{0, 1, \ldots, \hat{d}\}$. Thus $\hat{d} \cdot \Delta t$ is the maximal time point of interest.

Next we assume that the values of the variables can be observed with only finite precision and accordingly partition the range of each variable x_i into L^i intervals $[v_i^{min}, v_i^1)$, $[v_i^1, v_i^2)$, ..., $[v_i^{L_i-1}, v_i^{max}]$. We denote this set of intervals as $\mathcal{I}_i$. We also similarly discretize the range of each parameter p_j into a set of intervals denoted as $\mathcal{I}_{n+j}$. The set $\mathcal{I} = \{\mathcal{I}_i\}_{1 \leq i \leq n} \cup \{\mathcal{I}_{n+j}\}_{1 \leq j \leq m}$ is called the **discretization**. Again, we wish to emphasize that the value space can be discretized non-uniformly and our constructions will go through.

As pointed out earlier, the initial values as well as the rate constants (even when they are known) will be given not as point values but as distributions (usually uniform) over the intervals defined by the discretization. We correspondingly assume we are given a prior distribution in the form of a probability density function $\boldsymbol{\Upsilon}^0$ capturing the initial values.

For example, suppose we are given that the initial values are uniformly distributed within a hypercube $\hat{I}_1 \times \hat{I}_2 \times \ldots \times \hat{I}_{n+m}$, where $\hat{I}_i \in \mathcal{I}_i$ for each i. Let $\hat{I}_i = [l_i, u_i)$ and $\hat{w}_i = u_i - l_i$. Then the corresponding prior probability density function $\boldsymbol{\Upsilon}^0$ will be given by:

$$\boldsymbol{\Upsilon}^0(\mathbf{z}) = \begin{cases} \frac{1}{\hat{w}_1 \cdot \hat{w}_2 \cdot \ldots \cdot \hat{w}_{n+m}} & \text{if } \mathbf{z} \in \hat{I}_1 \times \hat{I}_2 \times \ldots \times \hat{I}_{n+m}, \\ 0 & \text{otherwise.} \end{cases} \tag{4.1}$$

The associated probability space we have in mind is $(\mathcal{D}, \mathcal{B}_\mathcal{D}, \mathbf{P}^0)$ where $\mathcal{D}$ is the domain of the ODEs (see Section 3.2), $\mathcal{B}_\mathcal{D}$ is the Borel σ-algebra over $\mathcal{D}$; the minimal σ-algebra containing the open sets of $\mathcal{D}$ under the usual topology. $\mathbf{P}^0$ is the probability distribution induced by $\boldsymbol{\Upsilon}^0$ and is given by:

$$\mathbf{P}^0(B) = \int_B \boldsymbol{\Upsilon}^0(\mathbf{z}) d\mathbf{z}, \text{ for every } B \in \mathcal{B}_\mathcal{D}. \tag{4.2}$$

Further, $TRAJ_{ideal} = \{\Phi_t(\mathbf{z})\}_{t \geq 0}$ with $\mathbf{z}$ ranging over $\hat{I}_1 \times \hat{I}_2 \times \ldots \times \hat{I}_{n+m}$ is the

family of trajectories starting from all the possible points in this hypercube. As before, Φ is the flow induced by the system ODEs.

Φ is measurable by Proposition 3.1. Hence we can define the probability distribution $\mathbf{P}^t$ over $\mathcal{B}_\mathcal{D}$ for every t as:

$$\mathbf{P}^t(B) = \mathbf{P}^0(\Phi_t^{-1}(B)), \text{ for every } B \in \mathcal{B}_\mathcal{D}. \tag{4.3}$$

Let v be a real number in the range of x_i. We define $[v]$ as the interval in which v falls. In other words, $[v] = I$ iff $v \in I$. Similarly, $[k] = J$ if $k \in J$ for a parameter value k of p_j with $J \in \mathcal{I}_{n+j}$.

Lifting this notation to the vector setting, if $\mathbf{z} = (v_1, v_2, \ldots, v_n, k_1, k_2, \ldots, k_m) \in \mathbb{R}_+^{n+m}$, we define $[\mathbf{z}] = ([v_1], [v_2], \ldots, [v_n], [k_1], \ldots, [k_m])$ and refer to it as a **discrete state**.

Definition 4.1. *An $\mathcal{MC}$-**state** is a pair $(\boldsymbol{s}, d)$, where $\boldsymbol{s}$ is a discrete state and $d \in \{0, 1, \ldots, \hat{d}\}$.*

We next define $Pr((\mathbf{s}, d)) = \mathbf{P}^{d\cdot\Delta t}(\{\mathbf{z} \mid \mathbf{z} \in I_1 \times I_2 \times \ldots \times I_{n+m}\})$, where $\mathbf{s} = (I_1, I_2, \ldots, I_{n+m})$. We term the $\mathcal{MC}$-state M to be *feasible* iff $Pr(M) > 0$.

Definition 4.2. *The transition relation denoted as $\rightarrow$, between $\mathcal{MC}$-states is defined via: $M = (\boldsymbol{s}, d) \rightarrow M' = (\boldsymbol{s}', d')$ iff $d' = d+1$ and both M and M' are feasible and there exist $\boldsymbol{z}_0$, $\boldsymbol{z}$, and $\boldsymbol{z}'$ such that $\Phi(d\cdot\Delta t,\, \boldsymbol{z}_0) = \boldsymbol{z}$ and $\Phi((d+1)\cdot\Delta t,\, \boldsymbol{z}_0) = \boldsymbol{z}'$. Furthermore, $[\boldsymbol{z}] = \boldsymbol{s}$ and $[\boldsymbol{z}'] = \boldsymbol{s}'$.*

Let E, F denote, respectively, the event that the system is in the discrete state $\mathbf{s}$ at time $d \cdot \Delta t$ and in the discrete state $\mathbf{s}'$ at time $(d+1) \cdot \Delta t$ for two feasible $\mathcal{MC}$-states $(\mathbf{s}, d \cdot \Delta t)$ and $(\mathbf{s}', (d+1) \cdot \Delta t)$. Let $EF = E \cap F$ denote joint event $\{\mathbf{z}_0 \mid \Phi(d \cdot \Delta t, \mathbf{z}_0) \in \mathbf{s}, \Phi((d+1) \cdot \Delta t, \mathbf{z}_0) \in \mathbf{s}'\}$. Consequently, we define the transition probability $Pr((\mathbf{s}, d) \rightarrow (\mathbf{s}', d')) = Pr(F|E) = Pr(EF)/Pr(E)$. Since $Pr(E) > 0$ this transition probability is well-defined.

Definition 4.3. *Let $\mathcal{M} = \{M_1, M_2, \ldots, M_{\hat{n}}\}$ be the set of $\mathcal{M}$-states. We can now define the Markov chain $\mathcal{MC}_{ideal} = (\mathcal{M}, \{p_{ij}\})$ with transition probabilities $p_{ij} = Pr(M_i \rightarrow M_j)$ as above.*

Example A typical biochemical equation depicting an enzyme catalyzed reaction can be written as follows:

$$S + E \underset{k_2}{\overset{k_1}{\rightleftharpoons}} ES \xrightarrow{k_3} E + P \tag{4.4}$$

As the basic component of signal transduction pathways (Stryer, 1988), it accounts for one step in the transduction of a signaling cascade. In this reaction, the enzyme E binds reversibly to the substrate S, before converting it into the product P and releasing it. The parameters k_1, k_2 and k_3 are the rate constants that govern the speed of these reactions. The corresponding ODE model will be:

$$\begin{aligned}
\frac{dS}{dt} &= -k_1 \cdot S \cdot E + k_2 \cdot ES \\
\frac{dE}{dt} &= -k_1 \cdot S \cdot E + (k_2 + k_3) \cdot ES \\
\frac{dES}{dt} &= k_1 \cdot S \cdot E - (k_2 + k_3) \cdot ES \\
\frac{dP}{dt} &= k_3 \cdot ES
\end{aligned}$$

Assuming that the range of each variable or parameter is: $S \in [0, 15]$, $E \in [0, 10]$, $ES \in [0, 10]$, $P \in [0, 15]$, $k_1 \in [0, 1]$, $k_2 \in [0, 1]$, $k_3 \in [0, 1]$ (for simplicity, we ignore all units in this example), we partition each range into 5 equal-sized intervals and form the discretization $\mathcal{I} = \{\mathcal{I}_S, \mathcal{I}_E, \mathcal{I}_{ES}, \mathcal{I}_P, \mathcal{I}_{k_1}, \mathcal{I}_{k_2}, \mathcal{I}_{k_3}\}$, where $\mathcal{I}_S = \mathcal{I}_P = \{[0, 3), [3, 6), [6, 9), [9, 12), [12, 15]\}$, $\mathcal{I}_E = \mathcal{I}_{ES} = \{[0, 2), [2, 4), [4, 6), [6, 8), [8, 10]\}$ and $\mathcal{I}_{k_1} = \mathcal{I}_{k_2} = \mathcal{I}_{k_3} = \{[0, 0.2), [0.2, 0.4), [0.4, 0.6), [0.6, 0.8), [0.8, 1]\}$. We fix the time step Δt to be 0.1 and fix the number of time points to be 100. We have adopted equal-sized intervals

and fixed time steps only for convenience.

Suppose we are given a prior distribution that the initial values $\mathbf{z}_0 = (S,\, E,\, ES,\, P,\, k_1,\, k_2,\, k_3)$ are uniformly distributed within a hypercube $\mathbf{C} = [12, 15] \times [8, 10] \times [0, 2) \times [0, 3) \times [0.2, 0.4) \times [0.4, 0.6) \times [0.2, 0.4)$. We then have the prior probability density function $\boldsymbol{\Upsilon}^0$ given by:

$$\boldsymbol{\Upsilon}^0(\mathbf{z}) = \begin{cases} \frac{1}{(15-12)(10-8)(2-0)(3-0)(0.4-0.2)(0.6-0.2)(0.4-0.2)} = \frac{125}{36} & \text{if } \mathbf{z} \in \mathbf{C}, \\ 0 & \text{otherwise.} \end{cases}$$

Thus, $M_0 = (\mathbf{s}_0 = ([12, 15], [8, 10], [0, 2), [0, 3), [0.2, 0.4), [0.4, 0.6), [0.2, 0.4)), 0)$ will be the initial $\mathcal{MC}$-state of the induced Markov chain $\mathcal{MC}_{ideal}$. Clearly $Pr(M_0) = 1$ since

$$Pr(M_0) = \mathbf{P}^0(\{\mathbf{z} \mid \mathbf{z} \in \mathbf{C}\}) = \int_{\{\mathbf{z}|\mathbf{z}\in\mathbf{C}\}} \boldsymbol{\Upsilon}^0(\mathbf{z}) d\mathbf{z} = 1. \tag{4.5}$$

The ODEs system will typically not admit a closed form solution. Hence Φ can not be derived explicitly and as a consequence, $\mathcal{MC}_{ideal}$ can not be explicitly computed either. Thus, one can only compute approximations of $\mathcal{MC}_{ideal}$. For instance, one could sample $\mathbf{z}$ (the initial state) many times according to the prior distribution $\mathbf{P}^0$ and for each sampled initial $\mathbf{z}$, determine through numerical integration the $\mathcal{M}$-states $[\Phi(d \cdot \Delta t,\, \mathbf{z})]$, with d ranging over $\{0, 1, \ldots, \hat{d}\}$ as well as the transitions along this trajectory. Then through a simple counting process involving the generated trajectories, a Markov chain can be computed as an approximation of $\mathcal{MC}_{ideal}$. However, the number of states of such approximated Markov chain will be exponential in n. As a result, for many biological pathways, it will be simply too large. Instead, we shall construct a time-variant two-slice DBN to compactly represent and approximate $\mathcal{MC}_{ideal}$.

4.3 The DBN Representation

The key observation is that the structure of the system of ODEs can be exploited to factorize $\mathcal{MC}_{ideal}$ into a time-variant 2-slice DBN. This DBN will have $(n+m)\times(\hat{d}+1)$ nodes. The node v will have associated with it a random variable X_i^d. This random variable will take as values the intervals in $\mathcal{I}_i$; the intervals into which the value space of x_i (in case $i \leq n$) or the parameter p_{i-n} (in case $i > n$) has been discretized. The superscript d will stand for the fact the probability distribution associated with X_i^d describes the probability of the value of the variable x_i (or the parameter p_{i-n}) falling into various intervals in $\mathcal{I}_i$ at time $d \cdot \Delta t$ (since the parameter p_{i-n} is a constant, it can be associated with $\frac{dp_{i-n}}{dt} = 0$). In what follows, for convenience, we will use the same name to denote a node and the random variable associated with it. From the context it should be clear which role is intended. We now proceed with the construction of the DBN $(B_0, \{B_{\rightarrow}^d\}_{d=1}^{\hat{d}}, Pa)$.

We assume that the prior distribution of initial values of the variables and parameters are independent of each other. This is often a reasonable assumption. Even when the assumption is violated it is certainly reasonable to assume that marginal prior probabilities of each variable and parameter can be computed and thus $B^0 = \{Pr(X_i^0)\}_{i=1}^{n+m}$ can be computed. Next, the parent relation Pa is defined as follows. In doing so, it will be convenient to identify the variable x_i with X_i and the parameter p_j with X_{n+j}.

Suppose $z, z' \in \{x_1, x_2, \ldots, x_n, p_1, p_2, \ldots, p_m\}$. Then $z' \in Pa(z)$ iff $z' = z$ or z is a variable and z' appears in the right-hand side of the equation for dz/dt in the system of ODEs.

Thus the structure of the ODEs and more precisely, the structure of the biochemical network induces the underlying graph of the DBN.

V^d, the set of nodes of the Bayesian network $B_{\rightarrow}^d = (V^d, E^d)$ will be: $V^d = \{X_i^{d-1} \mid 1 \leq i \leq n+m\} \cup \{X_i^d \mid 1 \leq i \leq n+m\}$. The edge relation E^d is defined in the obvious way now using the function Pa. To spell it out, it will be the subset of

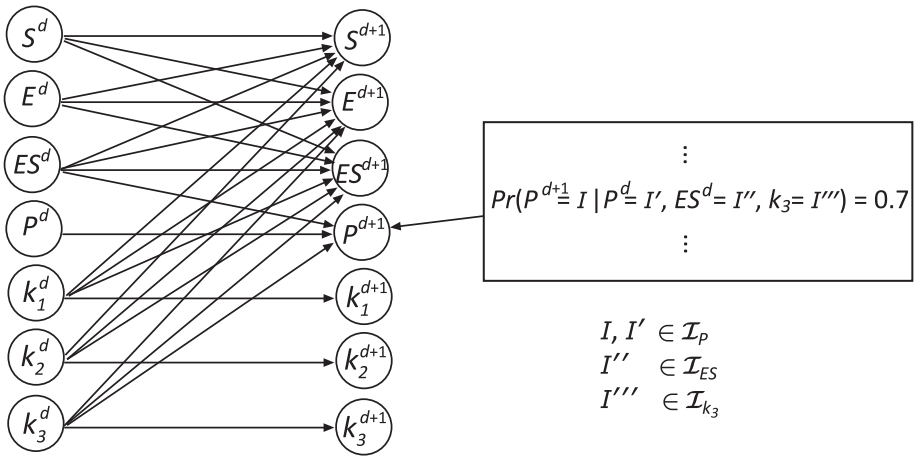


Figure 4.1: A slice of the DBN approximation of the enzyme-kinetic system.

$\{X_i^{d-1} \mid 1 \leq i \leq n+m\} \times \{X_i^d \mid 1 \leq i \leq n+m\}$ satisfying $(X_j^{d-1}, X_i^d) \in E^d$ iff $X_j \in Pa(X_i)$.

Finally, suppose $Pa(x_i) = \{z_1, \ldots, z_l\}$. Then conditional probability table (CPT) associated with the node X_i^d will have entries of the form $Pr(X_i^d = I \mid z_1^{d-1} = I^1, \ldots, z_l^{d-1} = I^l) = h$ with I ranging over $\mathcal{I}_i$ and I^j ranging over $\mathcal{I}_j$ for $1 \leq j \leq l$ and h ranging over $[0, 1]$. This entry captures probability of the value of the variable x_i (assuming $i \leq n$) falling in the interval I at time $d \cdot \Delta t$ given that at time $(d-1) \cdot \Delta t$, the value of the variable (parameter) z_j was in the interval I^j for $1 \leq j \leq l$. It is in this sense the dynamics defined by $\mathcal{MC}_{ideal}$ is captured in a factored form by the DBN.

Example (continued) Figure 4.1 shows two adjacent slices in the DBN approximation of the enzyme-kinetic system. The structure of this DBN is derived from the ODEs presented in section 4.2. For instance, the parent nodes of P^{d+1} are P^d, ES^d and k_3^d since P^d, P^{d+1} refer to the same variable P while ES, k_3 appear in the expression for dP/dt. As mentioned earlier, the parameters are assumed to retain their values during a run and hence we denote k_i^d as simply k_i and there will be no CPTs associated with these nodes. On the other hand, the CPT associated with the node P^{d+1} will have entries of the form $Pr(P^{d+1} = I \mid P^d = I', ES^d = I'', k_3 = I''') = h$,

where $I, I' \in \mathcal{I}_P$, $I'' \in \mathcal{I}_{ES}$, $I''' \in \mathcal{I}_{k_3}$ and $h \in [0,1]$. As illustrated in this example, the connectivity between the nodes in successive slices will remain invariant. However, due to the fact that the CPTs associated with the nodes capture the transition probabilities at different time points, they will be time variant.

$\mathcal{MC}_{ideal}$ will have, in the worst case, $O((\hat{d}+1)K^n)$ states and $O(\hat{d}K^{2n})$ transitions, where K is the maximum of $|\mathcal{I}_i|$ with $1 \leq i \leq n+m$. In contrast, the number of nodes in the DBN representation is $O(\hat{d}(n+m))$ and the conditional probability table associated each node will have at most $O(K^{R+1})$ entries, where R is the maximal number of parents a node can have. Usually, the reactants in pathway models will be sparsely coupled to each other and hence R will be much smaller than n. For instance, in the first case study to be presented in the next chapter, $n = 32$ and $R = 5$.

Since our ODE system will not admit a closed form solution, the conditional probabilities of the DBN can not be directly derived explicitly. To fill up the entries of the CPTs associated with the nodes of DBN, we shall approximately compute conditional probabilities as follows.

We sample $\mathbf{z}$ (the initial state) a sufficiently large number of times, say N, according to the prior distribution $\mathbf{P}^0$ (we say more about N below). Since we assume that the initial values are independent of each other, the values of a variable/parameter can be sampled according to its marginal prior distribution. For instance, in our running example, we can randomly choose a value from $[12, 15]$ for S, a value from $[8, 10]$ for E, a value from $[0, 2)$ for ES, a value from $[0, 3)$ for ES, a value from $[0, 1]$ for k_1, a value from $[0, 1]$ for k_2, a value from $[0, 1]$ for k_3, and then form a vector of initial values.

After picking N sample initial value vectors, we perform numerical integration to generate N trajectories and discretize those trajectories by the predefined intervals and compute the conditional probabilities for each node by simple counting. For example, suppose 132 trajectories hit $(P^0 = [0, 3), ES^0 = [0, 2), k_3 = [0.2, 0.4))$ at time 0 and 12 of them in turn hit $(P^1 = [3, 6))$ at time 0.1, then $Pr(P^1 = [3, 6) \mid P^0 = [0, 3), ES^0 =$

$[0, 2), k_3 = [0.2, 0.4)) = 12/132 = 0.091$.

As a result, we can obtain a DBN approximating the idealized DBN induced by introducing the conditional independences to $\mathcal{MC}_{ideal}$. It is not difficult to show that a canonical Markov chain can be recovered from the DBN (Nunez, 1989). We note that this Markov chain is an approximation of $\mathcal{MC}_{ideal}$. In next section, we discuss the error between them due to the following: (i) factorizing $\mathcal{MC}_{ideal}$ into the idealized DBN is based on assumed conditional independences,(ii) the N trajectories used for the construction of the approximated DBN are generated through numerical integration, and (iii) the sample size N is finite.

4.3.1 Error Analysis

The error induced by the numerical integration will depend on the method adopted. For example, the step's errors of Euler's method and fourth-order Runge-Kutta method described in Section 2.3.1 are $O(h^2)$ and $O(h^5)$ respectively, where h is the step size. In general, the error induced by the pth-order numerical integration method is $O(h^{p+1})$ (Press et al., 1992) and it will tend to 0 as h tends to 0 or p tends to ∞.

Further, numerical integration methods compute $z_i(t+\delta)$ using $z_i(t)$ and the values, at time t, of other variables/parameters that appear in the right-hand side of the equation for dz_i/dt (see the formulas of Euler's method and RK4 in Section 2.3.1). In other words, the value of $z_i(t + \delta)$ only depends on the values of $z_j(t)$ where $z_j \in Pa(z)$. Hence the independence assumption is consistent with the numerical integration methods. There will be no additional error induced by the independence assumption when Δt in the DBN equals to δ used for generating the trajectories.

Since N is finite, there will be an error between the conditional probabilities computed using the N trajectories and the ones induced by $\mathcal{MC}_{ideal}$. By the central limit theorem (Durrett, 2004), this error can be probabilistically bounded. For each entry of the DBN, let $\hat{r}$ represent the conditional probability computed via sampling and r

be the actual one induced by the $\mathcal{MC}_{ideal}$, we have:

Proposition 4.4. *Suppose the number of samples is N. Then ϵ will be the error with probability c between $\hat{r}$ in the DBN approximation and the corresponding r induced by the $\mathcal{MC}_{ideal}$ where*

$$\epsilon = \phi^{-1}(\frac{c+1}{2})\sqrt{\frac{r(1-r)}{N}}$$

with $\phi(x) = \frac{1}{\sqrt{2\pi}}\int_{-x}^{x} e^{-y^2/2}dy$.

Proof. Let X be a random variable such that $X = 1$ ($X = 0$ resp.) denote the event that a sample trajectory passes (not passes resp.) a discrete state $\mathbf{s}$ at time $d{\cdot}\Delta t$. Hence X will have a *Bernoulli distribution* with parameter p_{ij} with $\mu = r$ and $\sigma^2 = r(1-r)$. If $X_1, X_2, \ldots, X_N$ are the N measurements, by *Central Limit Theorem*, we have:

$$P\{-\epsilon \leq \frac{\sum_{i=1}^{N} X_i}{N} - \mu \leq \epsilon\} \approx 2\phi(\epsilon\frac{\sqrt{N}}{\sigma}) - 1$$

where ϵ is the error and $\phi(x) = \frac{1}{\sqrt{2\pi}}\int_{-x}^{x} e^{-y^2/2}dy$. Thus,

$$\epsilon = \phi^{-1}(\frac{c+1}{2})\sqrt{\frac{r(1-r)}{N}} \text{ with probability } c.$$

□

Therefore, given an error bound ϵ and a confidence level c, we can compute N, the number of samples required to get an error less than or equal to ϵ with likelihood c. For instance, let $\epsilon = 0.01$ and $c = 0.95$. To estimate $Pr(P^1 = [3,6) \mid P^0 = [0,3), ES^0 = [0,2), k_3 = [0.2, 0.4))$, we need $\lceil(\sqrt{0.091(1-0.091)} \cdot \phi^{-1}((0.95+1/2)/0.01)^2\rceil = 3178$ samples. Further, this error will tend to 0 with probability 1 as N tends to ∞.

The above error analysis is preliminary. It mainly aims to show that the error between our approximation and $\mathcal{MC}_{ideal}$ induced by the pathway dynamics can tend to 0 under certain condition. In practice, to construct the DBN, one will need to make

a chose of N. However, it is difficult to determine N with guaranteed error bounds since $\mathcal{MC}_{ideal}$ is not computable. Hence one must make a pragmatic choice of N. In the next section, we present several sampling methods and the corresponding ways of determining N.

4.3.2 Sampling Methods

Direct sampling

Since we have n variables and m parameters, the N sampled initial values vectors should be picked from a $(n+m)$-dimensional space. According to the prior distribution, the value of each variable or known parameter often lies in one interval. For instance, the value of species E in our running example lies in $[8, 10]$. Thus, for existing models or for those for which the parameter estimation has already been carried out, we can randomly pick a value for each variable/parameter according to their marginal prior and then form a vector of initial values. We term this type of sampling as *direct* sampling. This method does not ensure any coverage. One can determine N based on the computational time one would like to spend on the DBN construction.

Global sampling

However, the value of an unknown parameter will range over all its intervals. For instance, in our running example, the value of each unknown parameter should be sampled from 5 intervals. Thus if we have u unknown parameters whose value spaces have been discretized to K intervals each, one will require a sample size of $N = J \cdot K^u$ to ensure J samples for each possible combinations of interval values of the unknown parameters. We term this as *global* sampling. Thus the number of samples this method would require will be exponential in the number of unknown parameters. This will often be an unacceptably large number.

Local sampling

At the other end of the spectrum, we can pick J samples for each possible interval of variables or parameters. The idea is to randomly choose a point within the target interval I of, say, the variable x_i and then arbitrarily extend this point value to an $(n+m)$-dimensional vector over the allowed intervals of the other variables and parameters. We term this as *local* sampling and it will require a sample size of $N = (n+m) \cdot J \cdot K$ with coverage of J per interval. Hence local sampling will require a much smaller sample size. However, it can not guarantee adequate coverage for *combinations* of interval values of the parameters.

To bring this out through an artificial but simple case, suppose x takes values in the interval I_1 and I_2 while y takes values in the intervals I_3 and I_4. To get a local coverage of 100 samples per interval, we may pick 100 points from I_1 and extend it by combining it with a random value for y (which will fall in I_3 or I_4). Let S_1 be the set of such samples. Similarly let S_2 be the set of 100 samples obtained by picking 100 points from I_2 and extending each of them randomly to a value for y. Finally, let S_3 (S_4) be the 100 samples obtained by picking 100 values from I_3 (I_4) and extending each of them randomly to a value for x. In this way, with 400 samples we will be able to guarantee a minimum of 100 hits for each of the intervals $\{I_1, I_2, I_3, I_4\}$. However suppose the y-values of all the samples in S_1 (S_2) fall in S_3 (S_4) and the x-values of all the samples in $S_3(S_4)$ fall in $S_1(S_2)$. Then none of the 400 samples will fall in $I_1 \times I_4$ ($I_2 \times I_3$) and hence we will get 0-coverage for the *combination* of this pair of intervals!

To ensure that we are exploring the ODEs dynamics adequately, we need to ensure that all the possible combinations of interval values of unknown parameters governing any single equation are being sampled an adequate number of times. Otherwise, the probabilistic inference we need to to perform on the DBN approximation during parameter estimation and sensitivity analysis will have poor quality.

Equation sampling

Hence to get good coverage in the presence of many unknown parameters, one will have to resort to more sophisticated methods. Here we propose a method called *equation* sampling by which numerical simulations can be carried out in the presence of unknown parameters while ensuring that the local dynamics defined by the individual equations are being explored adequately. To bring out the main idea, suppose the equation for the variable x_i involves the unknown parameters k_1 and k_2 and the values of k_1 (k_2) have been divided into three intervals I_1 (I_1') , I_2 (I_2') and I_3 (I_3'). Then for a specific combination of intervals, say I_2 and I_3' we pick 100 samples such the k_1 value lies in I_2 and the k_2 value lies in I_3' for each of the samples. In this way we can pick 900 samples which ensure that there are at least 100 samples for each combination of interval values for k_1 and k_2. In general, we will be able provide a coverage of J samples for each possible combination of interval values of the unknown parameters in the equation for each variable with the help of $N = n \cdot J \cdot K^R$ samples, where R is the maximal number of unknown parameters appearing in *an equation*. Since the positive terms (negative terms) in the differential equation of a species describe the rates of reactions that producing (consuming) this species, equation sampling will provide of a coverage of all possible local conditions that determines the dynamics of a single species. Thus, with this type of sampling, the quality of model analysis tasks can be ensured with an acceptable sample size.

4.3.3 Optimizations

Various optimizations can be developed to reduce the practical complexity of the DBN construction. Specifically, the sampling process followed by the generation of a trajectory can be easily parallelized and executed on a computing cluster. In addition, the CPTs can be stored using a sparse representation. Yet another optimization is to split up a "fat" node with a large number of parents into nodes with smaller fan-

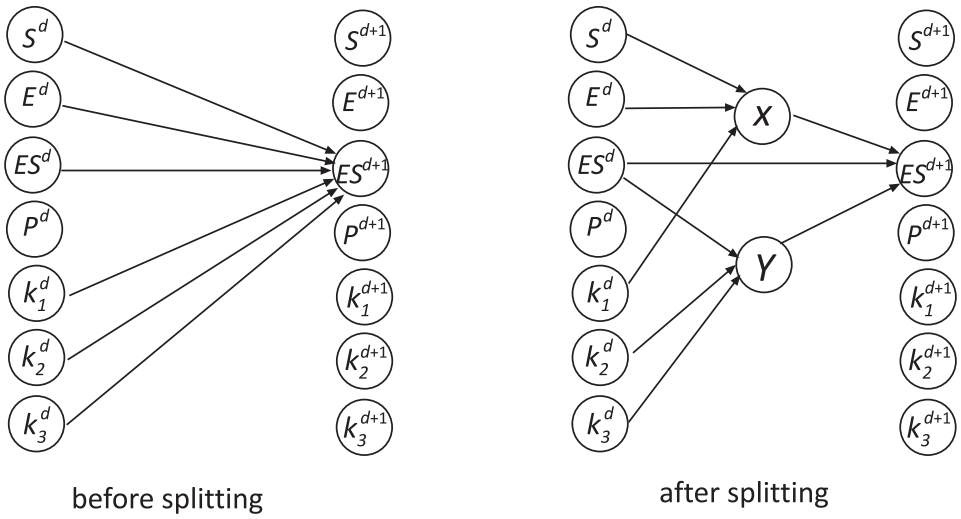


Figure 4.2: Node splitting.

in degrees and thus reduce R. As shown in Figure 4.2, the reduction can be based on the form of the differential equation associated with the variable. Given that $dES/dt = k_1 \cdot S \cdot E - (k_2 + k_3) \cdot ES$, we introduce two internal nodes X and Y, where X corresponds to the positive term of dES/dt, namely, $k_1 \cdot S \cdot E$ and Y corresponds to the negative term $(k_2 + k_3) \cdot ES$. As a result, R can be reduced from 6 to 3. We note however, at present we consider this optimization only to reduce the sizes of the CPTs and not to reduce the number of samples when using the equation sampling method.

4.4 Discussion

In this chapter, we have described our probabilistic approximation scheme for pathway dynamics specified as a systems of ODEs. For sure, the construction of the DBN approximation will involve a significant computational effort but it is a one time cost and significant optimizations can be deployed. Moreover, once the DBN approximation has been constructed, many of the analysis tasks can be performed very efficiently and

the one time cost of constructing the DBN approximation can be easily amortized. The experimental results presented in Chapter 6 will support this claim.

Chapter 5

Analysis Methods

In this chapter we present some of the analysis techniques we have developed for the DBN representation. These techniques are founded on a basic Bayesian inference method realized via the FF (Factored Frontier) algorithm (Murphy and Weiss, 2001). Specifically we develop parameter estimation and sensitivity analysis methods for the DBN approximation. Our goal here is not to develop new algorithms to solve these problems. Rather, we wish to demonstrate how standard techniques for tackling these problems can be adapted to DBN approximation framework.

5.1 Probabilistic Inference

Given a Bayesian network, some observed evidence and some knowledge about the distribution of values of a set of variables, Bayesian inference aims to compute posterior distribution for a set of query variables. In our setting, the observed evidence will consist of known initial conditions and parameters as well as experimental data. Query variables will typically be selected random variables in the DBN approximation.

Exact inference

As the state space of our DBN is discrete, exact inference is always theoretically possible (Murphy, 2002). In this case, there will be no error induced by performing inference. However, the exact inference often be computationally prohibitive. For instance, the time and space complexity of the frontier algorithm (Zweig, 1996) -a standard exact inference algorithm- is $O(\hat{d}(n+m)K^{n+m+2})$, where as before, $\hat{d}$ is the number of time slices of the DBN, n is the number of variables, m is the number of parameters and K is the maximal number of intervals associated with a variable or rate constant's value domain. Hence for large pathway models, we must resort to approximate inference methods.

Boyen-Koller algorithm

The Boyen-Koller (BK) algorithm (Boyen and Koller, 1998, 1999) is a standard algorithm for approximate inference on DBN. It approximates the joint distribution over the variables in one time slice (a belief state) as a product of marginals over clusters of variables. For example: $Pr(x_1^d, x_2^d, x_3^d) \approx Pr(x_1^d, x_2^d)Pr(x_3^d)$. When we do inference using BK in our setting, starting with the approximated belief state of current time slice $Pr(x_1^d, x_2^d, \ldots, x_{n+m}^d)$, we perform one step of exact Bayesian updating to get $Pr(x_1^{d+1}, x_2^{d+1}, \ldots, x_{n+m}^{d+1})$, which will be then projected as a product of marginals over clusters of variables.

A great advantage of the BK algorithm is that the error induced can be shown to be bounded over time. The detailed proof can be found in Boyen and Koller (1998). Intuitively, even though projection introduces an error at every time step, the stochastic nature of the transitions and the informative nature of the observations, will reduce the error sufficiently to stop it building up.

Factored frontier algorithm

Unfortunately, for large pathway models, the one-step exact updating of BK algorithm can still be intractable. Therefore, we adopt a more aggressive form of approximation than BK, namely, the factored frontier (FF) algorithm (Murphy and Weiss, 2001). The FF algorithm approximates, at each time point, joint distributions as products of marginal distributions. For example: $Pr(x_1^d, x_2^d, x_3^d) \approx \prod_{i=1}^{3} Pr(x_i^d)$. Hence the posterior distribution will be computed according to:

$$Pr(x_i^d|D) = \sum_I (Pr(x_i^d|Pa(x_i^d) = I) \prod_{u \in Pa(x_i^d)} Pr(u|D)). \tag{5.1}$$

Here $Pr(u|D)$ are the marginal distributions over the parents, D is the evidence regarding initial conditions and experimental observations, and $Pa(x_i)$ denotes the parents of x_i. The implementation of FF is straightforward. By storing $Pr(x_i^d|Pa(x_i^d))$ in the conditional probability tables and propagating $Pr(u|D)$ to the next time point, we can use (5.1) to compute $Pr(x_i^d|D)$. The time complexity of this algorithm is $O(\hat{d}(n+m)K^{R+1})$, where as before, K is the maximal number of intervals associated with a variable or rate constant's value domain and R is the maximal number of parents a node can have.

Except for the one-step exact updating, FF is very close to the fully factorized BK that use one cluster per variable. Currently, there are no error analysis for FF. However, experimentally comparisons results of exact, BK and FF show that the error induced by FF is acceptable and close to the fully factorized BK (see Figure 5.1). Here we define the total L_1 error in the marginals for variable x_i at time slice d as:

$$\epsilon_i^d = \sum_I |Pr(x_i^d = I) - \widehat{Pr}(x_i^d = I)| \tag{5.2}$$

where $\widehat{Pr}(\cdot)$ is the exact posterior and $Pr(\cdot)$ is the approximate posterior. The results shown in Figure 5.1 indicate that the error induced by FF is close to fully factorized

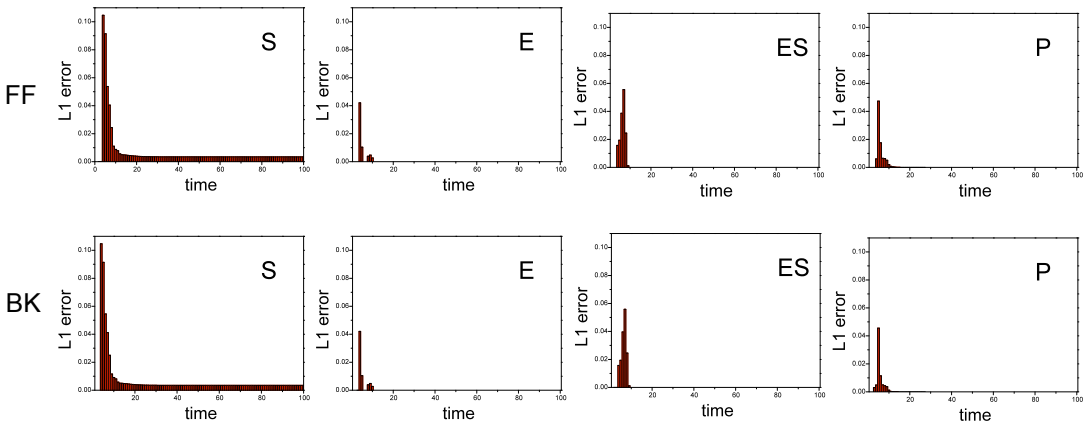


Figure 5.1: Comparison of exact, fully factorized BK and FF inference results of the enzyme-kinetic system.

BK.

Using the FF algorithm, and with some additional computations, many queries can be answered. For instance, given the initial conditions, a single run of FF algorithm will infer the marginal distributions of each variable at every time point. These probability distributions can then be used to validate the model by comparing them with experimental data. Flow cytometry data may provide direct information about the probability distributions of species concentration in a cell population. For such data, we may discretize it into distributions over intervals and supply it to the FF algorithm. On the other hand, western blot data, which is more common, will provide the averages of species concentration in a cell population. Suppose we have the data for x_i at time $d \cdot \Delta t$, denoted as $D_{x_i}^{d \cdot \Delta t}$. Note the marginal distribution of x_i^d inferred by FF algorithm is over discrete values $\mathcal{I}_i$. To compute the real-valued "mean" of x_i^d that can be compared with $D_{x_i}^{d \cdot \Delta t}$, we identify each interval $I = [l, u)$ in $\mathcal{I}$ with its mid-point $(l + u)/2$. Then the expected value $E(x_i^d)$ can be computed and compared with $D_{x_i}^{d \cdot \Delta t}$. The rationale of this approach is: (1) the western blot data $D_{x_i}^{d \cdot \Delta t}$ can be assumed to be the observations for the mean computed from the marginal probabilities induced by

$\mathcal{MC}_{ideal}$, denoted as $\hat{E}(x_i^{d\cdot\Delta t})$; and (2) the $E(x_i^d)$ we use to compare with the data is an approximation of $\hat{E}(x_i^{d\cdot\Delta t})$ with bounded error. Here we estimate the error between $\hat{E}(x_i^{d\cdot\Delta t})$ and $E(x_i^d)$ and show it is bounded as follows:

Proposition 5.1. *For any x_i at each time point $d\cdot\Delta t$, we have*

$$|E(x_i^d) - \hat{E}(x_i(d\cdot\Delta t)| \leq r_{\mathcal{I}_i}, \text{ where } r_{\mathcal{I}_i} = \max_{[u_k,l_k)\in\mathcal{I}_i}\{\frac{u_k - l_k}{2}\}. \tag{5.3}$$

Proof. As discussed in Section 3.2, Φ_t is a bijective and continuous function. Further, it can be proved that Φ_t is also differentiable (Hirsch et al., 2004). Given the prior probability density function Υ^0 we can define the probability density function Υ^t for every t as:

$$\Upsilon^t(\mathbf{z}) := \Upsilon^0(\Phi_t^{-1}(\mathbf{z}))\left|\det\left(J_{\Phi_t^{-1}}(\mathbf{z})\right)\right| \tag{5.4}$$

where $J_{\Phi_t^{-1}}$ denotes the Jacobian of the inverse of Φ. We then define a probability density function g_{x_i} for each variable x_i at time point t by marginalization:

$$g_{x_i}^t(z) := \int\ldots\int \Upsilon^t(z_1, z_2, \ldots, z_{i-1}, z, z_{i+1}\ldots, z_{n+m})dz_1dz_2\ldots dz_{i-1}dz_{i+1}\ldots dz_{n+m} \tag{5.5}$$

Similarly, we define a probability mass function m_{x_i} for each variable x_i at time point $d\cdot\Delta t$ by marginalize $Pr(\mathbf{s}, d)$ in the $\mathcal{MC}_{ideal}$ induced by a discretization $\mathcal{I}$:

$$m_{x_i}^d(s) := \sum_{s_1}\sum_{s_2}\ldots\sum_{s_{i-1}}\sum_{s_{i+1}}\ldots\sum_{s_{n+m}} Pr((s_1, s_2, \ldots, s_{i-1}, s, s_{i+1}\ldots, s_{n+m}), d) \tag{5.6}$$

Thus, we have

$$m_{x_i}^d(s) = \int_l^u g_{x_i}^{d\cdot\Delta t}(z)dz, \text{ where } s = [l, u) \in \mathcal{I}_i. \tag{5.7}$$

Let $\mathcal{I}_i = \{I_1 = [v^0, v^1), I_2 = [v^1, v^2), \ldots, I_L = [v^{L-1}, v^L]\}$, we set $r_i = (v^i - v^{i-1})/2$

and denote the maximal r_i as r_{max}.

$$
\begin{aligned}
E(x_i^d) &= \sum_{\mathcal{I}_i} \frac{v^{i-1}+v^i}{2} \cdot m_{z_i}^d(s) \\
\hat{E}(x_i(d\cdot\Delta t)) &= \int_{v^0}^{v^L} z \cdot g_{x_i}^{d\cdot\Delta t}(z)dz \\
&= \int_{v^0}^{v^1} z \cdot g_{x_i}^{d\cdot\Delta t}(z)dz + \int_{v^1}^{v^2} z \cdot g_{x_i}^{d\cdot\Delta t}(z)dz + \ldots + \int_{v^{L-1}}^{v^L} z \cdot g_{x_i}^{d\cdot\Delta t}(z)dz \\
&\leq \int_{v^0}^{v^1} v^1 \cdot g_{x_i}^{d\cdot\Delta t}(z)dz + \int_{v^1}^{v^2} v^2 \cdot g_{x_i}^{d\cdot\Delta t}(z)dz + \ldots + \int_{v^{L-1}}^{v^L} v^L \cdot g_{x_i}^{d\cdot\Delta t}(z)dz \\
&= v^1 \int_{v^0}^{v^1} g_{x_i}^{d\cdot\Delta t}(z)dz + v^2 \int_{v^1}^{v^2} g_{x_i}^{d\cdot\Delta t}(z)dz + \ldots + v^L \int_{v^{L-1}}^{v^L} g_{x_i}^{d\cdot\Delta t}(z)dz \\
&= v^1 m_{x_i}^d(I_1) + v^2 m_{x_i}^d(I_2) + \ldots + v^L m_{x_i}^d(I_L) = \sum_{\mathcal{I}_i} v^i m_{x_i}^d(I_i).
\end{aligned}
$$

Similarly, we have $\hat{E}(x_i(d\cdot\Delta t)) \geq \sum_{\mathcal{I}_i} v^{i-1} m_{x_i}^d(I_i)$.

Thus, $|E(x_i^d) - \hat{E}(x_i(d\cdot\Delta t))| \leq \sum_{\mathcal{I}_i} r_i \cdot m_{x_i}^d(I_i) \leq r_{max}$. □

5.2 Parameter Estimation

Lack of knowledge about the parameters and hence the need to perform parameter estimation using limited data is a major bottleneck to pathway modeling. As reviewed in Section 2.4, current approaches to parameter estimation formulate it as a non-linear optimization problem (Banga, 2008). A typical procedure will involve searching in a high dimensional solution space, in which each point represents a vector of parameter values. Whether a point is good or not is measured by an objective function, which will capture the difference between experimental data and prediction generated by simulations using the corresponding parameters.

For a large pathway model, one often needs to evaluate a very large number of solution points involving a numerical integration for each evaluation. This makes the process computationally intensive. The DBN representation allows us search for good

parameter values using a *two-stage method*. Due to the discretized nature of the DBN approximation, the solution space is transformed into a rectilinear grid tessellated by hyperrectangles that we call *blocks*. An important observation is that kinetic parameters are often robust (Gutenkunst et al., 2007). In other words, the points around a good solution in the search space will also have relatively small objective values. Note that searching in this discrete space with finite number of blocks is much simpler than in the original continuous solution space. Thus, instead of searching point by point in the solution space, we can first search for a few promising blocks and then take a closer look within these small blocks. Guided by this intuition, the general scheme of our "grid search" algorithm consists of two stages:

(1) identify good blocks,

(2) do local search within candidate blocks.

For executing Stage (1), we can apply any standard search algorithms over the discretized search space. As this space is much smaller than the original one, simple direct search algorithm such as Hooke & Jeeves's search (Hooke and Jeeves, 1961) can be adopted and the overall search process will only require a small number of evaluations of the objective function.

A block dictates a combination of intervals of parameter values. In order to evaluate the goodness of a block, we execute FF algorithm once by supplying the chosen parameter values -in terms of intervals- as evidence. Then the objective value can be computed by comparing the expected value of marginal distributions with the experimental data as described in the previous subsection.

Stage (1) will return a maximum likelihood estimate of a combination of intervals of parameter values. Through probabilistic inference techniques, it can be used to carry out model analysis for fixed distribution of initial concentrations. Hence, in principle, given the noisy and limited experimental data and the high dimensionality

of the system, one could stop with Stage (1) and try to work an interval of values for each parameter rather than a point value.

We note that Stage (2) is necessary only when we want to estimated the real values of parameters and use the ODE model too for some analysis and simulations that requires perturbing the initial concentrations and a finer granularity of parameters. For instance, in the case studies presented in Chapter 6: in the first and the second case study, we skip Stage (2), whereas in the third case study, we execute Stage (2) and further estimate the real values for unknown parameters for conducting *in silico* experiments such as varying initial concentrations.

For executing Stage (2), we treat the resulting combination of intervals of parameter values from Stage (1) as the (drastically reduced) search space. For an m-dimensional search space with K discretized intervals for each dimension, Stage (1) can reduce the search space by a factor of $1/K^m$. For most parameter estimation methods, reducing the solution space increases the chance of randomly picking good starting points which in turn will lead to faster convergence. Hence the reduction of solution space we achieved using the DBN approximation contributes in this way too in improving the performance of the parameter estimation procedure.

Note that during the construction of the DBN approximation, if we do not have any knowledge about the prior distributions of unknown parameters, we can assume they are uniformly distributed within their ranges. Then after filling up all the entries of the CPTs of the DBN, the FF algorithm will be able to evaluate the goodness of any block in the discretized search space.

Example (Continued from the enzyme catalysis example presented in Chapter 4) Assume that only k_1 and k_2 are unknown parameters and that $k_3 \in [0.2, 0.4)$. Assume further that we have experimental data for S and P at time points $\{1, 2, 5, 10\}$. We then construct a DBN approximation according to a prior distribution that the initial

values $\mathbf{z}_0 = (S, E, ES, P, k_1, k_2, k_3)$ are uniformly distributed within the hypercube $[12, 15] \times [8, 10] \times [0, 2) \times [0, 3) \times [0, 1] \times [0, 1] \times [0.2, 0.4)$. Since $|\mathcal{I}_{k_1}| = |\mathcal{I}_{k_2}| = 5$, the solution space $[0, 1] \times [0, 1]$ is discretized into 25 blocks. In phase(1), we try to search for good blocks among the 25 blocks. Suppose we conduct a Hooke & Jeeves's search and evaluate blocks one by one. For instance, to evaluate block $([0.4, 0.6), [0.2, 0.4))$, we set $Pr(k_1 = [0.4, 0.6) = 1$ and $Pr(k_2 = [0.2, 0, 4) = 1$ (the distributions of S^0, E^0, ES^0, P^0, and k_3 are the same as the prior distribution for constructing the DBN approximation) and execute FF algorithm. If the inferred distribution of S^{10} is $\{Pr(S^{10} = [12, 15]) = 0.6, Pr(S^{10} = [9, 12)) = 0.4\}$, we have $E(S^{10}) = (15-12)/2 \cdot 0.6 + (12-9)/2 \cdot 0.4 = 12.3$. Then we can compute the weighted square root error between $E(S^{10})$ and the available data. After searching suppose we find that $([0.2, 0.4), [0.4, 0.6))$ is the best block (i.e. it has the minimal objective value), we then can either execute phase(2) by searching within the solution space $[0.2, 0.4) \times [0.4, 0.6)$ or just return $\{k_1 \sim U(0.2, 0.4), k_2 \sim U(0.4, 0.6)\}$ as a probabilistic estimate (U stand for uniform distribution).

5.3 Global Sensitivity Analysis

As discussed in Section 2.5.1, sensitivity analysis has been used to identify the critical parameters in signal transduction (van Riel, 2006). To overcome the limitations of traditional local sensitivity analysis methods, global methods have been proposed recently such as multi-parametric sensitivity analysis (MPSA) (Cho et al., 2003). The MPSA procedure consists of:

(1) draw samples from parameter space and for each combination of parameters, compute the weighted sum of squared error between experimental data and predictions generated by selected parameters;

(2) classify the sampled parameter sets into two classes (good and bad) using a threshold error value;

(3) plot the cumulative frequency of the parameter values associated with the two classes;

(4) evaluate the sensitivities as the Kolmogorov-Smirnov statistic (Sheskin, 2004) of cumulative frequency curves.

Signaling pathway models often contain a large number of parameters. Hence it is necessary to sample a representative set from all possible combinations of parameter values. To improve this process, (Zi et al., 2005) adopts Latin Hypercube Sampling (LHS) since it requires fewer samples while guaranteeing that individual parameter ranges are evenly covered. Briefly, the range of each parameter is divided into K equal-sized intervals. Then for each parameter, one randomly sample K values, one from each interval of the parameter. Then to generate combinations of parameter values which is samples for MPSA, values are chosen in a random order from the K values for each parameter. This method helps to computationally manage the large number of parameters being varied simultaneously, while ensuring maximal sampling through each parameter dimension (McKay et al., 2000). In our DBN setting, MPSA can be performed in a similar manner using LHS since the parameter space is discretized into blocks. In addition, the number of samples used to reach convergence is reduced since we can quickly evaluate the goodness of the whole block using the FF algorithm instead of having to draw samples from a block.

Chapter 6

Case Studies

The algorithms presented in previous chapters for constructing and analyzing DBN approximations have been implemented in our software tool called PAthway Dynamics Approximator (PADA). PADA is open-source and is freely available at our website[1]. It is a Java program that supports the import of ODE-based pathway models in SBML format. It can generate parallelized code, to be executed on computer clusters, for the construction process of DBN approximations, as well as sequential code, to be executed on a single CPU, for carrying out probabilistic inference, parameter estimation and global sensitivity analysis.

In this chapter, we present three case studies which demonstrate the applicability of our probabilistic approximation technique. The first case study (Section 6.1) involves a signaling network built by Brown et al. (2004), which aims to study the influence of the nerve growth factor (NGF) and the mitogenic epidermal growth factor (EGF) in rat pheochromocytoma (PC12) cells. The second case study (Section 6.2) deals with a signaling network studied by (Goldbeter and Pourquie, 2008) to investigate a remarkable example of biological rhythms, namely, the segmentation clock. These two case studies validate our techniques and demonstrate good performance. The

[1] `http://www.comp.nus.edu.sg/~rpsysbio/pada`

results reported show that the constructed DBN approximations have high quality and the efficiency of performing parameter estimation and global sensitivity analysis has been improved. It is worth noting that, in the first case study, we also compare the performance (Section 6.1.4) of different sampling techniques presented in Chapter 4, as well as the accuracies of approximations constructed using different discretization schemes (Section 6.1.4). Furthermore, we also identified critical parameters in signal transduction of the two pathways via rapid global sensitivity analysis.

The evaluations of our DBN approximation approach in the first and the second case studies were done using synthetic (training) data. We further demonstrate the capability and effectiveness of this approach by the third study (Section 6.3), which is an integrated computational and experimental study of the regulatory mechanisms of the human complement system. In this study, we built and analyzed a "live" pathway model for the complement system in collaboration with Prof Ding Jeak Ling's group in Department of Biological Science, National University of Singapore and clinicians from National University Hospital (Liu et al., 2010). To overcome the computational challenges resulting from the large model size, we applied our techniques to train the model on *in vivo* experimental data and explored the key network features of the model. The results show the capability of our approach to deal with a large bio-pathway especially in the context of performing tasks such as parameter estimation and global sensitivity analysis. More importantly, this study has resulted in some crucial insights into the complement regulatory mechanisms and has the potential to contribute to the development of complement-based immunomodulation therapies.

6.1 The EGF-NGF Signaling Pathway

PC12 cells are a valuable model system in neuroscience. They proliferate in response to EGF stimulation but differentiate into sympathetic neurons in response to NGF. This interesting phenomenon has been intensively studied (Kholodenko, 2007). It has

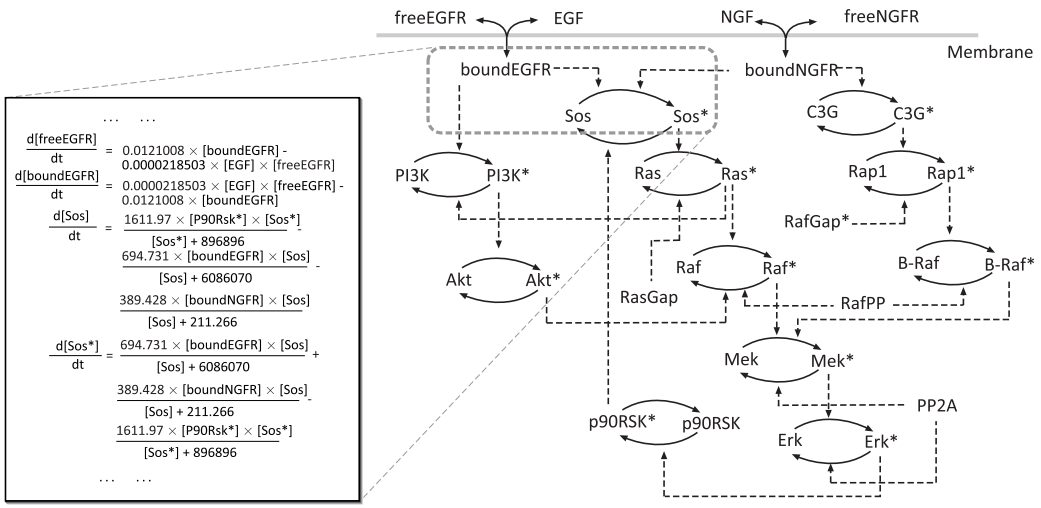


Figure 6.1: The reaction network diagram of the EGF-NGF pathway (Brown et al., 2004)

been reported that the signal specificity is correlated with different Erk dynamics. Specifically, a transient activation of Erk1/2 has been associated with cell proliferation, while a sustained activity has been linked to differentiation. How EGF and NGF affect the dynamics of active Erk through a network of intermediate signaling proteins is shown schematically in Figure 6.1.

This model includes a common pathway to Erk through Ras shared by both the EGFR and NGFR, and also two important side branches through PI3K and C3G. This introduces multiple feedback loops leading to sophisticated dynamics. The ODE model of this pathway is available in the BioModels database (Le Novere et al., 2006). It consists of 32 differential equations and 48 associated rate parameters (estimated from multiple sets of experimental data).

6.1.1 Construction of the DBN approximation

To construct the DBN approximation, we first derived its graph from its ODEs (see Table 6.1). We then discretized the ranges of each variable and parameter into 5

equal-size intervals and fixed the time step Δt to be 1 minute. These choices were made mainly in order to proceed with the DBN construction smoothly but without trivializing the effort. Further, the experimental data (western blot) is such that 5 uniform intervals seemed a reasonable choice. However our construction can be easily extended to non-uniform values intervals and time points. To fill up the conditional probability tables associated with the nodes, 3×10^6 trajectories were generated up to 100 mins by sampling initial states and parameters from the prior which are assumed to be uniform distributions over certain intervals (Table 6.2 and Table 6.3). Since we planned to study the effectiveness of our DBN based parameter estimation method (Section 5.2), we singled out 20 of the 48 parameters to be unknown (marked with * in Table 6.3) . When generating the 3×10^6 initial states, the sampled initial values of these parameters were chosen from their full range of possible values and not biased towards any specific intervals.

These samples were generated using the direct sampling method. We recall that in this method, the initial values of those trajectories are according to prior distribution (except for the parameters designated to be "unknown" as described above). Specifically, we randomly pick a value for each variable/parameter according to their marginal prior and then form a vector of initial values. The computational workload was distributed on 10 Opteron 2.2GHz processors in a cluster. It took around 4 hours to construct the DBN approximation. All the subsequent experiments reported below were done using an Intel Xeon 2.8GHz processor.

6.1.2 Probabilistic inference

To test the quality of our approximation, we implemented Monte Carlo integration for the ODE model to get good estimates by sampling and averaging. Specifically, we numerically generated a number of random trajectories -according to the prior- using ODEs and computed the average values of the variables at the chosen time

Name	Variable	Parents
EGF	x_1	k_1, x_1, x_3, k_2, x_4
NGF	x_2	k_3, x_2, x_5, k_4, x_6
free EGF Recepter	x_3	k_1, x_1, x_3, k_2, x_4
bound EGF Recepter	x_4	k_1, x_1, x_3, k_2, x_4
free NGF Recepter	x_5	k_3, x_2, x_5, k_4, x_6
bound NGF Recepter	x_6	k_3, x_2, x_5, k_4, x_6
inactive Sos	x_7	$k_9, x_{10}, x_8, x_8, k_{10}, k_5, x_4, x_7, x_7, k_6, k_7, x_6, x_7, x_7, k_8$
active Sos	x_8	$k_9, x_{10}, x_8, x_8, k_{10}, k_5, x_4, x_7, x_7, k_6, k_7, x_6, x_7, x_7, k_8$
inactive P90Rsk	x_9	$k_{27}, x_{21}, x_9, x_9, k_{28}$
active P90Rsk	x_{10}	$k_{27}, x_{21}, x_9, x_9, k_{28}$
inactive Ras	x_{11}	$k_{11}, x_{11}, x_{11}, k_{12}, k_{13}, x_{13}, x_{12}, x_{12}, k_{14}$
active Ras	x_{12}	$k_{11}, x_{11}, x_{11}, k_{12}, k_{13}, x_{13}, x_{12}, x_{12}, k_{14}$
active RasGap	x_{13}	x_{13}
inactive Raf	x_{14}	$k_{15}, x_{12}, x_{14}, x_{14}, k_{16}, k_{45}, x_{32}, x_{15}, x_{15}, k_{46}, k_{35}, x_{25}, x_{15}, x_{15}, k_{36}$
active Raf	x_{15}	$k_{15}, x_{12}, x_{14}, x_{14}, k_{16}, k_{45}, x_{32}, x_{15}, x_{15}, k_{46}, k_{35}, x_{25}, x_{15}, x_{15}, k_{36}$
inactive B-Raf	x_{16}	$k_{43}, x_{29}, x_{16}, x_{16}, k_{44}, k_{47}, x_{32}, x_{17}, x_{17}, k_{20}$
active B-Raf	x_{17}	$k_{43}, x_{29}, x_{16}, x_{16}, k_{44}, k_{47}, x_{32}, x_{17}, x_{17}, k_{20}$
inactive Mek	x_{18}	$k_{17}, x_{15}, x_{18}, x_{18}, k_{18}, k_{19}, x_{17}, x_{18}, x_{18}, k_{48}, k_{21}, x_{31}, x_{19}, x_{19}, k_{22}$
active Mek	x_{19}	$k_{17}, x_{15}, x_{18}, x_{18}, k_{18}, k_{19}, x_{17}, x_{18}, x_{18}, k_{48}, k_{21}, x_{31}, x_{19}, x_{19}, k_{22}$
inactive Erk	x_{20}	$k_{23}, x_{19}, x_{20}, x_{20}, k_{24}, k_{25}, x_{31}, x_{21}, x_{21}, k_{26}$
active Erk	x_{21}	$k_{23}, x_{19}, x_{20}, x_{20}, k_{24}, k_{25}, x_{31}, x_{21}, x_{21}, k_{26}$
inactive PI3K	x_{22}	$k_{29}, x_4, x_{22}, x_{22}, k_{30}, k_{31}, x_{12}, x_{22}, x_{22}, k_{32}$
active PI3K	x_{23}	$k_{29}, x_4, x_{22}, x_{22}, k_{30}, k_{31}, x_{12}, x_{22}, x_{22}, k_{32}$
inactive Akt	x_{24}	$k_{33}, x_{23}, x_{24}, x_{24}, k_{34}$
active Akt	x_{25}	$k_{33}, x_{23}, x_{24}, x_{24}, k_{34}$
inactive C3G	x_{26}	$k_{37}, x_6, x_{26}, x_{26}, k_{38}$
active C3G	x_{27}	$k_{37}, x_6, x_{26}, x_{26}, k_{38}$
inactive Rap1	x_{28}	$k_{39}, x_{27}, x_{28}, x_{28}, k_{40}, k_{41}, x_{30}, x_{29}, x_{29}, k_{42}$
active Rap1	x_{29}	$k_{39}, x_{27}, x_{28}, x_{28}, k_{40}, k_{41}, x_{30}, x_{29}, x_{29}, k_{42}$
active RapGap	x_{30}	x_{30}
active PP2A	x_{31}	x_{31}
active RafPP	x_{32}	x_{32}

Table 6.1: The DBN structure of the EGF-NGF signaling pathway model

Probability distribution
$x_1 \sim U(8801760.0, 1.10022 \times 10^7)$
$x_2 \sim U(401280.0, 501600.0)$
$x_3 \sim U(70400.0, 88000.0)$
$x_4 \sim U(0.0, 17600.0)$
$x_5 \sim U(8800.0, 11000.0)$
$x_6 \sim U(0.0, 2200.0)$
$x_7 \sim U(105600.0, 132000.0)$
$x_8 \sim U(0.0, 26400.0)$
$x_9 \sim U(105600.0, 132000.0)$
$x_{10} \sim U(0.0, 26400.0)$
$x_{11} \sim U(105600.0, 132000.0)$
$x_{12} \sim U(0.0, 26400.0)$
$x_{13} \sim U(105600.0, 132000.0)$
$x_{14} \sim U(105600.0, 132000.0)$
$x_{15} \sim U(0.0, 26400.0)$
$x_{16} \sim U(105600.0, 132000.0)$
$x_{17} \sim U(0.0, 26400.0)$
$x_{18} \sim U(528000.0, 660000.0)$
$x_{19} \sim U(0.0, 132000.0)$
$x_{20} \sim U(528000.0, 660000.0)$
$x_{21} \sim U(0.0, 132000.0)$
$x_{22} \sim U(105600.0, 132000.0)$
$x_{23} \sim U(0.0, 26400.0)$
$x_{24} \sim U(105600.0, 132000.0)$
$x_{25} \sim U(0.0, 26400.0)$
$x_{26} \sim U(105600.0, 132000.0)$
$x_{27} \sim U(0.0, 26400.0)$
$x_{28} \sim U(105600.0, 132000.0)$
$x_{29} \sim U(0.0, 26400.0)$
$x_{30} \sim U(105600.0, 132000.0)$
$x_{31} \sim U(105600.0, 132000.0)$
$x_{32} \sim U(105600.0, 132000.0)$

Table 6.2: Prior (initial) probability distribution of variables

Parameter	Range	Nominal probability distribution
k_1*	$[0, 4.37006 \times 10^{-5}]$	$k_1 \sim U(1.748024 \times 10^{-5}, 2.622036 \times 10^{-5})$
k_2*	$[0, 0.0242016]$	$k_2 \sim U(0.00968064, 0.01452096)$
k_3*	$[0, 2.76418 \times 10^{-7}]$	$k_3 \sim U(1.105672 \times 10^{-7}, 1.658508 \times 10^{-7})$
k_4*	$[0, 0.01447622]$	$k_4 \sim U(0.005790488, 0.008685732)$
k_5	$[0, 1389.462]$	$k_5 \sim U(555.7848, 833.6772)$
k_6	$[0, 1.217214 \times 10^{7}]$	$k_6 \sim U(4868856.0, 7303284.0)$
k_7	$[0, 778.856]$	$k_7 \sim U(311.5424, 467.3136)$
k_8	$[0, 4225.32]$	$k_8 \sim U(1690.128, 2535.192)$
k_9	$[0, 3223.94]$	$k_9 \sim U(1289.576, 1934.364)$
k_{10}	$[0, 1793792.0]$	$k_{10} \sim U(717516.8, 1076275.2)$
k_{11}*	$[0, 64.688]$	$k_{11} \sim U(25.8752, 38.8128)$
k_{12}*	$[0, 71908.6]$	$k_{12} \sim U(28763.44, 43145.16)$
k_{13}	$[0, 3018.72]$	$k_{13} \sim U(1207.488, 1811.232)$
k_{14}	$[0, 2864820.0]$	$k_{14} \sim U(1145928.0, 1718892.0)$
k_{15}*	$[0, 1.768192]$	$k_{15} \sim U(0.7072768, 1.0609152)$
k_{16}	$[0, 124929.2]$	$k_{16} \sim U(49971.68, 74957.52)$
k_{17}*	$[0, 371.518]$	$k_{17} \sim U(148.6072, 222.9108)$
k_{18}	$[0, 9536700.0]$	$k_{18} \sim U(3814680.0, 5722020.0)$
k_{19}	$[0, 250.178]$	$k_{19} \sim U(100.0712, 150.1068)$
k_{20}	$[0, 315896.0]$	$k_{20} \sim U(126358.4, 189537.6)$
k_{21}	$[0, 5.66486]$	$k_{21} \sim U(2.265944, 3.398916)$
k_{22}	$[0, 1037506.0]$	$k_{22} \sim U(415002.4, 622503.6)$
k_{23}*	$[0, 19.70734]$	$k_{23} \sim U(7.882936, 11.824404)$
k_{24}	$[0, 2014680.0]$	$k_{24} \sim U(805872.0, 1208808.0)$
k_{25}	$[0, 17.7824]$	$k_{25} \sim U(7.11296, 10.66944)$
k_{26}	$[0, 6992980.0]$	$k_{26} \sim U(2797192.0, 4195788.0)$
k_{27}*	$[0, 0.0427394]$	$k_{27} \sim U(0.01709576, 0.02564364)$
k_{28}*	$[0, 1527046.0]$	$k_{28} \sim U(610818.4, 916227.6)$
k_{29}*	$[0, 21.3474]$	$k_{29} \sim U(8.53896, 12.80844)$
k_{30}	$[0, 369824.0]$	$k_{30} \sim U(147929.6, 221894.4)$
k_{31}	$[0, 0.1542134]$	$k_{31} \sim U(0.06168536, 0.09252804)$
k_{32}	$[0, 544112.0]$	$k_{32} \sim U(217644.8, 326467.2)$
k_{33}*	$[0, 0.1132558]$	$k_{33} \sim U(0.04530232, 0.06795348)$
k_{34}*	$[0, 1307902.0]$	$k_{34} \sim U(523160.8, 784741.2)$
k_{35}	$[0, 30.2424]$	$k_{35} \sim U(12.09696, 18.14544)$
k_{36}	$[0, 238710.0]$	$k_{36} \sim U(95484.0, 143226.0)$
k_{37}*	$[0, 293.824]$	$k_{37} \sim U(117.5296, 176.2944)$
k_{38}*	$[0, 25752.4]$	$k_{38} \sim U(10300.96, 15451.44)$
k_{39}*	$[0, 2.8029]$	$k_{39} \sim U(1.12116, 1.68174)$
k_{40}	$[0, 21931.2]$	$k_{40} \sim U(8772.48, 13158.72)$
k_{41}*	$[0, 54.53]$	$k_{41} \sim U(21.812, 32.718)$
k_{42}	$[0, 591980.0]$	$k_{42} \sim U(236792.0, 355188.0)$
k_{43}*	$[0, 4.4199]$	$k_{43} \sim U(1.76796, 2.65194)$
k_{44}*	$[0, 2050920.0]$	$k_{44} \sim U(820368.0, 1230552.0)$
k_{45}	$[0, 0.252658]$	$k_{45} \sim U(0.1010632, 0.1515948)$
k_{46}	$[0, 2123.42]$	$k_{46} \sim U(849.368, 1274.052)$
k_{47}	$[0, 882.574]$	$k_{47} \sim U(353.0296, 529.5444)$
k_{48}	$[0, 2.1759 \times 10^{7}]$	$k_{48} \sim U(8703600.0, 1.30554 \times 10^{7})$

Table 6.3: The range and nominal probability distributions of parameters. For unknown parameters (marked with *), we assume the their prior are uniform distributions over their ranges.

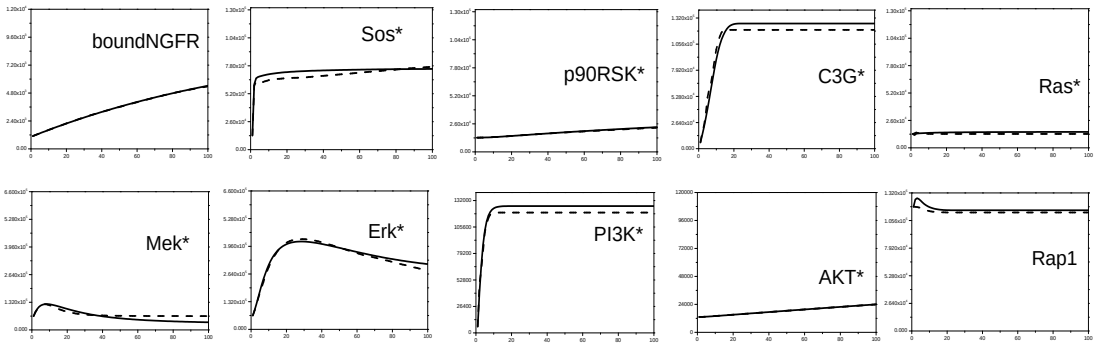


Figure 6.2: Simulation results of the EGF-NGF signaling pathway. Solid lines represent nominal profiles and dash lines represent DBN simulation profiles.

points. Our experiments show that the average values converge when the number of random trajectories generated is roughly 10^4. The averaged trajectories projected to individual protein concentration time series values are termed to be the nominal simulation profiles.

Using the implemented FF algorithm the mean of each variable over time was computed. In doing so, for the 20 parameters which were assumed to be unknown during the DBN construction process, their values were presented as specific intervals (derived from the original ODE model) in the form of evidence.

The time profiles resulting from the execution of the FF algorithm are termed to be the DBN-simulation profiles. As summarized in Figure 6.2, our DBN-simulation profiles fit the nominal simulation profiles quite well for most of the cases.

In terms of running time, a single execution of FF inference required 0.08 seconds while generating a stable nominal profile requires 105.4 seconds. Thus, the total computation time will be sharply reduced for our approach when many such "queries" need to be answered.

6.1.3 Parameter estimation

In order to test the performance of the DBN-based parameter estimation method, we synthesized experimental time series data for 9 (out of 32) proteins {bounded EGFR, bounded NGFR, active Sos, active C3G, active Akt, active p90RSK, active Erk, active Mek, active PI3K}, measured at the time points {2, 5, 10, 20, 30, 40, 50, 60, 80, 100} (min). This data was synthesized using prior knowledge about initial conditions and parameters (see Table 6.2 and Table 6.3). To mimic western blot data which is cell population based, we first averaged 10^4 random trajectories generated by sampling initial states and rate constants, and then added observation noise with variance 5% to the simulated values. With the assumed measurement precision, those values were discretized into 5 intervals, which represent the concentration levels in western blot data. We reserved the data of 7 proteins for training the parameters and reserved the rest data for testing the quality of the estimated parameter values.

With 20 of the 48 parameters having been designated during the DBN construction as being unknown, the Hooke & Jeeves algorithm was implemented to search in the discretized parameter space. The estimated parameter values in terms of maximal likelihoods of certain combination of interval values (of the 20 unknown parameters) can be found in Table 6.4. As shown in Figure 6.3, the DBN-simulation profiles generated using the estimated parameters matches the training data as shown and also has good agreement with the test data.

We compared the efficiency and quality of our results with the following ODE-based optimization algorithms: Levenberg-Marquardt (LM) (Levenberg, 2), Genetic Algorithm (GA) (Back et al., 1997), Stochastic Ranking Evolutionary Strategy (SRES) (Runarsson and Yao, 2000), and Particle Swarm Optimization (PSO) (Kennedy and Eberhart, 1995). These optimization algorithms were executed using the COPASI (Hoops et al., 2006a) tool. We scored the resulting parameters obtained from all the algorithms using the weighted sum-of-squares *difference* between the experimental data

Parameter	Range	Posterior probability distribution
k_1*	$[0, 4.37006 \times 10^{-5}]$	$k_1 \sim U(2.62204E \times 10^{-5}, 3.49605 \times 10^{-5})$
k_2*	$[0, 0.0242016]$	$k_2 \sim U(0.01452096, 0.01936128)$
k_3*	$[0, 2.76418 \times 10^{-7}]$	$k_3 \sim U(1.65851 \times 10^{-7}, 2.21134 \times 10^{-7})$
k_4*	$[0, 0.01447622]$	$k_4 \sim U(0.011580976, 0.01447622)$
k_{11}*	$[0, 64.688]$	$k_{11} \sim U(38.8128, 51.7504)$
k_{12}*	$[0, 71908.6]$	$k_{12} \sim U(28763.44, 43145.16)$
k_{15}*	$[0, 1.768192]$	$k_{15} \sim U(1.4145536, 1.7681922)$
k_{17}*	$[0, 371.518]$	$k_{17} \sim U(74.3036, 148.6072)$
k_{23}*	$[0, 19.70734]$	$k_{23} \sim U(7.882936, 11.824404)$
k_{27}*	$[0, 0.0427394]$	$k_{27} \sim U(0, 0.00854788)$
k_{28}*	$[0, 1527046]$	$k_{28} \sim U(0, 305409.2)$
k_{29}*	$[0, 21.3474]$	$k_{29} \sim U(0, 4.26948)$
k_{33}*	$[0, 0.1132558]$	$k_{33} \sim U(0.06795348, 0.09060464)$
k_{34}*	$[0, 1307902]$	$k_{34} \sim U(784741.2, 1046321.6)$
k_{37}*	$[0, 293.824]$	$k_{37} \sim U(117.5296, 176.2944)$
k_{38}*	$[0, 25752.4]$	$k_{38} \sim U(20601.92, 25752.4)$
k_{39}*	$[0, 2.8029]$	$k_{39} \sim U(2.24232, 2.8029)$
k_{41}*	$[0, 54.53]$	$k_{41} \sim U(43.624, 54.53)$
k_{43}*	$[0, 4.4199]$	$k_{43} \sim U(3.53592, 4.4199)$
k_{44}*	$[0, 2050920]$	$k_{44} \sim U(1230552, 1640736)$

Table 6.4: Parameter estimation results. The posterior distributions of unknown parameters inferred by our method.

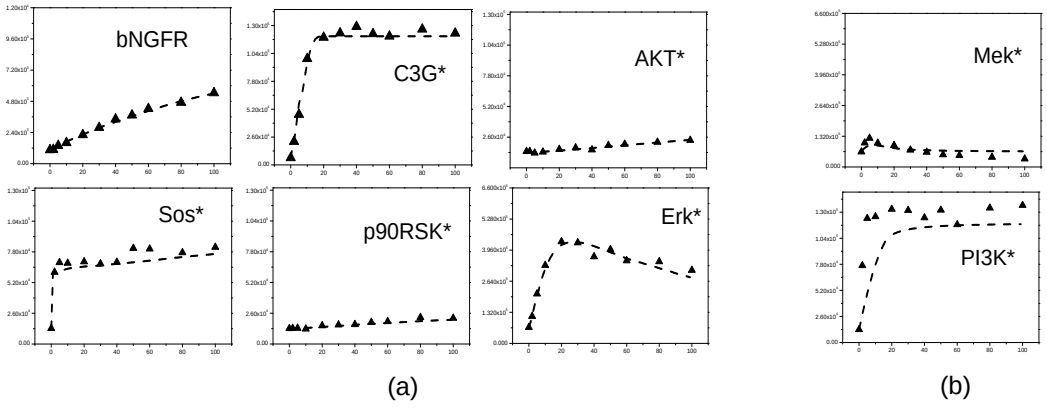


Figure 6.3: Parameter estimation results. (a) DBN-simulation profiles vs. training data. (b) DBN-simulation profiles vs. test data.

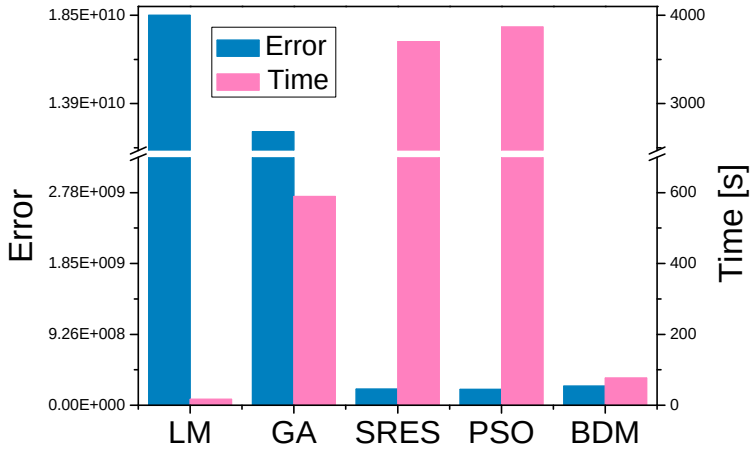


Figure 6.4: Performance comparison of our parameter estimation method (BDM) and four other methods.

and the corresponding simulation profiles (i.e. low scores correspond to low errors). The results are summarized in Figure 6.4, which suggests that our method achieves a good balance between accuracy and performance. We also note that the cost of constructing the DBN representation gets rapidly amortized.

6.1.4 Global sensitivity analysis

We modified and implemented the MPSA method for the DBN approximation setting. Using the same experimental data set introduced in previous subsection, the global sensitivities (K-S statistics) of the rate constants were computed. The results are shown in Figure 6.5. The cumulative frequency distributions for the acceptable and unacceptable cases can be found in Figure 6.6. Specifically, the reactions involved in the phosphorylation of Erk (k_{23}), Mek (k_{17}), Akt (k_{34}) and p90RSK (k_{28}) have the highest sensitivities, indicating that these reactions affect the system behavior most directly. These results are consistent with previous findings (Kholodenko, 2007; Babu et al., 2004).

The MPSA method adopts Monte Carlo strategy for the ODE model. We recorded the running time of the algorithm till the K-S values converged. The total running time of the ODE-based MPSA method was about 22 *hours*, while the MPSA method based on the DBN approximation required only 34 *minutes*. Thus the cost of constructing the DBN approximation can be easily recovered when one performs parameter estimation followed by sensitivity analysis.

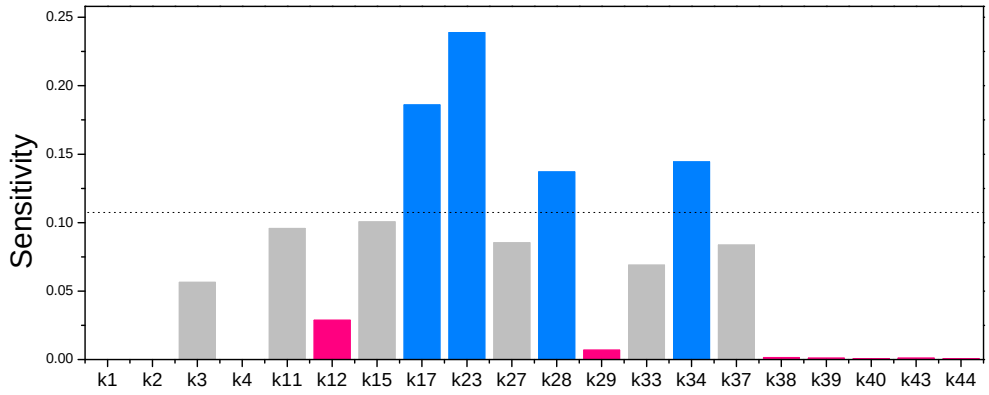


Figure 6.5: Parameter sensitivities

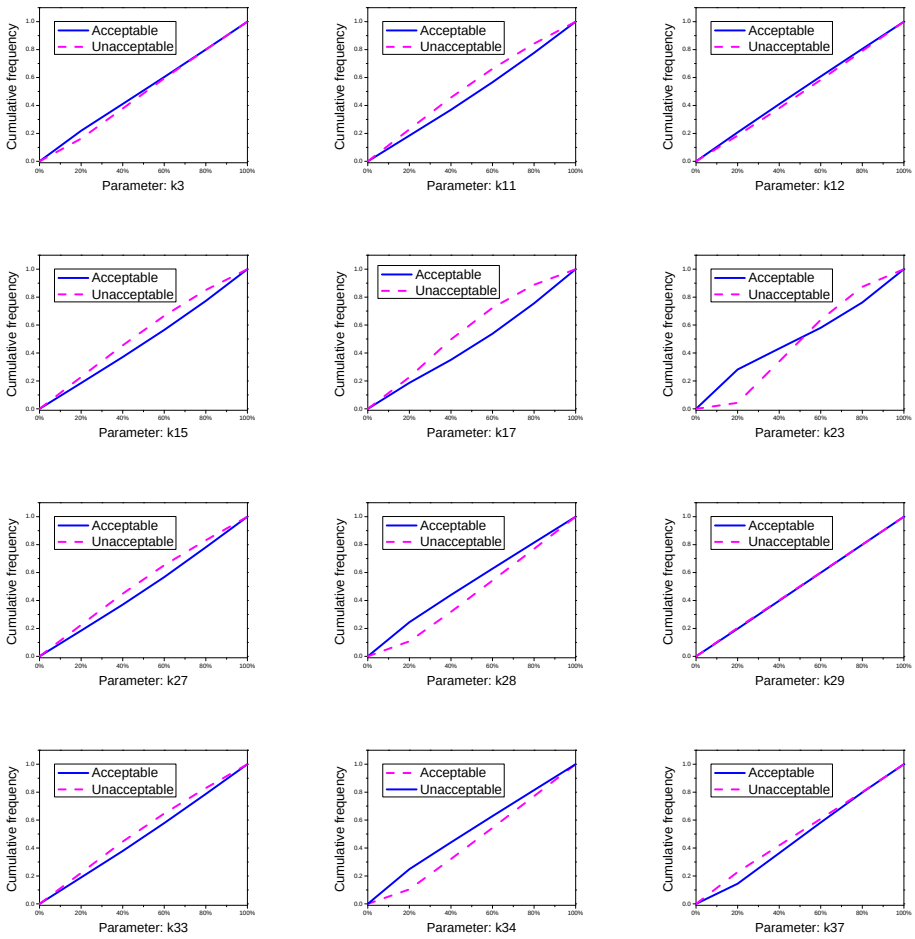


Figure 6.6: Cumulative frequency distributions of the MPSA with respect to the unknown parameters. Solid line denotes the acceptable samples and the dashed line indicates the unacceptable samples. The sensitivity of a parameter is defined as the maximum vertical difference between its two curves (K-S statistic) for the parameter.

Different discretizations

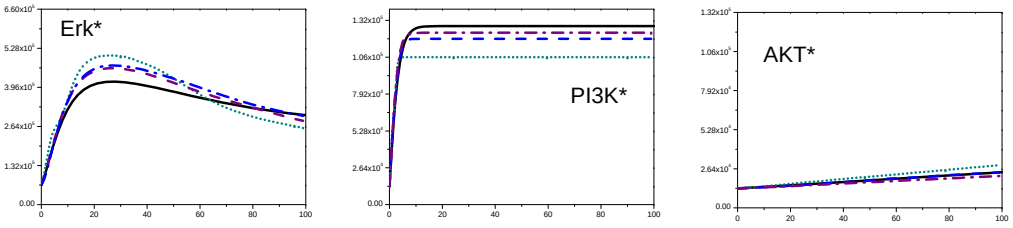


Figure 6.7: The effects of different discretizations. Solid black lines represent nominal profiles, dash-dotted purple lines present BDM profiles with $K = 8$, dashed blue lines present BDM profiles with $K = 5$, dotted cyan lines present BDM profiles with $K = 3$. (b) Accuracy and efficiency comparison of different discretizations.

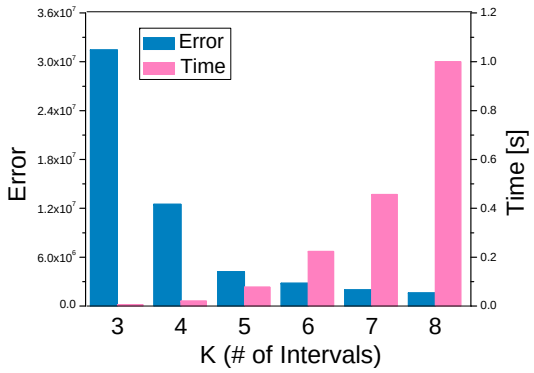


Figure 6.8: Accuracy and efficiency comparison of different discretizations.

To evaluate the effects of different discretizations, we constructed DBN approximations for the EGF-NGF pathway by fixing K intervals for each variable, with K ranging over $\{3, 4, 5, 6, 7, 8\}$. We then computed the mean of each variable over time using FF algorithm for each DBN approximation. The resulting profiles were compared with the nominal profiles. The comparison results are shown in Figure 6.7 . As might be expected, as K increases, the quality of our approximations will improve. However, since the time and space complexity of DBN based analysis depends on K, there is

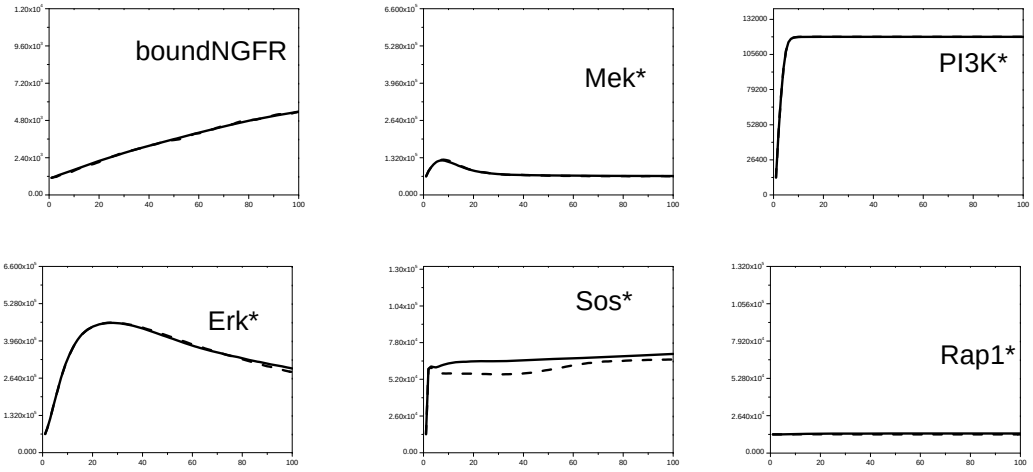


Figure 6.9: The comparison of two sampling methods. Solid lines represent direct sampling with 3 millions samples and dash lines present J-coverage sampling with $J = 1000$.

a tradeoff between efficiency and accuracy. To help decide on a good value of K, we scored the discretizations with different Ks using the weight sum-of-square difference between nominal profiles and DBN profiles, and measured the running time of a single FF inference. The results are summarized in Figure 6.8 showing that discretizations with 5 or 6 intervals might be good choices, at least in the present context.

Equation sampling

We also implemented the equation sampling method described in section 4 that provides a coverage of J samples for each possible combination of interval values of the unknown parameters in each equation. Using this method, we generated $495,000$ trajectories to get a coverage of 1000 per combination. Figure 6.9 shows the comparison of time profiles generated the using two sampling methods. The two set of profiles are nearly indistinguishable, suggesting that the equation sampling can efficiently reduce the number of samples required. This also motivated us to conduct our next case study using this method.

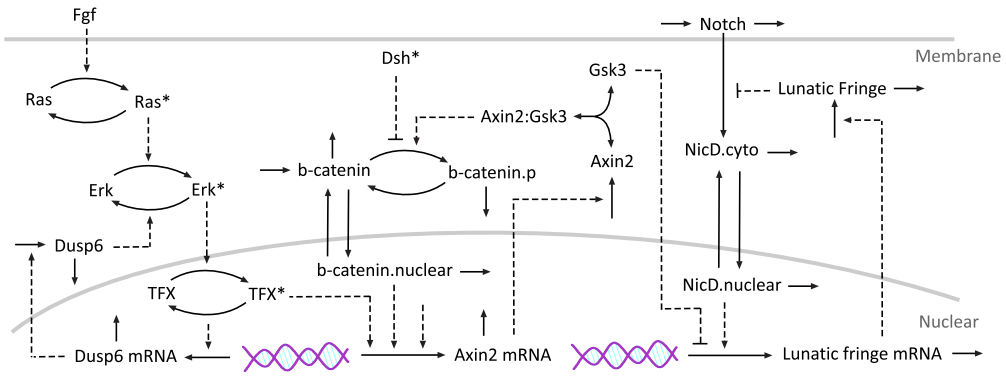


Figure 6.10: Segmentation clock pathway (Goldbeter and Pourquie, 2008)

6.2 The Segmentation Clock Network

In the developing vertebrate embryos, the segmental pattern of the spine is established when the somites are rhythmically produced. The periodic formation process of somites is governed by an oscillator called the segmentation clock, which drives the oscillatory expression of a large network of signaling genes (Dequeant et al., 2006). The underlying signaling network proposed by Goldbeter and Pourquie (2008) is shown in Figure 6.10. It couples three oscillating pathways consisting of the FGF, Wnt and Notch signaling pathways, whose periodic behaviors are produced by negative feedback loops. The corresponding ODE model can be accessed in the BioModels database (Le Novere et al., 2006). It includes 21 differential equations and 75 associated rate parameters. Again, anticipating our goal of evaluating the DBN based parameter estimation method, 39 of the 75 parameter values were singled out to be unknown. The rest of the experiments were conducted as described in our first case study.

6.2.1 Construction of the DBN approximation

We first constructed a DBN for the segmentation clock model. The graph of DBN is shown in Table 6.5. Similar to the previous case study, we discretized the ranges of each variable and parameter into 5 equal-size intervals and fixed the time step Δt to

be 5 minutes. To provide an equation of coverage of 1000 per combination, $2,585,000$ trajectories were generated up to 500 mins by sampling the prior (Table 6.6 and Table 6.7). It is worth noting that even though equation sampling was used, a much larger sample size (in relation to the first case study in the context of equation sampling) is required due to the larger number of parameters and "fatness" (i.e. the number of parameters appearing on the right hand side) of the equations involved. The construction process consumed around 3.1 hours on a cluster consisted of 10 processors.

Name	**Variable**	**Parents**
Notch protien	x_1	x_1, x_5, k_1, k_2, k_3,
cytosolic NicD	x_2	x_1, x_2, x_3, x_5, k_4, k_5,
nuclear NicD	x_3	x_2, x_3, k_4, k_5, k_6, k_7,
Lunatic fringe mRNA	x_4	x_3, x_4, x_{21}, k_8, k_{11},
Lunatic Fringe protien	x_5	x_4, x_5, k_{12}, k_{13}, k_{14},
phosph. beta-catenin	x_6	x_6, x_{10}, x_{20}, k_{15},
nuclear beta-catenin	x_7	x_7, x_{10}, k_{22}, k_{23},
Axin2 protien	x_8	x_8, x_9, x_{11}, x_{20}, k_{19}, k_{20},
Gsk3	x_9	x_8, x_9, x_{20},
beta-catenin	x_{10}	x_6, x_7, x_{10}, x_{20}, k_{22}, k_{23},
Axin2 mRNA	x_{11}	x_7, x_{11}, x_{14}, k_{16}, k_{17}, k_{18},
active Ras	x_{12}	x_{12}, x_{17}, k_{28}, k_{29}, k_{35}, k_{36},
active ERK	x_{13}	x_{12}, x_{13}, x_{16}, x_{18}, k_{27}, k_{37},
active TF X	x_{14}	x_{13}, x_{14}, x_{19}, k_{38}, k_{39},
Dusp6 mRNA	x_{15}	x_{14}, x_{15}, k_{31}, k_{32}, k_{33}, k_{34},
Dusp6 protien	x_{16}	x_{15}, x_{16}, k_{24}, k_{25}, k_{26},
inactive Ras	x_{17}	x_{12},
inactive ERK	x_{18}	x_{13},
inactive TF X	x_{19}	x_{14},
Axin2/Gsk3 destruction complex	x_{20}	x_9,
vsFK	x_{21}	x_9, k_9, k_{10},

Table 6.5: The DBN structure of the segmentation clock pathway model. (Known parameters are not shown in the parent sets)

6.2.2 Probabilistic inference

To generate stable nominal profiles, we averaged 10^4 trajectories according to the prior. The nominal profiles were then compared with the DBN-simulation profiles computed from the FF inference results. The comparison results are shown in Figure 6.11, which

Probability distribution
$x_1 \sim U(0.16, 0.2)$
$x_2 \sim U(0.0, 0.4)$
$x_3 \sim U(0.0, 0.02)$
$x_4 \sim U(0.0, 0.8)$
$x_5 \sim U(0.0, 0.8)$
$x_6 \sim U(0.08, 0.1)$
$x_7 \sim U(0.0, 0.2)$
$x_8 \sim U(0.0, 0.2)$
$x_9 \sim U(2.56, 3.2)$
$x_{10} \sim U(0.0, 0.4)$
$x_{11} \sim U(0.0, 3.2)$
$x_{12} \sim U(0.44, 0.88)$
$x_{13} \sim U(0.0, 0.44)$
$x_{14} \sim U(0.0, 0.44)$
$x_{15} \sim U(0.0, 1.4)$
$x_{16} \sim U(0.0, 2.4)$
$x_{17} \sim U(0.0, 0.4)$
$x_{18} \sim U(0.0, 0.4)$
$x_{19} \sim U(0.0, 0.4)$
$x_{20} \sim U(0.0, 0.6)$
$x_{21} \sim U(0.0, 0.6)$

Table 6.6: Prior (initial) probability distribution of variables

Parameter	Range	Nominal probability distribution
k_1*	$[1.26, 1.54]$	$k_1 \sim U(1.372, 1.428)$
k_2*	$[0.207, 0.253]$	$k_2 \sim U(0.2254, 0.2346)$
k_3*	$[2.538, 3.102]$	$k_3 \sim U(2.7636, 2.8764)$
k_4*	$[0.09, 0.11]$	$k_4 \sim U(0.098, 0.102)$
k_5*	$[0.09, 0.11]$	$k_5 \sim U(0.098, 0.102)$
k_6*	$[0.0009, 0.0011]$	$k_6 \sim U(0.00098, 0.00102)$
k_7*	$[0.09, 0.11]$	$k_7 \sim U(0.098, 0.102)$
k_8*	$[0.6912, 0.8448]$	$k_8 \sim U(0.75264, 0.78336)$
k_9*	$[2.25, 2.75]$	$k_9 \sim U(2.45, 2.55)$
k_{10}*	$[2.7, 3.3]$	$k_{10} \sim U(2.94, 3.06)$
k_{11}*	$[1.728, 2.112]$	$k_{11} \sim U(1.8816, 1.9584)$
k_{12}*	$[0.333, 0.407]$	$k_{12} \sim U(0.3626, 0.3774)$
k_{13}*	$[0.351, 0.429]$	$k_{13} \sim U(0.3822, 0.3978)$
k_{14}*	$[0.27, 0.33]$	$k_{14} \sim U(0.294, 0.306)$
k_{15}*	$[6.3558, 7.7682]$	$k_{15} \sim U(6.92076, 7.20324)$
k_{16}*	$[1.476, 1.804]$	$k_{16} \sim U(1.6072, 1.6728)$
k_{17}*	$[0.63, 0.77]$	$k_{17} \sim U(0.686, 0.714)$
k_{18}*	$[0.45, 0.55]$	$k_{18} \sim U(0.49, 0.51)$
k_{19}*	$[0.018, 0.022]$	$k_{19} \sim U(0.0196, 0.0204)$
k_{20}*	$[0.54, 0.66]$	$k_{20} \sim U(0.588, 0.612)$
k_{21}*	$[0.567, 0.693]$	$k_{21} \sim U(0.6174, 0.6426)$
k_{22}*	$[0.63, 0.77]$	$k_{22} \sim U(0.686, 0.714)$
k_{23}*	$[1.35, 1.65]$	$k_{23} \sim U(1.47, 1.53)$
k_{24}*	$[0.45, 0.55]$	$k_{24} \sim U(0.49, 0.51)$
k_{25}*	$[1.8, 2.2]$	$k_{25} \sim U(1.96, 2.04)$
k_{26}*	$[0.45, 0.55]$	$k_{26} \sim U(0.49, 0.51)$
k_{27}*	$[1.215, 1.485]$	$k_{27} \sim U(1.323, 1.377)$
k_{28}*	$[0.45, 0.55]$	$k_{28} \sim U(0.49, 0.51)$
k_{29}*	$[0.0927, 0.1133]$	$k_{29} \sim U(0.10094, 0.10506)$
k_{30}*	$[0.09, 0.11]$	$k_{30} \sim U(0.098, 0.102)$
k_{31}*	$[0.45, 0.55]$	$k_{31} \sim U(0.49, 0.51)$
k_{32}*	$[0.45, 0.55]$	$k_{32} \sim U(0.49, 0.51)$
k_{33}*	$[0.81, 0.99]$	$k_{33} \sim U(0.882, 0.918)$
k_{34}*	$[0.45, 0.55]$	$k_{34} \sim U(0.49, 0.51)$
k_{35}*	$[4.4712, 5.4648]$	$k_{35} \sim U(4.86864, 5.06736)$
k_{36}*	$[0.369, 0.451]$	$k_{36} \sim U(0.4018, 0.4182)$
k_{37}*	$[2.97, 3.63]$	$k_{37} \sim U(3.234, 3.366)$
k_{38}*	$[1.44, 1.76]$	$k_{38} \sim U(1.568, 1.632)$
k_{39}*	$[0.45, 0.55]$	$k_{39} \sim U(0.49, 0.51)$

Table 6.7: The range and nominal probability distributions of unknown parameters.

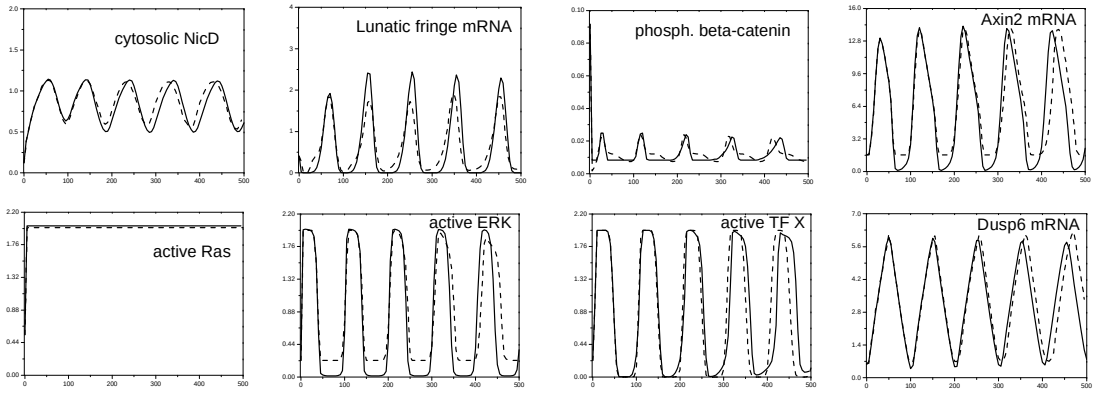


Figure 6.11: Simulation results of segmentation clock pathway. Solid lines represent nominal profiles and dash lines represent DBN-simulation profiles.

shows a good fit between them. In terms of running time, a single execution of FF inference required 0.01 seconds while generating a stable nominal profile took 407.3 seconds.

6.2.3 Parameter estimation

We next tested the performance of the DBN-based parameter estimation method. We synthesized population based experimental time series data for 8 (out of 22) proteins {Notch protein, nuclear NicD, Lunatic fringe mRNA, Axin2 mRNA, active ERK, Dusp6 mRNA, Dusp6 protein, cytosolic NicD}, measured at the time points {400, 410, 420, 430, 440, 450, 460, 470, 480, 490} (min) based on the prior knowledge about initial conditions and parameters (see Tables 6.6 and Table 6.7). We averaged 10^4 random trajectories generated by sampling initial states and rate constants, and then added observation noise with variance 5% to the simulated values. The data of 6 proteins were reserved for training the parameters and the rest data were used for testing the quality of the estimated parameter values.

For the 40 parameters which had been designated to be unknown, the DBN based implementation of Hooke & Jeeves algorithm was applied. The results can be found in

Table 6.7. As shown in Figure 6.12 (a) and (b), the DBN-simulation profiles generated using the estimated parameters obtained (with the match to training data as shown) has good agreement with the test data.

We then compared the efficiency and quality of our results with the ODE-based optimization algorithms: LM, GA, SRES, and PSO introduced in previous case study. The results are summarized in Figure 6.12 (c) suggesting again that our method achieves a good balance between accuracy and performance and the cost of constructing the DBN representation gets rapidly amortized.

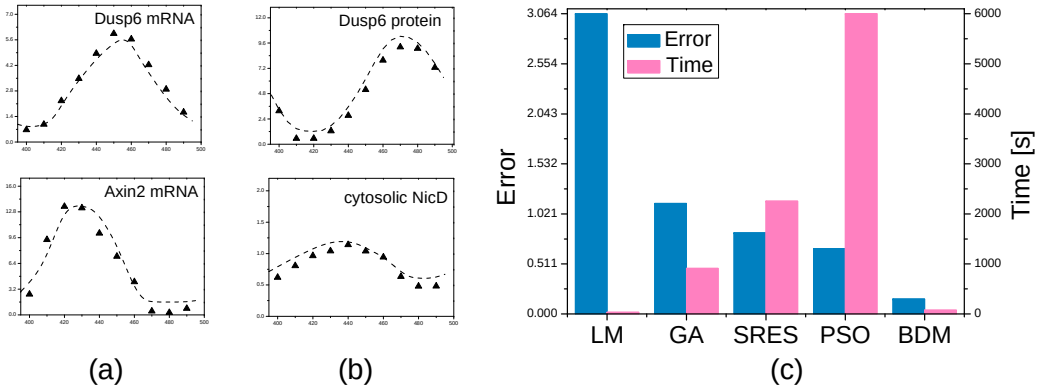


Figure 6.12: Parameter estimation results. (a) DBN-simulation profiles vs. training data. (b) DBN-simulation profiles vs. test data. (c) Performance comparison of our parameter estimation method (BDM) and 4 other methods.

6.2.4 Global sensitivity analysis

The global sensitivities of the parameters were computed using DBN based MPSA method and are shown in Figure 6.13. Specifically, the reactions involved in the degradation of Dusp6 mRNA (k_{34}), the transcription of Dusp6 gene induced by TF X (k_{33}) and the transcription of the Axin2 gene induced by factor TF X (k_{18}) have the highest sensitivities, indicating that these reactions affect the system behavior most directly. Since all these reactions are present in the FGF pathway, we hypothesize that FGF pathway is the key regulatory mechanism that drives and synchronizes the oscillatory

gene expression of segmentation clock.

The total running time of the ODE-based MPSA method was about 81.25 *hours*, while the MPSA method based on the DBN required only 3.25 *hours*.

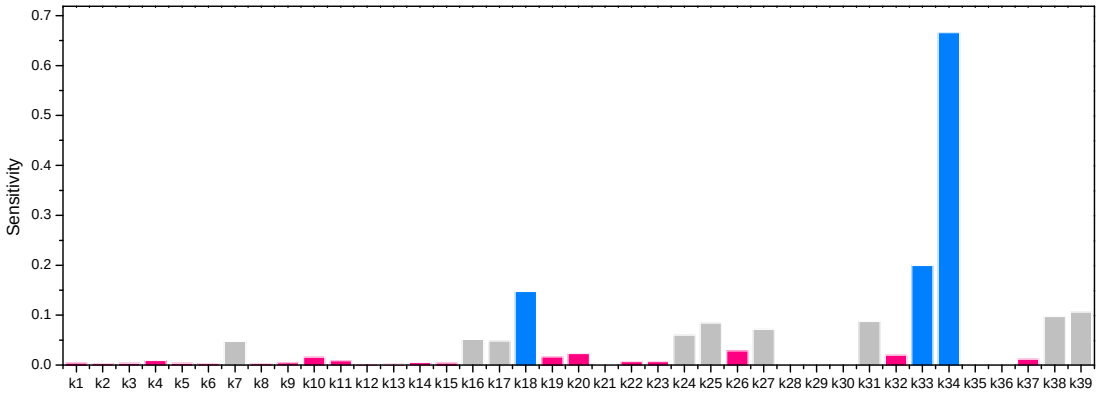


Figure 6.13: Parameter sensitivities

6.3 The Complement System

The complement system is pivotal to defending against invading microorganisms. The complement proteins recognize conserved pathogen-associated molecular patterns (PAMPs) on the surface of the invading pathogens (Walport, 2001a) to initiate the innate immunity response. The complement proteins in the blood normally circulate as inactive zymogens. Upon stimulation, proteases in the system cleave the zymogens to release active fragments and initiate an amplifying cascade of further cleavages. The complement system constitutes over 30 proteins including serum proteins and cell membrane receptors. There are three major complement activation routes: the classical, the lectin and the alternative pathways (Walport, 2001b). Regardless of how these pathways are initiated, the complement activity leads to proteolytic activation and deposition of the major complement proteins C4 and C3, which induces phagocytosis, and the subsequent assembly of the membrane attack complex which lyses the invading microbes. In the

process, potent chemoattractant anaphylatoxins are released. However, complement is a double-edged sword; adequate complement activation is necessary for killing the bacteria and removing the apoptotic cells, while excessive complement activation can harm the host by generating inflammation and exacerbating tissue injury. Dysregulation of the balance between complement activation and inhibition can lead to rheumatoid arthritis (Okroj et al., 2007), systemic lupus erythematosus (Truedsson et al., 2007), Alzheimer's disease (Veerhuis et al., 2005) and age-related macular degeneration (Anderson et al., 2010). Since the final outcome of complement related diseases may be attributable to the imbalance between activation and inhibition, which is induced by inappropriate initiation of the cascade or deficiencies in specific regulators (Sjoberg et al., 2009), manipulation of this balance using drugs represents an interesting therapeutic opportunity awaiting further investigation. In light of this potential, complement inhibitors such as factor H and C4b-binding protein (C4BP) are critical since they play important roles in tightly controlling the proteolytic cascade of complement and avoiding excessive activation. Therefore, a systems-level understanding of the complement activation and inhibition, as well as the roles of complement inhibitors, will contribute towards the development of complement-based immunomodulation therapies.

As the frontline of host defense, complement is usually initiated by the interaction of several pattern-recognition receptors with the surface of pathogens. C-reactive protein (CRP), which is an acute phase reactant (Mold et al., 1981) and ficolins are two initiators of the classical and lectin pathways, which boost immune responses by recognizing phosphorylcholine (PC) or N-acetylglucosamine (GlcNAc), respectively, displayed on the surface of invading bacteria (Marnell et al., 2005; Fujita et al., 2004; Ng et al., 2007). Recently, it was discovered that under local infection-inflammation conditions as reflected by pH and calcium levels, the conformations of CRP and L-ficolin change which leads to a strong interaction between them (Zhang et al., 2009). This interaction triggers crosstalk between classical and lectin pathways and induces new amplification

mechanisms, which in turn reinforces the overall antibacterial activity and bacterial clearance.

On the other hand, C4BP, a major complement inhibitor, is a large glycoprotein synthesized and secreted by the liver. The estimated plasma concentration of C4BP is 260 nM under normal physiological condition (Griffin et al., 1992). However, as an acute phase reactant, its plasma level can be elevated up to four-fold during inflammation (Barnum and Dahlback, 1990; Boerger et al., 1987). Through its α-chain (Blom et al., 2001, 1999), C4BP modulates complement pathways by controlling C4b-mediated reactions in multiple ways: First, C4BP acts as a cofactor to factor I, in the proteolytic inactivation of C4b, which prevents the formation and reconstitution of the classical C3-convertase (C4bC2a) (Scharfstein et al., 1978). Second, C4BP prevents the assembly of the classical C3 convertase by binding to nascent C4b, and accelerates the natural decay of the C4bC2a complex (Gigli, 1979). Third, C4BP can compete with C1q for the immobilized CRP (Sjoberg et al., 2006). Further, C4BP has been proposed as a therapeutic agent for complement-related autoimmune diseases on the premise that mice models supplemented with human C4BP showed attenuation in the progression of arthritis (Blom et al., 2009). Therefore, it is important to understand the systemic effect and the underlying inhibitory mechanism of C4BP.

With this background, we carried out a combined computational and experimental study and obtained the following results.

6.3.1 Construction of the ODE model

A schematic representation of the complement system is shown in Figure 6.14. The reaction network diagram of our model is shown in Figure 6.15. Processes such as protein association, degradation and translocation are modeled with mass action kinetics and processes such as cleavage, activation and inhibition with Michaelis-Menten kinetics. The resulting ODE model was implemented using the open source software COPASI

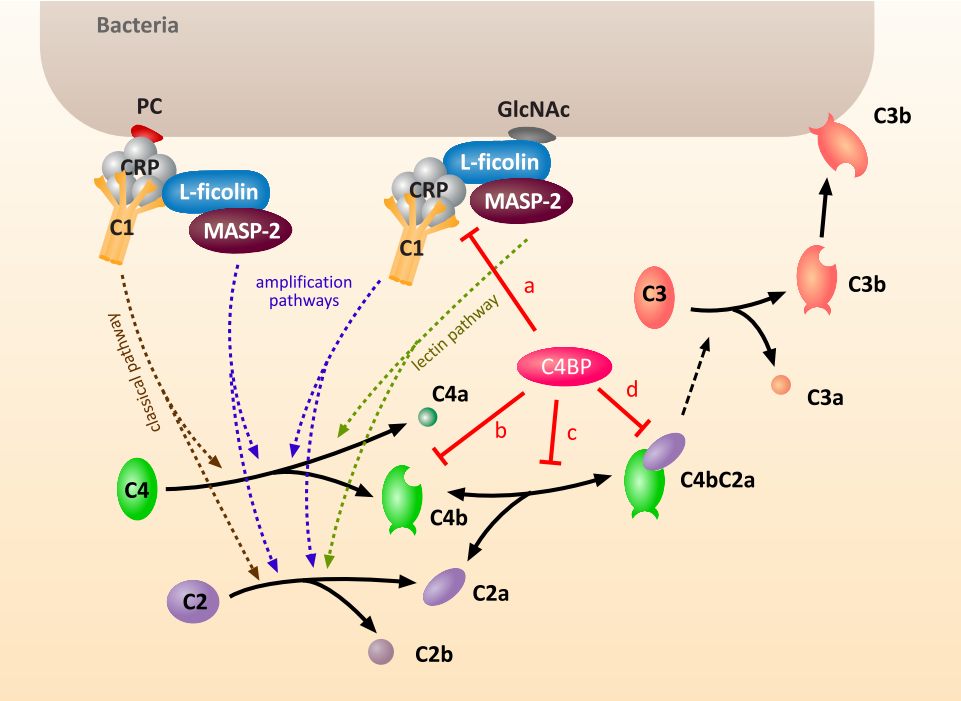


Figure 6.14: Simplified schematic representation of the complement system. The complement cascade is triggered when CRP or L-ficolin is recruited to the bacterial surface by binding to ligand PC (classical pathway) or GlcNAc (lectin pathway). Under inflammation condition, CRP and ficolin interact with each other and induce amplification pathways. The activated CRP and L-ficolin on the surface interacts with C1 and MASP-2 respectively and leads to the formation of the C3 convertase (C4bC2a), which cleaves C3 to C3b and C3a. Deposition of C3b initiates the opsonization, phagocytosis, and lysis. C4BP regulates the activation of complement pathways by: (a) binding to CRP, (b) accelerating the decay of the C4bC2a, (c) binding to C4b, and (d) preventing the assembly of C4bC2a (red bars). Solid arrows and dotted arrows indicate protein conversions and enzymatic reactions, respectively.

(Hoops et al., 2006b). It consists of 42 species, 45 reactions and 85 kinetic parameters with 71 unknown. The details can be found in Liu et al. (2010).

6.3.2 Construction of the DBN approximation

We next constructed the DBN approximations of the ODE model to carry out parameter estimation and global sensitivity analysis.

In the ODE model the PC-initiated and GlcNAc-initiated complement cascades were merged for convenience. By suppressing these two cascades one at a time (by setting the corresponding expressions in the reaction equations to zero), we constructed two DBNs; one for the PC-initiated complement cascade (Table 6.8) and the other for GlcNAc-initiated complement cascade (Table 6.9). The range of each variable and parameter was discretized into 6 non-equal size intervals and 5 equal size intervals, respectively. The time points of interest were set to $\{0, 100, 200, \ldots, 12600\}$ (seconds). Here 12600 seconds is equivalent to 3.5 hours, which is the largest time point of our training experimental data. We then employed the equation sampling method (Section 4.3.2) with a coverage of 1000 to construct the two DBNs. Each of the resulting DBN approximations encoded 1.2×10^6 trajectories generated by sampling the initial values of the variables and the parameters from the prior, which was assumed to be uniform distributions over certain intervals (Liu et al., 2010). The computational workload was distributed on 20 processors in a cluster and the running time was around 12 h.

6.3.3 Parameter estimation

The values of initial concentrations and 14 kinetic parameters were obtained from literature data (Table 6.10 and Table 6.11). To estimate the remaining 71 kinetic parameters, we generated training data by incubating human blood under normal and infection-inflammation conditions with beads coated with PC or GlcNAc followed by immunodetection of the deposited CRP, C4, C3 and C4BP in time series. For PC-

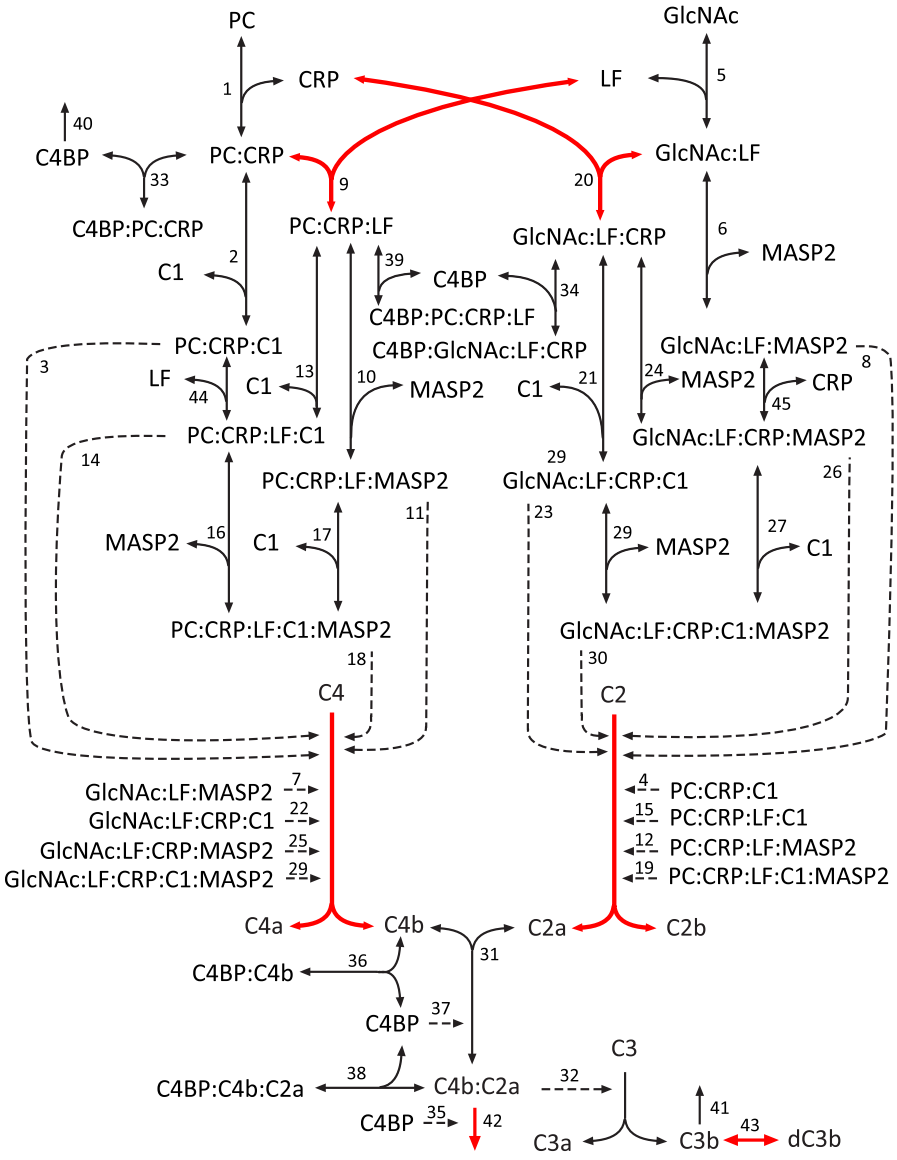


Figure 6.15: The reaction network diagram of the mathematical model. Complexes are denoted by the names of their components, separated by a ":". Single-headed solid arrows characterize irreversible reactions and double-headed arrows characterize reversible reactions. Dotted arrows represent enzymatic reactions. The kinetic equations of individual reactions are presented in the supplementary material. The reactions with high global sensitivities are labeled in red.

Name	Variable	Parents
CRP	$px1$	$px1$, $px2$, $px3$, $k1$, $k2$,
PC	$px2$	$px1$, $px2$, $px3$, $k1$, $k2$,
PC/CRP	$px3$	$px3$, $pt9$, $pt10$, $pt11$, $pt12$,
C4	$px4$	$px4$, $pt1$, $pt2$, $pt3$, $pt4$,
C4a	$px5$	$px4$, $px5$,
C4b	$px6$	$px6$, $pt1$, $pt3$, $pt4$, $pt34$, $pa1$,
C2	$px7$	$px7$, $pt5$, $pt6$, $pt7$, $pt8$,
C1	$px8$	$px8$, $px19$, $pt15$, $pt16$, $k66$,
PC/CRP/C1	$px9$	$px3$, $px8$, $px9$, $k3$, $k4$,
C2a	$px10$	$px10$, $pt5$, $pt13$, $pt14$, $pa2$,
C2b	$px11$	$px7$, $px11$,
C4b/C2a	$px12$	$px12$, $pt17$, $pt18$, $pt19$, $pt20$,
C3	$px13$	$px12$, $px13$, $k9$,
C3a	$px14$	$px13$, $px14$,
C3b	$px15$	$px12$, $px13$, $px15$, $k9$, $k89$,
MASP	$px16$	$px16$, $px18$, $px24$, $pt30$, $k16$, $k64$,
LF	$px17$	$px3$, $px17$, $px18$, $k28$, $k29$,
PC/CRP/LF	$px18$	$px18$, $pt21$, $pt22$, $pt23$, $pt24$,
PC/CRP/LF/MASP	$px19$	$px8$, $px19$, $pt31$, $k15$, $k66$,
C4BP	$px20$	$px20$, $pt25$, $pt26$, $pt27$, $pt28$, $k88$,
C4BP/PC/CRP	$px21$	$px3$, $px20$, $px21$, $k40$, $k41$,
C4BP/C4b	$px22$	$px6$, $px20$, $px22$, $k45$, $k46$,
C4b/C2a/C4BP	$px23$	$px12$, $px20$, $px23$, $k48$, $k49$,
PC/CRP/LF/C1	$px24$	$px16$, $px24$, $pt32$, $k53$, $k64$,
C4BP/PC/CRP/LF	$px25$	$px18$, $px20$, $px25$, $k91$, $k92$,
PC/CRP/LF/C1/MASP	$px26$	$px26$, $pt33$, $k65$, $k67$,
$TmpVar_{t1}$	$pt1$	$px4$, $px9$, $k5$, $k13$,
$TmpVar_{t2}$	$pt2$	$px4$, $px19$, $k17$, $k18$,
$TmpVar_{t3}$	$pt3$	$px4$, $px24$, $k54$, $k55$,
$TmpVar_{t4}$	$pt4$	$px4$, $px26$, $k68$, $k69$,
$TmpVar_{t5}$	$pt5$	$px7$, $px9$, $k6$, $k14$,
$TmpVar_{t6}$	$pt6$	$px7$, $px19$, $k30$, $k31$,
$TmpVar_{t7}$	$pt7$	$px7$, $px24$, $k56$, $k57$,
$TmpVar_{t8}$	$pt8$	$px7$, $px26$, $k70$, $k71$,
$TmpVar_{t9}$	$pt9$	$px1$, $px2$, $px9$, $k1$, $k4$,
$TmpVar_{t10}$	$pt10$	$px18$, $px21$, $k29$, $k41$,
$TmpVar_{t11}$	$pt11$	$px3$, $px8$, $k2$, $k3$,
$TmpVar_{t12}$	$pt12$	$px3$, $px17$, $px20$, $k28$, $k40$,
$TmpVar_{t13}$	$pt13$	$px12$, $px20$, $k8$, $k47$,
$TmpVar_{t14}$	$pt14$	$px6$, $px10$, $k7$,
$TmpVar_{t15}$	$pt15$	$px3$, $px8$, $px18$, $k3$, $k52$,
$TmpVar_{t16}$	$pt16$	$px9$, $px24$, $px26$, $k4$, $k53$, $k67$,
$TmpVar_{t17}$	$pt17$	$px2$, $px6$, $px10$, $k7$, $k49$,
$TmpVar_{t18}$	$pt18$	$px12$, $px20$, $k47$,
$TmpVar_{t19}$	$pt19$	$px12$, $px20$, $k8$, $k44$,
$TmpVar_{t20}$	$pt20$	$px12$, $px20$, $k48$, $k90$,
$TmpVar_{t21}$	$pt21$	$px16$, $px18$, $k16$, $k29$,
$TmpVar_{t22}$	$pt22$	$px3$, $px17$, $px19$, $k15$, $k28$,
$TmpVar_{t23}$	$pt23$	$px8$, $px18$, $px20$, $k52$, $k91$,
$TmpVar_{t24}$	$pt24$	$px24$, $px25$, $k53$, $k92$,
$TmpVar_{t25}$	$pt25$	$px3$, $px6$, $px20$, $k40$, $k46$,
$TmpVar_{t26}$	$pt26$	$px21$, $px22$, $k41$, $k45$,
$TmpVar_{t27}$	$pt27$	$px12$, $px18$, $px20$, $k48$, $k91$,
$TmpVar_{t28}$	$pt28$	$px23$, $px25$, $k49$, $k92$,
$TmpVar_{t29}$	$pt29$	$px12$, $px22$, $k8$, $k45$,
$TmpVar_{t30}$	$pt30$	$px19$, $px26$, $k15$, $k65$,
$TmpVar_{t31}$	$pt31$	$px16$, $px18$, $px26$, $k16$, $k67$,
$TmpVar_{t32}$	$pt32$	$px8$, $px18$, $px26$, $k52$, $k65$,
$TmpVar_{t33}$	$pt33$	$px8$, $px16$, $px19$, $px24$, $k64$, $k66$,
$TmpVar_{t34}$	$pt34$	$px6$, $px10$, $px20$, $k7$, $k46$,
$TmpVar_{a1}$	$pa1$	$pt2$, $pt18$, $pt29$,
$TmpVar_{a2}$	$pa2$	$pt6$, $pt7$, $pt8$,

Table 6.8: The structure of DBN approximation of PC-initiated classical complement pathway

Name	Variable	Parents
CRP	$gx1$	$gx1$, $gx15$, $gx18$, $k32$, $k33$,
C4	$gx2$	$gx2$, $gt1$, $gt2$, $gt3$, $gt4$,
C4a	$gx3$	$gx2$, $gx3$,
C4b	$gx4$	$gx4$, $gt3$, $gt4$, $gt9$, $gt10$, $ga1$,
C2	$gx5$	$gx5$, $gt5$, $gt6$, $gt7$, $gt8$,
C1	$gx6$	$gx6$, $gt11$, $gt12$,
C2a	$gx7$	$gx4$, $gx7$, $gt8$, $gt13$, $ga2$, $k7$,
C2b	$gx8$	$gx8$, $gt5$, $gt6$, $gt7$, $gt8$,
C4b/C2a	$gx9$	$gx9$, $gt14$, $gt15$, $gt16$, $k8$,
C3	$gx10$	$gx9$, $gx10$, $k9$,
C3a	$gx11$	$gx9$, $gx10$, $gx11$, $k9$,
C3b	$gx12$	$gx9$, $gx10$, $gx12$, $k9$, $k89$,
MASP	$gx13$	$gx13$, $gx15$, $gt17$, $gt18$, $k21$,
GlcNAc	$gx14$	$gx14$, $gx15$, $gx16$, $k19$, $k20$,
GlcNAc/LF	$gx15$	$gx15$, $gt19$, $gt20$, $gt21$, $k20$,
LF	$gx16$	$gx14$, $gx15$, $gx16$, $k19$, $k20$,
GlcNAc/LF/MASP	$gx17$	$gx13$, $gx15$, $gx17$, $k21$, $k22$,
GlcNAc/LF/CRP	$gx18$	$gx18$, $gt22$, $gt23$, $gt24$, $gt25$,
GlcNAc/LF/CRP/C1	$gx19$	$gx13$, $gx19$, $gt26$, $k1$, $k35$,
C4BP	$gx20$	$gx9$, $gx20$, $gt27$, $gt28$, $k48$, $k88$,
C4BP/GlcNAc/LF/CRP	$gx21$	$gx18$, $gx20$, $gx21$, $k42$, $k43$,
C4BP/C4b	$gx22$	$gx4$, $gx20$, $gx22$, $k45$, $k46$,
C4b/C2a/C4BP	$gx23$	$gx9$, $gx20$, $gx23$, $k48$, $k49$,
GlcNAc/LF/CRP/MASP	$gx24$	$gx6$, $gx24$, $gt29$, $k59$, $k66$,
GlcNAc/LF/CRP/C1/MASP	$gx25$	$gx25$, $gt30$, $k2$, $k67$,
$TmpVar_{t1}$	$gt1$	$gx2$, $gx17$, $k23$, $k24$,
$TmpVar_{t2}$	$gt2$	$gx2$, $gx19$, $k36$, $k37$,
$TmpVar_{t3}$	$gt3$	$gx2$, $gx24$, $k60$, $k61$,
$TmpVar_{t4}$	$gt4$	$gx2$, $gx25$, $k84$, $k85$,
$TmpVar_{t5}$	$gt5$	$gx5$, $gx17$, $k25$, $k26$,
$TmpVar_{t6}$	$gt6$	$gx5$, $gx19$, $k38$, $k39$,
$TmpVar_{t7}$	$gt7$	$gx5$, $gx24$, $k62$, $k63$,
$TmpVar_{t8}$	$gt8$	$gx5$, $gx25$, $k86$, $k87$,
$TmpVar_{t9}$	$gt9$	$gx9$, $gx22$, $k8$, $k45$,
$TmpVar_{t10}$	$gt10$	$gx4$, $gx7$, $gx20$, $k7$, $k46$,
$TmpVar_{t11}$	$gt11$	$gx19$, $gx25$, $k35$, $k67$,
$TmpVar_{t12}$	$gt12$	$gx6$, $gx18$, $gx24$, $k34$, $k66$,
$TmpVar_{t13}$	$gt13$	$gx9$, $gx20$, $k8$, $k47$,
$TmpVar_{t14}$	$gt14$	$gx4$, $gx7$, $gx23$, $k7$, $k49$,
$TmpVar_{t15}$	$gt15$	$gx9$, $gx20$, $k44$, $k47$,
$TmpVar_{t16}$	$gt16$	$gx9$, $gx20$, $k48$, $k90$,
$TmpVar_{t17}$	$gt17$	$gx17$, $gx24$, $gx25$, $k2$, $k22$, $k59$,
$TmpVar_{t18}$	$gt18$	$gx13$, $gx18$, $gx19$, $k1$, $k58$,
$TmpVar_{t19}$	$gt19$	$gx14$, $gx16$, $k19$,
$TmpVar_{t20}$	$gt20$	$gx17$, $gx18$, $k22$, $k33$,
$TmpVar_{t21}$	$gt21$	$gx1$, $gx13$, $gx15$, $k21$, $k32$,
$TmpVar_{t22}$	$gt22$	$gx1$, $gx15$, $gx19$, $k32$, $k35$,
$TmpVar_{t23}$	$gt23$	$gx21$, $gx24$, $k43$, $k59$,
$TmpVar_{t24}$	$gt24$	$gx13$, $gx18$, $gx20$, $k42$, $k58$,
$TmpVar_{t25}$	$gt25$	$gx6$, $gx18$, $k33$, $k34$,
$TmpVar_{t26}$	$gt26$	$gx6$, $gx18$, $gx25$, $k2$, $k34$,
$TmpVar_{t27}$	$gt27$	$gx21$, $gx22$, $gx23$, $k43$, $k45$, $k49$,
$TmpVar_{t28}$	$gt28$	$gx4$, $gx18$, $gx20$, $k42$, $k46$,
$TmpVar_{t29}$	$gt29$	$gx13$, $gx18$, $gx25$, $k58$, $k67$,
$TmpVar_{t30}$	$gt30$	$gx6$, $gx13$, $gx19$, $gx24$, $k1$, $k66$,
$TmpVar_{a1}$	$ga1$	$gx9$, $gx20$, $gt1$, $gt2$, $k47$,
$TmpVar_{a2}$	$ga2$	$gt5$, $gt6$, $gt7$,

Table 6.9: The structure of DBN approximation of GlcNAc-initiated classical complement pathway

beads, the concentration levels of deposited CRP, C4, C3 and C4BP were measured at 8 time points from 0 to 3.5 hours (Figure 6.16:A-B, red dots). For GlcNAc-beads, the concentration levels of deposited MASP-2, C4, C3 and C4BP were also measured at 8 time points from 0 to 3.5 hours (Figure 6.16:C-D, red dots).

We then deployed the two-stage DBN based method to estimate unknown kinetic parameters. As mentioned above, each unknown parameter's value space was divided into 5 equal intervals. In the first stage, we used stochastic ranking evolutionary strategy (SRES) to search in the discretized parameter space consisting of 5^{71} combinations of interval values of the unknown parameters. The SRES search was done using a modified version of the tool libSRES (Ji and Xu, 2006) (The modification enables one to perform search in a discrete solution space). The result of this first stage was a maximum likelihood estimate of a combination of intervals of parameter values.

In the first and the second case studies, we stopped with the first stage and worked with this combination of intervals of parameter values. However, in this case study we wanted to use the ODE model too for conducting *in silico* experiments such as varying initial concentrations including the down and over expression of C4BP. This would have been difficult to achieve by working solely with our current DBN approximation.

Thus, we then proceeded the second the stage and searched within this combination of intervals having maximal likelihood. Consequently, the size of the search space for the second stage was just $1/5^{71}$ of the original search space. We performed the standard SRES algorithm using libSRES tool to search for the vector of parameter values with minimum objective value. The parameter values thus estimated are shown in Table 6.11.

Figure 6.16:A-D shows the comparison of the experimental time course training data (red dots) with the model simulation profiles generated using the estimated parameters (blue lines). The model predictions fit the training data well for most of the cases. In some cases, the simulations were unable to reproduce the trends of the data well. This

is likely due to the simplifications assumed by our model.

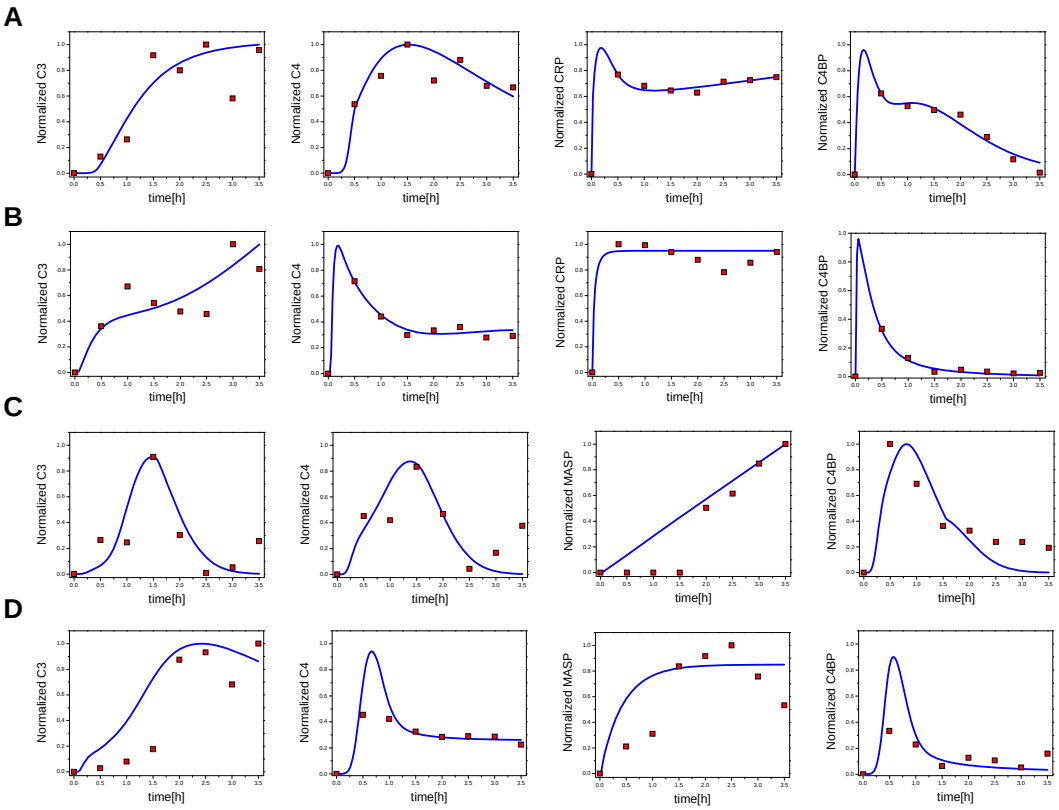


Figure 6.16: Experimental and simulated dynamics of the complement pathway. The time profiles of deposited C3, C4, MASP-2, CRP and C4BP under the following four conditions are simulated using estimated parameters and compared against the experimental data: (A) PC-initiated complement activation under inflammation condition, (B) PC-initiated complement activation under normal condition. (C) GlcNAc-initiated complement activation under inflammation condition; (D) GlcNAc-initiated complement activation under normal condition. Blue solid lines depict the simulation results and red dots indicate experimental data.

6.3.4 Model validation

We next validated the model using previously published experimental observations (Zhang et al., 2009). In particular, normalized concentration level of deposited C3 was used to predict the antibacterial activity since C3 deposition initiated the opsonization

Name	Initial Concentrations [nM]
CRP	0.2
PC	0.0327796
PC/CRP	0
C4	77
C4a	0
C4b	0
C2	31
C1	247
PC/CRP/C1	0
C2a	0
C2b	0
C4b/C2a	0
C3	465
C3a	0
C3b	0
dC3b	0
MASP	0.68
dC4b/C2a	0
GlcNac	0
GlcNac/LF	0
LF	2
GlcNac/LF/MASP	0
PC/CRP/LF	0
PC/CRP/LF/MASP	0
GlcNac/LF/CRP	0
GlcNac/LF/CRP/C1	0
C4BP	26
C4BP/PC/CRP	0
C4BP/GlcNac/LF/CRP	0
iC4b/C2a	0
C4BP/C4b	0
C4b/C2a/C4BP	0
dC4b/C2a/C4BP	0
PC/CRP/LF/C1	0
C4BP/PC/CRP/LF	0
GlcNac/LF/CRP/MASP	0
PC/CRP/LF/C1/MASP	0
GlcNac/HF	0
HF	0
GlcNac/HF/MASP	0
X	0
GlcNac/LF/CRP/C1/MASP	0

Table 6.10: The initial concentrations.

Parameter	Values
$ka01_1$	0.027599856
$ka01_2$	0.0109
$ka02_1$	$7.4E-4$
$ka02_2$	0.0011
$ka03_1*$	2.0
$ka04_1*$	10.5
$kc01_1$	0.64564663
$kc01_2$	0.19455111
$kc02$	$5.91E-4$
$kc03_1$	0.41400447
$kc03_2$	0.9964757
$kc04_1$	0.97783655
$ka03_2$	500.0
$ka04_2$	2500.0
$kd02_2$	0.1
$kd02_1$	0.0368011
$kd03_1$	66.3777
$kd03_2$	829.116
$kb01_1$	$1.45E-4$
$kb01_2$	0.07761722
$kb02_1$	$2.14E-7$
$kb02_2$	0.1
$kb03_1$	93.97925
$kb03_2$	8815.971
$kb04_1*$	1.1
$kb04_2*$	2000.0
$kc04_2$	0.19916244
$kd01_1$	$7.07E-5$
$kd01_2$	$7.23E-5$
$kd04_1*$	1.1
$kd04_2*$	2000.0
$ke01_1$	$7.07E-5$
$ke01_2$	$1.0E-4$
$ke02_1$	$7.4E-4$
$ke02_2$	0.0011
$ke03_1$	2.0
$ke03_2$	500.0
$ke04_1$	10.5
$ke04_2$	2500.0
$kf01_1$	0.9699983
$kf01_2$	0.06902058
$kf02_1$	0.25880134
$kf02_2$	0.4837216
$kf03$	0.06135372
$kf04_2$	0.9836912
$kf04_1$	0.6134161

Parameter	Values
$kf05$	0.98077756
$kf06_1$	0.613416
$kf06_2$	0.983691
$kf07_1$	0.613416
$kf07_2$	0.983691
$kd05_1$	7.4×10^{-4}
$kd05_2$	0.0011
$kd06_1*$	2.0
$kd06_2*$	500.0
$kd07_1*$	10.5
$kd07_2*$	2500.0
$ke05_1$	2.14×10^{-7}
$ke05_2$	0.1
$ke06_1$	93.97925
$ke06_2$	8815.971
$ke07_1*$	1.1
$ke07_2*$	2000.0
$kd08_1$	0.0368011
$kd08_2$	0.1
$kd09_1$	7.4×10^{-4}
$kd09_2$	0.0011
$kd10_1$	71.17058
$kd10_2$	3796.2268
$kd11_1$	38.96259
$kd11_2$	5972.3066
$kg01_1$	1.45×10^{-4}
$kg01_2$	0.07761722
$kg02_1$	2.14×10^{-7}
$kg02_2$	0.1
$kg03_1$	93.97925
$kg03_2$	8815.971
$kg04_1*$	1.1
$kg04_2*$	2000.0
$ke08_1$	2.14×10^{-7}
$ke08_2$	0.1
$ke09_1$	7.4×10^{-4}
$ke09_2$	0.0011
$ke10_1$	83.52653
$ke10_2$	0.010678623
$ke11_1$	79.544876
$ke11_2$	42.56355
$ktmp1$	3.42×10^{-4}
$ktmp2$	0.492901
$ktmp3$	0.0470911
$ktmpf1_1$	0.0
$ktmpf1_2$	0.0

Table 6.11: Parameter values. Known parameters are marked with *.

process and the lysis of bacteria. We first simulated the concentration level of deposited C3 at 1 hour under different conditions. We next normalized the results so that the maximum value among them equals to 95% which is the maximum bacterial killing rate reported in the experimental observations (Zhang et al., 2009). The normalized values were then treated as predicted bacterial killing rates. The simulation results are shown in Figure 6.17:A and 6.17:B as black bars. Consistent with the experimental data (Figure 6.17:A, grey bars), our simulation results showed that under the infection-inflammation conditions, the *P. aeruginosa* can be efficiently killed (95% bacterial killing rate) by complement whereas under the normal condition, only 28% of the bacteria succumbed (Figure 6.17:A, black bars). In the patient serum, depletion of CRP or ficolin induced a significant drop in the killing rate from 95% to 33% or 25% respectively, indicating that the synergistic action of CRP and L-ficolin accounted for around 40% of the enhanced killing effect. However, in the normal serum, depletion of CRP or ficolin only resulted in a slight drop in the killing rate from 28% to 18% or 10% respectively. Furthermore, simulating a high CRP level (such as in the case of cardiovascular disease) under the normal healthy condition did not further increase the bacterial killing rate. As shown in Figure 6.17:B, the simulation results matched the experimental data. Thus, our model was able to reproduce the published experimental observations shown in both Figure 6.17:A and 6.17:B with less than 10% error. This not only validated our model thus promoting its use for generating predictions, but also yielded positive evidence in support of the hypothesized amplification pathways induced by infection-inflammation condition. It also suggested that the antibacterial activity can be simulated efficiently by the level of deposited C3 and this was used to generate model predictions described in later sections.

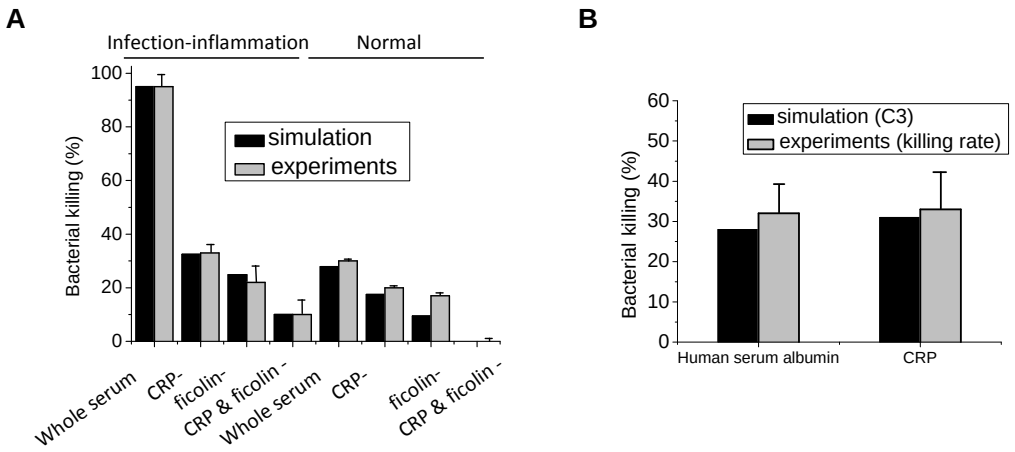


Figure 6.17: Model predictions and experimental validation of effects of the crosstalk. (A) Simulation results (black bar) of end-point bacterial killing rate in whole serum, CRP depleted serum (CRP-), ficolin-depleted serum (ficolin-), both CRP- and ficolin-depleted serum (CRP- & ficolin-) under normal and infection-inflammation conditions agree with the previous experimental observations (gray bar). (B) The simulated bacterial killing effect of high CRP level agrees with the experimental data.

6.3.5 Sensitivity analysis

In order to identify critical reactions that control complement activation during infection, we performed global sensitivity analysis using the DBN approximations. Multi-parametric sensitivity analysis (MPSA) (Zi et al., 2005) was performed on the DBN for PC-initiated complement cascade (the details are presented in Section 5.3). The results are shown in Figure 6.18. Strong controls over the whole system are distributed among the parameters associated with the immobilisation of C3b with the surface, interaction between CRP and L-ficolin, cleavage of C2 and C4, and the decay of C3 convertase (see Figure 6.15, reactions labeled in red). The sensitivity of reactions associated with C3, C2 and C4 highlight the significant role of major complement components. The high sensitivity of interaction of CRP and L-ficolin confirms that the overall antibacterial response depends on the strength of the crosstalk between the classical and lectin pathways. In addition, since the decay of C3 convertase is one of the regulatory targets

of C4BP, the sensitivity of the system to a change in the rate of decay of C3 convertase suggested that the regulatory mechanism by C4BP plays an important role in complement. Since the critical reactions identified are common in PC- and GlcNAc-initiated complement cascades, MPSA results using the other DBN will produce similar results and hence this analysis was not performed. We next focused our investigation on the enhancement mechanism by the crosstalk and the regulatory mechanism by C4BP.

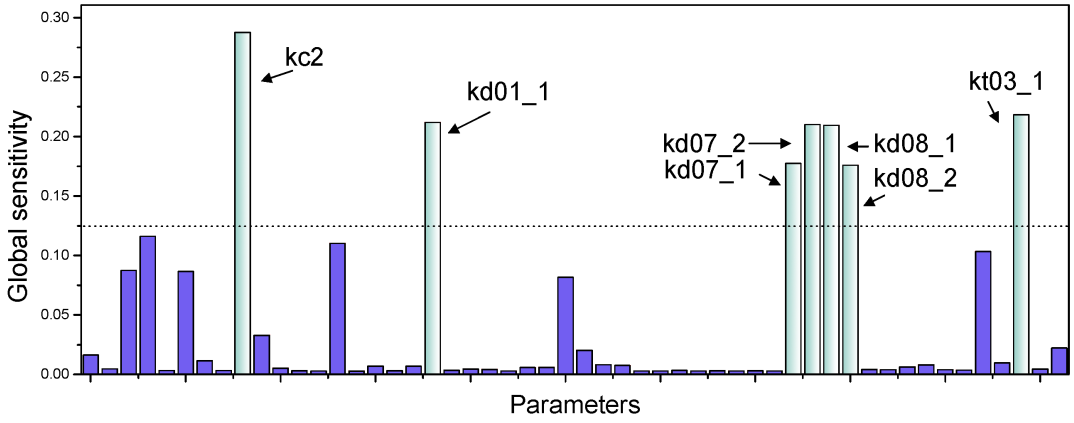


Figure 6.18: Global sensitivity analysis. Global sensitivities were calculated according to the MPSA method. The most sensitive parameters are colored in light blue. $kc2$ refers to the association rate of C3b with the surface. $kd01\_1$ refers to the association rate of CRP and ficolin. $kd07\_1$ and kd_07_2 are the Michaelis-Menten constants governing the cleavage rate of C2. $kd08\_1$ and kd_08_2 are the Michaelis-Menten constants governing the cleavage rate of C4. $kt03\_1$ refers to the decay rate of C4bC2a. Those reactions are colored in red in Figure 6.15.

6.3.6 The enhancement mechanism of the antimicrobial response

Under infection-inflammation conditions where PC-CRP:L-ficolin or GlcNAc-L-ficolin:CRP complex is formed, the amplification pathways are triggered. Model simulation showed that if C1 and L-ficolin or CRP and MASP-2 competed against each other, the antibacterial activity of the classical pathway or lectin pathway might be deprived of

the amplification pathways. Therefore, in order to achieve a stable enhancement, C1 and L-ficolin (or CRP and MASP-2) must simultaneously bind to CRP (or L-ficolin). Further, the abilities of CRP and L-ficolin to trigger subsequent complement cascade were not affected by the formation of this complex. This is consistent with the previous experimental observation that two amplification pathways co-exist with the classical and lectin pathways (Zhang et al., 2009).

The pH value and calcium level influence the conformations of CRP and L-ficolin which in turn govern their binding affinities. To investigate the effects of pH and calcium on the antibacterial response, We simulated the C3 deposition dynamics using the predicted binding affinities at pH ranging from 5.5 to 7.4 in the presence of 2 mM and 2.5 mM calcium. The results are shown in Figure 6.19. Under both 2 mM and 2.5 mM calcium conditions, decreasing pH induces not only the increase of the peak amplitude (maximum activation) but also hastens the peak time (time of maximum activation).

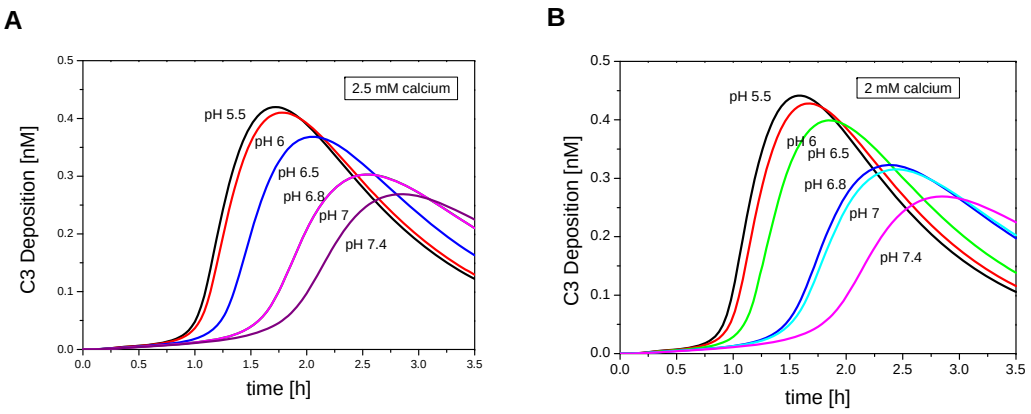


Figure 6.19: Simulation of antibacterial response with different pH and calcium level. (A) The deposited C3 time profile at pH ranging from 5.5 to 7.4, in the presence of 2 mM calcium. (B) The deposited C3 time profile at pH ranging from 5.5 to 7.4, in the presence of 2.5 mM calcium.

To further compare the effects of the two calcium levels, the dose-response curves

were generated as shown in Figure 6.20. At 2 mM calcium (blue curve), the antibacterial response was clearly greater than at 2.5 mM calcium (pink curve) indicating that slight hypocalcaemia enhanced the antibacterial activity in a stable manner. In addition, the pH-responses were reaching saturation levels when pH was near 5.5 (Figure 6.20), implying that the undesirable complement-enhancement by extreme low pH condition can be avoided. This also suggests that the saturation of the pH-response was influenced by the calcium level in the milieu.

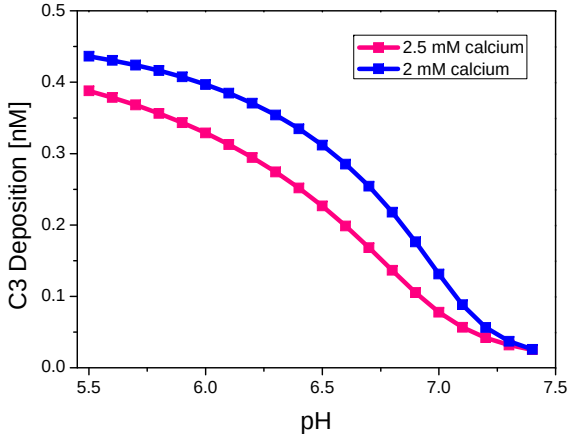


Figure 6.20: The pH-antibacterial response curves of complement activation in the presence of 2 mM calcium (pink) or 2.5 mM calcium (blue).

6.3.7 The regulatory mechanism of C4BP on the complement system

We next investigated the complement regulation by the major inhibitor, C4BP, under infection-inflammation conditions.

We varied the initial concentration of C4BP and simulated the PC- and GlcNAc-initiated complement under infection-inflammation conditions. The simulation time was chosen to be 5 hours which is slightly beyond the largest time point of our training experimental data. The predicted effects of the initial concentration of C4BP on the antibacterial response in terms of C3 deposition are shown in Figure 6.21:A-B.

For PC-initiated complement activation, when the starting amount of C4BP was perturbed around the normal level of 260 nM (Griffin et al., 1992), increasing C4BP level only delayed the peak time but did not decrease the peak amplitude significantly. In contrast, reducing the initial C4BP level clearly hastened the complement activation and maximized the activity. Interestingly, the GlcNAc-initiated complement activation (Figure 6.21:B) behaved differently from the PC-mediated complement activation (Figure 6.21:A). Around the normal level of 260 nM, perturbing the initial C4BP changed the maximum activity but did not affect the peak time, suggesting that C4BP plays distinct roles in regulating the classical and lectin pathways.

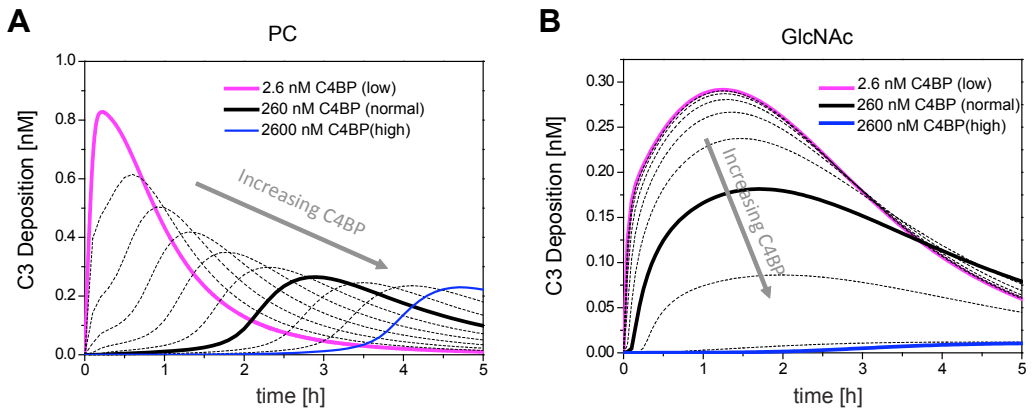


Figure 6.21: Model prediction of effects of C4BP under infection-inflammation condition. Predicted profiles of the deposited C3 after knocking down or over-expressing C4BP in the presence of PC (A) or GlcNAc (B).

Our results imply that C4BP regulates the lectin pathway more stringently than the classical pathway, which is consistent with previous experimental findings (Rawal et al., 2009). Further, for PC-initiated complement cascade, the over-expression of C4BP only delays but does not "turn off" the antibacterial response. In contrast, increased C4BP can efficiently inhibit GlcNAc-initiated complement activation. This may explain previous observations that bacteria such as *Yersinia enterocolitica*, *Streptococcus pyogenes*, *Neisseria gonorrhoeae*, *Escherichia coli* K1, *Moraxella catarrhalis*,

Candida albicans, *Bordetella pertussis* (Kirjavainen et al., 2008; Thern et al., 1995; Ram et al., 2001; Prasadarao et al., 2002; Nordstrom et al., 2004; Meri et al., 2004; Berggard et al., 1997) can exploit C4BP to evade complement.

We next investigated how C4BP mediates its inhibitory function. As shown in Figure 6.14, the inhibitory effects of C4BP target different sites in complement: (a) binding to CRP and blocking C1, (b) preventing the formation of C4bC2a by binding to C4b, (c) acting as a cofactor for factor I in the proteolytic inactivation of C4b, and (d) accelerating the natural decay of the C4bC2a complex, which prevents the formation of C4bC2a and disrupts already formed convertase. To identify the dominant mechanism, we employed *in silico* knockout of the reactions involved for each mechanism and performed simulations. Figure 6.22 shows the model predictions. Among the four inhibitory mechanisms, only the knockout of reaction (d) significantly enhanced the complement activation suggesting that facilitating the natural decay of C4bC2a (C3 convertase) is the most important inhibitory function of C4BP. This is consistent with our previous observations derived from sensitivity analysis, which identified the decay of C3 convertase as a critical reaction. In addition, as the inhibitory effect of reaction (d) is stronger than others, knocking out reaction (a) and (b) can even reduce the complement activity, which is counter-intuitive and emphasizes the significance of the systems-level understanding. As the enhancement mechanism by the crosstalk between CRP and L-ficolin occurs upstream of the cascade, we envisage C4BP acts downstream to 'quality control' and modulate C3 convertase activity. Thus our results suggest that efficient regulation of complement can be achieved by targeting the C3 convertase, where the complement pathways merge.

Our model predictions on the effects of C4BP have been experimentally verified. The experimental methods and data can be found in Liu et al. (2010).

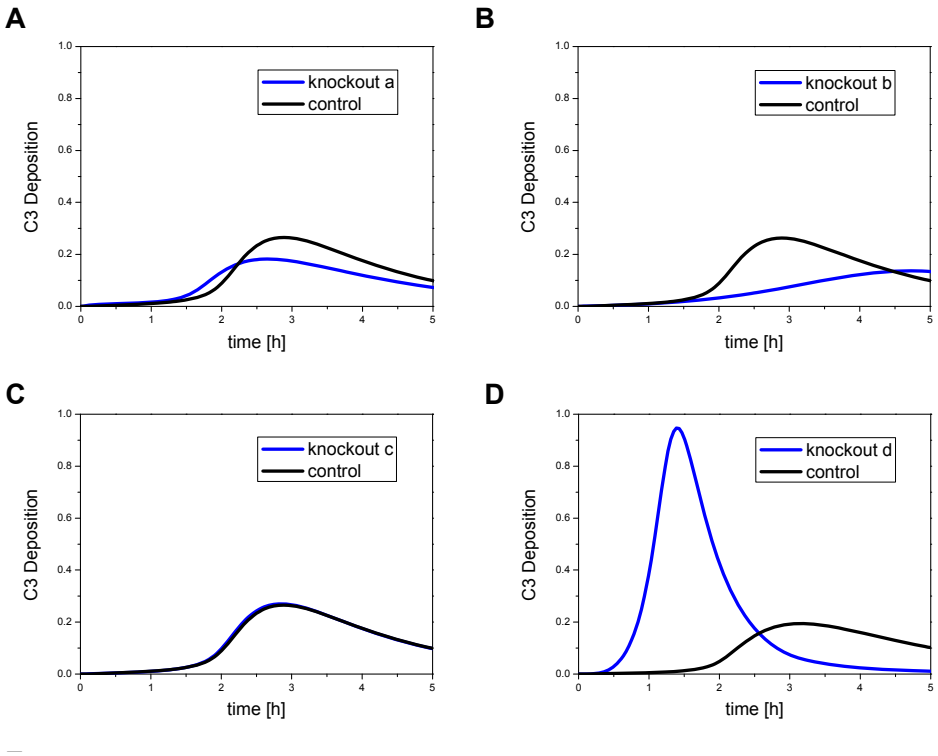


Figure 6.22: Knockout simulations reveal the major role of C4BP. (A) Simulation profiles of C3 deposition with or without reaction *a*. (B) Simulation profiles of C3 deposition with or without reaction *b*. (C) Simulation profiles of C3 deposition with or without reaction *c*. (D) Simulation profiles of C3 deposition with or without reaction *d*. Reactions (*a*-*d*) are labeled red in Figure 6.14 and explained in the caption: (*a*) C4BP binds to CRP, (*b*) C4BP binds to C4b, (*c*) C4BP prevents the assembly of C4bC2a, and (*d*) C4BP accelerates the decay of the C4bC2a.

In summary, by integrating our computational model and experimental observations we have obtained novel insights into how the complement activation is enhanced during infection and how excessive complement activity may be avoided. This introduces a new level of understanding of the host defense against bacterial infection. It also provides a platform for the potential development of complement-based immunomodulation therapies by exploiting the sensitivities of the perturbations of the pH, calcium and C4BP levels.

Chapter 7

Conclusion

We have proposed a probabilistic approximation scheme for biological pathway dynamics specified as a system of ODEs. Assuming a discretization and an initial distribution, it consists of pre-computing and storing a representative sample of trajectories induced by the system of ODEs. We use a dynamic Bayesian network representation to compactly represent these trajectories by exploiting the pathway structure. Basically, the underlying graph of the DBN approximation captures the dependencies of the variables on other variables and rate constants as defined by the system of ODEs. Due to the probabilistic graphical representation, a variety of analysis questions concerning the pathway dynamics traditionally addressed using Monte Carlo simulations can be converted to Bayesian inference and solved more efficiently. Using the FF algorithm for doing basic Bayesian inference, we have adapted standard parameter estimation and sensitivity analysis algorithms to the DBN setting.

We have demonstrated the applicability of our techniques with the help of the EGF-NGF signaling pathway, the segmentation clock pathway and the complement system. The DBN approximations we constructed successfully captured the dynamics of the three pathways. We showed that with the DBN approximations the unknown rate constants can be efficiently estimated from noisy experimental data.

We also gained insights about the pathway dynamics by identifying critical parameters in signal transduction via global sensitivity analysis. At the end of performing these analysis tasks we had easily regained the initial computational investment made to construct the DBN approximation.

Apart from its computational efficiency, it is worth noting that the DBN approximation is a more realistic model for recording the current state of knowledge about a biological pathway. In particular, the probabilistic and intervals-based estimates it returns will better match the noisy experimental data with limited precisions.

Turning next to the biological contributions of this thesis, in the third case study, we developed an ODE-based model for the complement system accompanied by DBN approximations. The motivation was to understand how the complement activity is boosted under local inflammation conditions while a tight surveillance is established to attain homeostasis.

Our study has involved a tight integration of computational and experimental aspects. The model analysis confirmed that the enhancement of complement activity under infection-inflammation condition was attributable to the synergistic action of CRP and L-ficolin and supported the existence of the amplification pathways. We also showed that the antimicrobial response is sensitive to changes in pH and calcium levels, which determines the strength of the crosstalk between CRP and L-ficolin.

Through model analysis we found that the inhibitor C4BP regulates the lectin pathway more stringently than the classical patwhay. The over-expresion of C4BP only delays but dose not reduce classical complement activation, whereas it attenuates but does not delay the complement activation of lectin pathway. We also found that, of the four documented inhibitory roles, C4BP acts mainly by facilitating the natural decay of the C3 convertase. These predictions were validated empirically. As the enhancement mechanism by the crosstalk between CRP and L-ficolin occurs upstream of the cascade, we envisage C4BP acts downstream to 'quality control' and modulate

C3 convertase activity. Thus our results suggest that efficient regulation of complement can be achieved by targeting the C3 convertase, where the complement pathways merge. These insights concerning the regulatory mechanisms of the complement system can potentially contribute to the development of complement-based immunomodulation therapies.

7.1 Future Work

A crucial ingredient in the construction of the DBN is the family of sample trajectories that are needed to get a good approximation. The DBN we construct approximates idealized Markov chain induced by the ODEs dynamics. Given an error bound, a confidence level and the transition probabilities of the idealized Markov chain, we can estimate (upper bound) the sample size required to fall within the given error bound with the required confidence level. However, the transition probabilities of the idealized Markov chain will not computable. Hence we pragmatically determine the sample size based on our sampling methods. How to determine the sample size with guaranteed error bounds is an issue we are continuing to study.

Though it is only a one-time cost, generating a representative set of trajectories to construct the DBN approximation for large pathway models will be computationally intensive. In the studies presented in this thesis, we parallelized the DBN construction code and executed it on a PC cluster. Clusters and supercomputers are expensive and the resources are often shared by a crowd of researchers. To enhance the usability of our approach, we are mapping our implementation onto graphics processing units (GPUs) in a on-going work in collaboration with computer architecture experts. We have been able to map the DBN construction process onto the GPU. A preliminary study of the EGF-NGF signaling pathway show promising results by achieving 5 fold speedup when compared to the cluster implementation present in this thesis (Chattopdhyay, 2010). We are continuing to explore the applicability of GPUs in our setting.

The long term aim of our approach is to aid biologists to understand the mechanisms of biological pathways. On this light, we need to apply our method to a variety of pathway models. We are currently doing so in collaboration with biologists in the settings of Apoptosis/Autophage pathways and DNA damage/repair pathways.

Further, it will be useful to augment the ODE model with some discrete features (e.g. HFPN models) but this should be easy to achieve.

The PRISM tool facilitates the rule-based stochastic formalisms to explore complex properties of the pathway dynamics via probabilistic model checking. However, as the state space of the underlying CTMCs of such models are exponential in the number of species, analyzing large pathways will be computation prohibitive. Hence, it is also important to develop formal verification techniques based on the DBN representation. In this context, we note that the FF algorithm can compute -although approximately- the marginal probabilities of the discretized values of variables at specific time points. Hence it will be appropriate to develop probabilistic bounded model checking methods for the DBN approximation and we are beginning to pursue this.

Related probabilistic formalisms such as Multi Terminal Binary Decision Diagrams (MTBDDs) and Probabilistic Decision Graphs (PDGs) are available for model checking. It is not clear at present how they can be derived directly from the ODE model. One could however try to convert our DBNs to MTBDDs for purposes of model checking (Langmead et al., 2006b) or develop statistical model checking methods (Jha et al., 2009). As compact representations of the probability distributions, PDGs are, in spirit, similar to Bayesian networks (Bozga and Maler, 1999) and can be computationally as efficient as Bayesian networks (Jaeger, 2004). Further, probabilistic inference can be carried out with a time complexity linear in the size of the PDGs (Jaeger, 2004). Thus, it will be an interesting future direction for us to explore the performance of PDGs in our setting.

Finally, our approximation technique might have wider applicability. A rich class

of dynamical systems can be captured via ODEs and in a variety of situations it may be appropriate and useful to abstract their behaviors as dynamic Bayesian networks as we have done here.

Appendix A

Supplementary Information for Chapter 6

This appendix contains the supplementary information for the third case study presented in Chapter 6. We first show the ODEs of the complement system model we constructed. We then present the details of the materials and methods for generating the experimental data we used in this case study. More information can be found in Liu et al. (2010).

A.1 The ODE Model

$$
\begin{aligned}
\frac{\mathrm{d}\left([\mathrm{CRP}]\right)}{\mathrm{d}t} &= -\left((\mathrm{ka01}_1 \cdot [\mathrm{PC}] \cdot [\mathrm{CRP}] - \mathrm{ka01}_2 \cdot [\mathrm{PC/CRP}])\right) \\
&\quad -\left((\mathrm{ke01}_1 \cdot [\mathrm{GlcNac/LF}] \cdot [\mathrm{CRP}] - \mathrm{ke01}_2 \cdot [\mathrm{GlcNac/LF/CRP}])\right) \\
\frac{\mathrm{d}\left([\mathrm{PC}]\right)}{\mathrm{d}t} &= -\left((\mathrm{ka01}_1 \cdot [\mathrm{PC}] \cdot [\mathrm{CRP}] - \mathrm{ka01}_2 \cdot [\mathrm{PC/CRP}])\right) \\
\frac{\mathrm{d}\left([\mathrm{PC/CRP}]\right)}{\mathrm{d}t} &= +\left((\mathrm{ka01}_1 \cdot [\mathrm{PC}] \cdot [\mathrm{CRP}] - \mathrm{ka01}_2 \cdot [\mathrm{PC/CRP}])\right) \\
&\quad -\left((\mathrm{ka02}_1 \cdot [\mathrm{PC/CRP}] \cdot [\mathrm{C1}] - \mathrm{ka02}_2 \cdot [\mathrm{PC/CRP/C1}])\right) \\
&\quad -\left((\mathrm{kf01}_1 \cdot [\mathrm{C4BP}] \cdot [\mathrm{PC/CRP}] - \mathrm{kf01}_2 \cdot [\mathrm{C4BP/PC/CRP}])\right) \\
&\quad -\left((\mathrm{kd01}_1 \cdot [\mathrm{PC/CRP}] \cdot [\mathrm{LF}] - \mathrm{kd01}_2 \cdot [\mathrm{PC/CRP/LF}])\right) \\
\frac{\mathrm{d}\left([\mathrm{C4}]\right)}{\mathrm{d}t} &= -\left(\frac{\mathrm{kd03}_1 \cdot [\mathrm{PC/CRP/LF/MASP}] \cdot [\mathrm{C4}]}{\mathrm{kd03}_2 + [\mathrm{C4}]}\right) \\
&\quad -\left(\frac{\mathrm{kb03}_1 \cdot [\mathrm{GlcNac/LF/MASP}] \cdot [\mathrm{C4}]}{\mathrm{kb03}_2 + [\mathrm{C4}]}\right) \\
&\quad -\left(\frac{\mathrm{ke03}_1 \cdot [\mathrm{GlcNac/LF/CRP/C1}] \cdot [\mathrm{C4}]}{\mathrm{ke03}_2 + [\mathrm{C4}]}\right) \\
&\quad -\left(\frac{\mathrm{ka03}_1 \cdot [\mathrm{PC/CRP/C1}] \cdot [\mathrm{C4}]}{\mathrm{ka03}_2 + [\mathrm{C4}]}\right) \\
&\quad -\left(\frac{\mathrm{kd06}_1 \cdot [\mathrm{PC/CRP/LF/C1}] \cdot [\mathrm{C4}]}{\mathrm{kd06}_2 + [\mathrm{C4}]}\right) \\
&\quad -\left(\frac{\mathrm{ke06}_1 \cdot [\mathrm{GlcNac/LF/CRP/MASP}] \cdot [\mathrm{C4}]}{\mathrm{ke06}_2 + [\mathrm{C4}]}\right) \\
&\quad -\left(\frac{\mathrm{kd10}_1 \cdot [\mathrm{PC/CRP/LF/C1/MASP}] \cdot [\mathrm{C4}]}{\mathrm{kd10}_2 + [\mathrm{C4}]}\right) \\
&\quad -\left(\frac{\mathrm{kg03}_1 \cdot [\mathrm{GlcNac/HF/MASP}] \cdot [\mathrm{C4}]}{\mathrm{kg03}_2 + [\mathrm{C4}]}\right) \\
\frac{\mathrm{d}\left([\mathrm{C4a}]\right)}{\mathrm{d}t} &= +\left(\frac{\mathrm{kd03}_1 \cdot [\mathrm{PC/CRP/LF/MASP}] \cdot [\mathrm{C4}]}{\mathrm{kd03}_2 + [\mathrm{C4}]}\right) \\
&\quad +\left(\frac{\mathrm{kb03}_1 \cdot [\mathrm{GlcNac/LF/MASP}] \cdot [\mathrm{C4}]}{\mathrm{kb03}_2 + [\mathrm{C4}]}\right) \\
&\quad +\left(\frac{\mathrm{ke03}_1 \cdot [\mathrm{GlcNac/LF/CRP/C1}] \cdot [\mathrm{C4}]}{\mathrm{ke03}_2 + [\mathrm{C4}]}\right) \\
&\quad +\left(\frac{\mathrm{ka03}_1 \cdot [\mathrm{PC/CRP/C1}] \cdot [\mathrm{C4}]}{\mathrm{ka03}_2 + [\mathrm{C4}]}\right) \\
&\quad +\left(\frac{\mathrm{kd06}_1 \cdot [\mathrm{PC/CRP/LF/C1}] \cdot [\mathrm{C4}]}{\mathrm{kd06}_2 + [\mathrm{C4}]}\right) \\
&\quad +\left(\frac{\mathrm{ke06}_1 \cdot [\mathrm{GlcNac/LF/CRP/MASP}] \cdot [\mathrm{C4}]}{\mathrm{ke06}_2 + [\mathrm{C4}]}\right) \\
&\quad +\left(\frac{\mathrm{kd10}_1 \cdot [\mathrm{PC/CRP/LF/C1/MASP}] \cdot [\mathrm{C4}]}{\mathrm{kd10}_2 + [\mathrm{C4}]}\right) \\
&\quad +\left(\frac{\mathrm{kg03}_1 \cdot [\mathrm{GlcNac/HF/MASP}] \cdot [\mathrm{C4}]}{\mathrm{kg03}_2 + [\mathrm{C4}]}\right)
\end{aligned}
$$

$$
\begin{aligned}
\frac{\mathrm{d}\left([\mathrm{C4b}]\right)}{\mathrm{d}t} &= +\left(\frac{\mathrm{kd03}_1 \cdot [\mathrm{PC/CRP/LF/MASP}] \cdot [\mathrm{C4}]}{\mathrm{kd03}_2 + [\mathrm{C4}]}\right) \\
&\quad +\left(\frac{\mathrm{kb03}_1 \cdot [\mathrm{GlcNac/LF/MASP}] \cdot [\mathrm{C4}]}{\mathrm{kb03}_2 + [\mathrm{C4}]}\right) \\
&\quad +\left(\frac{\mathrm{ke03}_1 \cdot [\mathrm{GlcNac/LF/CRP/C1}] \cdot [\mathrm{C4}]}{\mathrm{ke03}_2 + [\mathrm{C4}]}\right) \\
&\quad +\left(\frac{\mathrm{ka03}_1 \cdot [\mathrm{PC/CRP/C1}] \cdot [\mathrm{C4}]}{\mathrm{ka03}_2 + [\mathrm{C4}]}\right) \\
&\quad -((\mathrm{kf04}_1 \cdot [\mathrm{C4BP}] \cdot [\mathrm{C4b}] - \mathrm{kf04}_2 \cdot [\mathrm{C4BP/C4b}])) \\
&\quad +(\mathrm{kf05} \cdot [\mathrm{C4b/C2a}] \cdot [\mathrm{C4BP}]) \\
&\quad +\left(\frac{\mathrm{kd06}_1 \cdot [\mathrm{PC/CRP/LF/C1}] \cdot [\mathrm{C4}]}{\mathrm{kd06}_2 + [\mathrm{C4}]}\right) \\
&\quad +\left(\frac{\mathrm{ke06}_1 \cdot [\mathrm{GlcNac/LF/CRP/MASP}] \cdot [\mathrm{C4}]}{\mathrm{ke06}_2 + [\mathrm{C4}]}\right) \\
&\quad +\left(\frac{\mathrm{kd10}_1 \cdot [\mathrm{PC/CRP/LF/C1/MASP}] \cdot [\mathrm{C4}]}{\mathrm{kd10}_2 + [\mathrm{C4}]}\right) \\
&\quad -((\mathrm{kc01}_1 \cdot [\mathrm{C4b}] \cdot [\mathrm{C2a}] - \mathrm{kc01}_2 \cdot [\mathrm{C4b/C2a}])) \\
&\quad +\left(\frac{\mathrm{kg03}_1 \cdot [\mathrm{GlcNac/HF/MASP}] \cdot [\mathrm{C4}]}{\mathrm{kg03}_2 + [\mathrm{C4}]}\right) \\
\frac{\mathrm{d}\left([\mathrm{C2}]\right)}{\mathrm{d}t} &= -\left(\frac{\mathrm{kd04}_1 \cdot [\mathrm{PC/CRP/LF/MASP}] \cdot [\mathrm{C2}]}{\mathrm{kd04}_2 + [\mathrm{C2}]}\right) \\
&\quad -\left(\frac{\mathrm{kb04}_1 \cdot [\mathrm{GlcNac/LF/MASP}] \cdot [\mathrm{C2}]}{\mathrm{kb04}_2 + [\mathrm{C2}]}\right) \\
&\quad -\left(\frac{\mathrm{ke04}_1 \cdot [\mathrm{GlcNac/LF/CRP/C1}] \cdot [\mathrm{C2}]}{\mathrm{ke04}_2 + [\mathrm{C2}]}\right) \\
&\quad -\left(\frac{\mathrm{ka04}_1 \cdot [\mathrm{PC/CRP/C1}] \cdot [\mathrm{C2}]}{\mathrm{ka04}_2 + [\mathrm{C2}]}\right) \\
&\quad -\left(\frac{\mathrm{kd07}_1 \cdot [\mathrm{PC/CRP/LF/C1}] \cdot [\mathrm{C2}]}{\mathrm{kd07}_2 + [\mathrm{C2}]}\right) \\
&\quad -\left(\frac{\mathrm{ke07}_1 \cdot [\mathrm{GlcNac/LF/CRP/MASP}] \cdot [\mathrm{C2}]}{\mathrm{ke07}_2 + [\mathrm{C2}]}\right) \\
&\quad -\left(\frac{\mathrm{kd11}_1 \cdot [\mathrm{PC/CRP/LF/C1/MASP}] \cdot [\mathrm{C2}]}{\mathrm{kd11}_2 + [\mathrm{C2}]}\right) \\
&\quad -\left(\frac{\mathrm{kg04}_1 \cdot [\mathrm{GlcNac/HF/MASP}] \cdot [\mathrm{C2}]}{\mathrm{kg04}_2 + [\mathrm{C2}]}\right) \\
\frac{\mathrm{d}\left([\mathrm{C1}]\right)}{\mathrm{d}t} &= -((\mathrm{ka02}_1 \cdot [\mathrm{PC/CRP}] \cdot [\mathrm{C1}] - \mathrm{ka02}_2 \cdot [\mathrm{PC/CRP/C1}])) \\
&\quad -((\mathrm{ke02}_1 \cdot [\mathrm{GlcNac/LF/CRP}] \cdot [\mathrm{C1}] - \mathrm{ke02}_2 \cdot [\mathrm{GlcNac/LF/CRP/C1}])) \\
&\quad -((\mathrm{kd05}_1 \cdot [\mathrm{PC/CRP/LF}] \cdot [\mathrm{C1}] - \mathrm{kd05}_2 \cdot [\mathrm{PC/CRP/LF/C1}])) \\
&\quad -((\mathrm{kd09}_1 \cdot [\mathrm{PC/CRP/LF/MASP}] \cdot [\mathrm{C1}] - \mathrm{kd09}_2 \cdot [\mathrm{PC/CRP/LF/C1/MASP}])) \\
\frac{\mathrm{d}\left([\mathrm{PC/CRP/C1}]\right)}{\mathrm{d}t} &= +((\mathrm{ka02}_1 \cdot [\mathrm{PC/CRP}] \cdot [\mathrm{C1}] - \mathrm{ka02}_2 \cdot [\mathrm{PC/CRP/C1}]))
\end{aligned}
$$

$$
\begin{aligned}
\frac{\mathrm{d}\left([\mathrm{C2a}]\right)}{\mathrm{d}t} &= +\left(\frac{\mathrm{kd04}_1 \cdot [\mathrm{PC/CRP/LF/MASP}] \cdot [\mathrm{C2}]}{\mathrm{kd04}_2 + [\mathrm{C2}]}\right) \\
&\quad + \left(\frac{\mathrm{kb04}_1 \cdot [\mathrm{GlcNac/LF/MASP}] \cdot [\mathrm{C2}]}{\mathrm{kb04}_2 + [\mathrm{C2}]}\right) \\
&\quad + \left(\frac{\mathrm{ke04}_1 \cdot [\mathrm{GlcNac/LF/CRP/C1}] \cdot [\mathrm{C2}]}{\mathrm{ke04}_2 + [\mathrm{C2}]}\right) \\
&\quad + (\mathrm{kf05} \cdot [\mathrm{C4b/C2a}] \cdot [\mathrm{C4BP}]) \\
&\quad + \left(\frac{\mathrm{ka04}_1 \cdot [\mathrm{PC/CRP/C1}] \cdot [\mathrm{C2}]}{\mathrm{ka04}_2 + [\mathrm{C2}]}\right) \\
&\quad + \left(\frac{\mathrm{kd07}_1 \cdot [\mathrm{PC/CRP/LF/C1}] \cdot [\mathrm{C2}]}{\mathrm{kd07}_2 + [\mathrm{C2}]}\right) \\
&\quad + \left(\frac{\mathrm{ke07}_1 \cdot [\mathrm{GlcNac/LF/CRP/MASP}] \cdot [\mathrm{C2}]}{\mathrm{ke07}_2 + [\mathrm{C2}]}\right) \\
&\quad - ((\mathrm{kc01}_1 \cdot [\mathrm{C4b}] \cdot [\mathrm{C2a}] - \mathrm{kc01}_2 \cdot [\mathrm{C4b/C2a}])) \\
&\quad + \left(\frac{\mathrm{kd11}_1 \cdot [\mathrm{PC/CRP/LF/C1/MASP}] \cdot [\mathrm{C2}]}{\mathrm{kd11}_2 + [\mathrm{C2}]}\right) \\
&\quad + \left(\frac{\mathrm{kg04}_1 \cdot [\mathrm{GlcNac/HF/MASP}] \cdot [\mathrm{C2}]}{\mathrm{kg04}_2 + [\mathrm{C2}]}\right) \\
\frac{\mathrm{d}\left([\mathrm{C2b}]\right)}{\mathrm{d}t} &= +\left(\frac{\mathrm{kd04}_1 \cdot [\mathrm{PC/CRP/LF/MASP}] \cdot [\mathrm{C2}]}{\mathrm{kd04}_2 + [\mathrm{C2}]}\right) \\
&\quad + \left(\frac{\mathrm{kb04}_1 \cdot [\mathrm{GlcNac/LF/MASP}] \cdot [\mathrm{C2}]}{\mathrm{kb04}_2 + [\mathrm{C2}]}\right) \\
&\quad + \left(\frac{\mathrm{ke04}_1 \cdot [\mathrm{GlcNac/LF/CRP/C1}] \cdot [\mathrm{C2}]}{\mathrm{ke04}_2 + [\mathrm{C2}]}\right) \\
&\quad + \left(\frac{\mathrm{ka04}_1 \cdot [\mathrm{PC/CRP/C1}] \cdot [\mathrm{C2}]}{\mathrm{ka04}_2 + [\mathrm{C2}]}\right) \\
&\quad + \left(\frac{\mathrm{kd07}_1 \cdot [\mathrm{PC/CRP/LF/C1}] \cdot [\mathrm{C2}]}{\mathrm{kd07}_2 + [\mathrm{C2}]}\right) \\
&\quad + \left(\frac{\mathrm{ke07}_1 \cdot [\mathrm{GlcNac/LF/CRP/MASP}] \cdot [\mathrm{C2}]}{\mathrm{ke07}_2 + [\mathrm{C2}]}\right) \\
&\quad + \left(\frac{\mathrm{kd11}_1 \cdot [\mathrm{PC/CRP/LF/C1/MASP}] \cdot [\mathrm{C2}]}{\mathrm{kd11}_2 + [\mathrm{C2}]}\right) \\
&\quad + \left(\frac{\mathrm{kg04}_1 \cdot [\mathrm{GlcNac/HF/MASP}] \cdot [\mathrm{C2}]}{\mathrm{kg04}_2 + [\mathrm{C2}]}\right) \\
\frac{\mathrm{d}\left([\mathrm{C4b/C2a}]\right)}{\mathrm{d}t} &= -(\mathrm{kf03} \cdot [\mathrm{C4b/C2a}] \cdot [\mathrm{C4BP}]) \\
&\quad - (\mathrm{kf05} \cdot [\mathrm{C4b/C2a}] \cdot [\mathrm{C4BP}]) \\
&\quad - ((\mathrm{kf06}_1 \cdot [\mathrm{C4b/C2a}] \cdot [\mathrm{C4BP}] - \mathrm{kf06}_2 \cdot [\mathrm{C4b/C2a/C4BP}])) \\
&\quad - \left(\mathrm{k1}_{(\mathrm{tmp3})} \cdot [\mathrm{C4b/C2a}]\right) \\
&\quad + ((\mathrm{kc01}_1 \cdot [\mathrm{C4b}] \cdot [\mathrm{C2a}] - \mathrm{kc01}_2 \cdot [\mathrm{C4b/C2a}])) \\
&\quad - ((\mathrm{kc04}_1 \cdot [\mathrm{C4b/C2a}] - \mathrm{kc04}_2 \cdot [\mathrm{dC4b/C2a}])) \\
\frac{\mathrm{d}\left([\mathrm{C3}]\right)}{\mathrm{d}t} &= -(\mathrm{kc02} \cdot [\mathrm{C4b/C2a}] \cdot [\mathrm{C3}])
\end{aligned}
$$

$$
\begin{aligned}
\frac{\mathrm{d}\left([\mathrm{C3a}]\right)}{\mathrm{d}t} &= +\left(\mathrm{kc02}\cdot[\mathrm{C4b/C2a}]\cdot[\mathrm{C3}]\right)\\
\frac{\mathrm{d}\left([\mathrm{C3b}]\right)}{\mathrm{d}t} &= -\left(\mathrm{k1}_{(\mathrm{tmp2})}\cdot[\mathrm{C3b}]\right)\\
&\quad +\left(\mathrm{kc02}\cdot[\mathrm{C4b/C2a}]\cdot[\mathrm{C3}]\right)\\
&\quad -\left((\mathrm{kc03}_1\cdot[\mathrm{C3b}]-\mathrm{kc03}_2\cdot[\mathrm{dC3b}])\right)\\
\frac{\mathrm{d}\left([\mathrm{dC3b}]\right)}{\mathrm{d}t} &= +\left((\mathrm{kc03}_1\cdot[\mathrm{C3b}]-\mathrm{kc03}_2\cdot[\mathrm{dC3b}])\right)\\
\frac{\mathrm{d}\left([\mathrm{MASP}]\right)}{\mathrm{d}t} &= -\left((\mathrm{kb02}_1\cdot[\mathrm{GlcNac/LF}]\cdot[\mathrm{MASP}]-\mathrm{kb02}_2\cdot[\mathrm{GlcNac/LF/MASP}])\right)\\
&\quad -\left((\mathrm{ke05}_1\cdot[\mathrm{GlcNac/LF/CRP}]\cdot[\mathrm{MASP}]-\mathrm{ke05}_2\cdot[\mathrm{GlcNac/LF/CRP/MASP}])\right)\\
&\quad -\left((\mathrm{kd08}_1\cdot[\mathrm{PC/CRP/LF/C1}]\cdot[\mathrm{MASP}]-\mathrm{kd08}_2\cdot[\mathrm{PC/CRP/LF/C1/MASP}])\right)\\
&\quad -\left((\mathrm{kg02}_1\cdot[\mathrm{GlcNac/HF}]\cdot[\mathrm{MASP}]-\mathrm{kg02}_2\cdot[\mathrm{GlcNac/HF/MASP}])\right)\\
&\quad -\left((\mathrm{kd02}_1\cdot[\mathrm{PC/CRP/LF}]\cdot[\mathrm{MASP}]-\mathrm{kd02}_2\cdot[\mathrm{PC/CRP/LF/MASP}])\right)\\
\frac{\mathrm{d}\left([\mathrm{dC4b/C2a}]\right)}{\mathrm{d}t} &= -\left((\mathrm{kf07}_1\cdot[\mathrm{dC4b/C2a}]\cdot[\mathrm{C4BP}]-\mathrm{kf07}_2\cdot[\mathrm{dC4b/C2a/C4BP}])\right)\\
&\quad +\left((\mathrm{kc04}_1\cdot[\mathrm{C4b/C2a}]-\mathrm{kc04}_2\cdot[\mathrm{dC4b/C2a}])\right)\\
\frac{\mathrm{d}\left([\mathrm{GlcNac}]\right)}{\mathrm{d}t} &= -\left((\mathrm{kb01}_1\cdot[\mathrm{GlcNac}]\cdot[\mathrm{LF}]-\mathrm{kb01}_2\cdot[\mathrm{GlcNac/LF}])\right)\\
\frac{\mathrm{d}\left([\mathrm{GlcNac/LF}]\right)}{\mathrm{d}t} &= +\left((\mathrm{kb01}_1\cdot[\mathrm{GlcNac}]\cdot[\mathrm{LF}]-\mathrm{kb01}_2\cdot[\mathrm{GlcNac/LF}])\right)\\
&\quad -\left((\mathrm{kb02}_1\cdot[\mathrm{GlcNac/LF}]\cdot[\mathrm{MASP}]-\mathrm{kb02}_2\cdot[\mathrm{GlcNac/LF/MASP}])\right)\\
&\quad -\left((\mathrm{ke01}_1\cdot[\mathrm{GlcNac/LF}]\cdot[\mathrm{CRP}]-\mathrm{ke01}_2\cdot[\mathrm{GlcNac/LF/CRP}])\right)\\
\frac{\mathrm{d}\left([\mathrm{LF}]\right)}{\mathrm{d}t} &= -\left((\mathrm{kb01}_1\cdot[\mathrm{GlcNac}]\cdot[\mathrm{LF}]-\mathrm{kb01}_2\cdot[\mathrm{GlcNac/LF}])\right)\\
&\quad -\left((\mathrm{kd01}_1\cdot[\mathrm{PC/CRP}]\cdot[\mathrm{LF}]-\mathrm{kd01}_2\cdot[\mathrm{PC/CRP/LF}])\right)\\
\frac{\mathrm{d}\left([\mathrm{GlcNac/LF/MASP}]\right)}{\mathrm{d}t} &= +\left((\mathrm{kb02}_1\cdot[\mathrm{GlcNac/LF}]\cdot[\mathrm{MASP}]-\mathrm{kb02}_2\cdot[\mathrm{GlcNac/LF/MASP}])\right)\\
\frac{\mathrm{d}\left([\mathrm{PC/CRP/LF}]\right)}{\mathrm{d}t} &= -\left((\mathrm{kd05}_1\cdot[\mathrm{PC/CRP/LF}]\cdot[\mathrm{C1}]-\mathrm{kd05}_2\cdot[\mathrm{PC/CRP/LF/C1}])\right)\\
&\quad -\left(\left(\mathrm{k1}_{(\mathrm{tmpf1})}\cdot[\mathrm{C4BP}]\cdot[\mathrm{PC/CRP/LF}]-\mathrm{k2}_{(\mathrm{tmpf1})}\cdot[\mathrm{C4BP/PC/CRP/LF}]\right)\right)\\
&\quad +\left((\mathrm{kd01}_1\cdot[\mathrm{PC/CRP}]\cdot[\mathrm{LF}]-\mathrm{kd01}_2\cdot[\mathrm{PC/CRP/LF}])\right)\\
&\quad -\left((\mathrm{kd02}_1\cdot[\mathrm{PC/CRP/LF}]\cdot[\mathrm{MASP}]-\mathrm{kd02}_2\cdot[\mathrm{PC/CRP/LF/MASP}])\right)\\
\frac{\mathrm{d}\left([\mathrm{PC/CRP/LF/MASP}]\right)}{\mathrm{d}t} &= -\left((\mathrm{kd09}_1\cdot[\mathrm{PC/CRP/LF/MASP}]\cdot[\mathrm{C1}]-\mathrm{kd09}_2\cdot[\mathrm{PC/CRP/LF/C1/MASP}])\right)\\
&\quad +\left((\mathrm{kd02}_1\cdot[\mathrm{PC/CRP/LF}]\cdot[\mathrm{MASP}]-\mathrm{kd02}_2\cdot[\mathrm{PC/CRP/LF/MASP}])\right)\\
\frac{\mathrm{d}\left([\mathrm{GlcNac/LF/CRP}]\right)}{\mathrm{d}t} &= +\left((\mathrm{ke01}_1\cdot[\mathrm{GlcNac/LF}]\cdot[\mathrm{CRP}]-\mathrm{ke01}_2\cdot[\mathrm{GlcNac/LF/CRP}])\right)\\
&\quad -\left((\mathrm{ke02}_1\cdot[\mathrm{GlcNac/LF/CRP}]\cdot[\mathrm{C1}]-\mathrm{ke02}_2\cdot[\mathrm{GlcNac/LF/CRP/C1}])\right)\\
&\quad -\left((\mathrm{kf02}_1\cdot[\mathrm{C4BP}]\cdot[\mathrm{GlcNac/LF/CRP}]-\mathrm{kf02}_2\cdot[\mathrm{C4BP/GlcNac/LF/CRP}])\right)\\
&\quad -\left((\mathrm{ke05}_1\cdot[\mathrm{GlcNac/LF/CRP}]\cdot[\mathrm{MASP}]-\mathrm{ke05}_2\cdot[\mathrm{GlcNac/LF/CRP/MASP}])\right)
\end{aligned}
$$

$$
\begin{aligned}
\frac{\mathrm{d}\left([\mathrm{GlcNac/LF/CRP/C1}]\right)}{\mathrm{d}t} &= +\left(\left(\mathrm{ke02}_1 \cdot [\mathrm{GlcNac/LF/CRP}] \cdot [\mathrm{C1}] - \mathrm{ke02}_2 \cdot [\mathrm{GlcNac/LF/CRP/C1}]\right)\right) \\
\frac{\mathrm{d}\left([\mathrm{C4BP}]\right)}{\mathrm{d}t} &= -\left(\left(\mathrm{kf01}_1 \cdot [\mathrm{C4BP}] \cdot [\mathrm{PC/CRP}] - \mathrm{kf01}_2 \cdot [\mathrm{C4BP/PC/CRP}]\right)\right) \\
&\quad -\left(\left(\mathrm{kf02}_1 \cdot [\mathrm{C4BP}] \cdot [\mathrm{GlcNac/LF/CRP}] - \mathrm{kf02}_2 \cdot [\mathrm{C4BP/GlcNac/LF/CRP}]\right)\right) \\
&\quad -\left(\left(\mathrm{kf04}_1 \cdot [\mathrm{C4BP}] \cdot [\mathrm{C4b}] - \mathrm{kf04}_2 \cdot [\mathrm{C4BP/C4b}]\right)\right) \\
&\quad -\left(\left(\mathrm{kf06}_1 \cdot [\mathrm{C4b/C2a}] \cdot [\mathrm{C4BP}] - \mathrm{kf06}_2 \cdot [\mathrm{C4b/C2a/C4BP}]\right)\right) \\
&\quad -\left(\left(\mathrm{kf07}_1 \cdot [\mathrm{dC4b/C2a}] \cdot [\mathrm{C4BP}] - \mathrm{kf07}_2 \cdot [\mathrm{dC4b/C2a/C4BP}]\right)\right) \\
&\quad -\left(\mathrm{k1}_{(\mathrm{tmp1})} \cdot [\mathrm{C4BP}]\right) \\
&\quad -\left(\left(\mathrm{k1}_{(\mathrm{tmp_f1})} \cdot [\mathrm{C4BP}] \cdot [\mathrm{PC/CRP/LF}] - \mathrm{k2}_{(\mathrm{tmp_f1})} \cdot [\mathrm{C4BP/PC/CRP/LF}]\right)\right) \\
\frac{\mathrm{d}\left([\mathrm{C4BP/PC/CRP}]\right)}{\mathrm{d}t} &= +\left(\left(\mathrm{kf01}_1 \cdot [\mathrm{C4BP}] \cdot [\mathrm{PC/CRP}] - \mathrm{kf01}_2 \cdot [\mathrm{C4BP/PC/CRP}]\right)\right) \\
\frac{\mathrm{d}\left([\mathrm{C4BP/GlcNac/LF/CRP}]\right)}{\mathrm{d}t} &= +\left(\left(\mathrm{kf02}_1 \cdot [\mathrm{C4BP}] \cdot [\mathrm{GlcNac/LF/CRP}] - \mathrm{kf02}_2 \cdot [\mathrm{C4BP/GlcNac/LF/CRP}]\right)\right) \\
\frac{\mathrm{d}\left([\mathrm{iC4b/C2a}]\right)}{\mathrm{d}t} &= +\left(\mathrm{kf03} \cdot [\mathrm{C4b/C2a}] \cdot [\mathrm{C4BP}]\right) \\
\frac{\mathrm{d}\left([\mathrm{C4BP/C4b}]\right)}{\mathrm{d}t} &= +\left(\left(\mathrm{kf04}_1 \cdot [\mathrm{C4BP}] \cdot [\mathrm{C4b}] - \mathrm{kf04}_2 \cdot [\mathrm{C4BP/C4b}]\right)\right) \\
\frac{\mathrm{d}\left([\mathrm{C4b/C2a/C4BP}]\right)}{\mathrm{d}t} &= +\left(\left(\mathrm{kf06}_1 \cdot [\mathrm{C4b/C2a}] \cdot [\mathrm{C4BP}] - \mathrm{kf06}_2 \cdot [\mathrm{C4b/C2a/C4BP}]\right)\right) \\
\frac{\mathrm{d}\left([\mathrm{dC4b/C2a/C4BP}]\right)}{\mathrm{d}t} &= +\left(\left(\mathrm{kf07}_1 \cdot [\mathrm{dC4b/C2a}] \cdot [\mathrm{C4BP}] - \mathrm{kf07}_2 \cdot [\mathrm{dC4b/C2a/C4BP}]\right)\right) \\
\frac{\mathrm{d}\left([\mathrm{PC/CRP/LF/C1}]\right)}{\mathrm{d}t} &= +\left(\left(\mathrm{kd05}_1 \cdot [\mathrm{PC/CRP/LF}] \cdot [\mathrm{C1}] - \mathrm{kd05}_2 \cdot [\mathrm{PC/CRP/LF/C1}]\right)\right) \\
&\quad -\left(\left(\mathrm{kd08}_1 \cdot [\mathrm{PC/CRP/LF/C1}] \cdot [\mathrm{MASP}] - \mathrm{kd08}_2 \cdot [\mathrm{PC/CRP/LF/C1/MASP}]\right)\right) \\
\frac{\mathrm{d}\left([\mathrm{C4BP/PC/CRP/LF}]\right)}{\mathrm{d}t} &= +\left(\left(\mathrm{k1}_{(\mathrm{tmp_f1})} \cdot [\mathrm{C4BP}] \cdot [\mathrm{PC/CRP/LF}] - \mathrm{k2}_{(\mathrm{tmp_f1})} \cdot [\mathrm{C4BP/PC/CRP/LF}]\right)\right) \\
\frac{\mathrm{d}\left([\mathrm{GlcNac/LF/CRP/MASP}]\right)}{\mathrm{d}t} &= +\left(\left(\mathrm{ke05}_1 \cdot [\mathrm{GlcNac/LF/CRP}] \cdot [\mathrm{MASP}] - \mathrm{ke05}_2 \cdot [\mathrm{GlcNac/LF/CRP/MASP}]\right)\right) \\
\frac{\mathrm{d}\left([\mathrm{PC/CRP/LF/C1/MASP}]\right)}{\mathrm{d}t} &= +\left(\left(\mathrm{kd08}_1 \cdot [\mathrm{PC/CRP/LF/C1}] \cdot [\mathrm{MASP}] - \mathrm{kd08}_2 \cdot [\mathrm{PC/CRP/LF/C1/MASP}]\right)\right) \\
&\quad +\left(\left(\mathrm{kd09}_1 \cdot [\mathrm{PC/CRP/LF/MASP}] \cdot [\mathrm{C1}] - \mathrm{kd09}_2 \cdot [\mathrm{PC/CRP/LF/C1/MASP}]\right)\right) \\
\frac{\mathrm{d}\left([\mathrm{GlcNac/HF}]\right)}{\mathrm{d}t} &= +\left(\left(\mathrm{kg01}_1 \cdot [\mathrm{X}] \cdot [\mathrm{HF}] - \mathrm{kg01}_2 \cdot [\mathrm{GlcNac/HF}]\right)\right) \\
&\quad -\left(\left(\mathrm{kg02}_1 \cdot [\mathrm{GlcNac/HF}] \cdot [\mathrm{MASP}] - \mathrm{kg02}_2 \cdot [\mathrm{GlcNac/HF/MASP}]\right)\right) \\
\frac{\mathrm{d}\left([\mathrm{HF}]\right)}{\mathrm{d}t} &= -\left(\left(\mathrm{kg01}_1 \cdot [\mathrm{X}] \cdot [\mathrm{HF}] - \mathrm{kg01}_2 \cdot [\mathrm{GlcNac/HF}]\right)\right) \\
\frac{\mathrm{d}\left([\mathrm{GlcNac/HF/MASP}]\right)}{\mathrm{d}t} &= +\left(\left(\mathrm{kg02}_1 \cdot [\mathrm{GlcNac/HF}] \cdot [\mathrm{MASP}] - \mathrm{kg02}_2 \cdot [\mathrm{GlcNac/HF/MASP}]\right)\right) \\
\frac{\mathrm{d}\left([\mathrm{X}]\right)}{\mathrm{d}t} &= -\left(\left(\mathrm{kg01}_1 \cdot [\mathrm{X}] \cdot [\mathrm{HF}] - \mathrm{kg01}_2 \cdot [\mathrm{GlcNac/HF}]\right)\right)
\end{aligned}
$$

A.2 Experimental Materials and Methods

Antibodies, proteins & sera

Human C1 complex protein was purchased from Sigma-Aldrich (St. Louis, MO). Human C4b-binding protein was from Complement Technology (Tyler, Texas). Goat anti-rabbit secondary antibody with HRP conjugation, polyclonal rabbit anti-C3d and anti-C4c antibodies were purchased from Dako A/S (Glostrup, Denmark). Secondary anti-sheep antibody was from Upstate (Lake Placid, NY). Rabbit anti-human C4BP antibody and mouse anti-human C4BP antibody targeting N-terminal part of C4BP were raised according to standard protocols. C4BP used as standard for ELISA was purified from human plasma (Zadura et al., 2009; Dahlback, 1983). Serum samples were obtained from healthy adults and infected patient volunteers with informed consent. As an infecion marker, the CRP levels in the serum samples were determined using the CRP Bioassay ELISA kit (BD Biosciences, San Jose, CA) to confirm the healthy and infectious status of the samples. All experiments were performed according to national and institutional guidelines on ethics and biosafety (Institutional Review Board, Reference Code: NUS-IRB 08-296).

Manipulation of C4BP level in the serum

The level of C4BP in the serum was increased by exogenously adding 100 μg purified C4BP protein per ml serum. The C4BP level in the serum was reduced by immunoprecipitation. One ml of serum was pre-cleared using 20 μl Protein G Sepharose (GE healthcare, Uppsala, Sweden) at 4°C for 1 hour with gentle shaking. Sheep polyclonal anti-C4BP antibody (GeneTex Inc, Irvine, CA) was incubated with the pre-cleared serum with gentle shaking at 4°C for 1 hour. Protein G Sepharose (20 mul) was then added to the serum containing the antibody-C4BP complex with gentle shaking at 4° C for 1 hour. The supernatant with reduced C4BP level was stored. For both treated

and untreated serum samples, C4BP level was measured by C4BP sandwich ELISA to ensure the successfully addition and depletion of C4BP (Figure A.3). 10% (v/v) healthy serum, which was used in the subsequent experiments, was prepared by diluting the serum from healthy adult, in TBS buffer (25 mM Tris, 145 mM NaCl, pH 7.4, 2.5 mM $CaCl_2$) and 10% (v/v) patient serum was prepared by diluting in MBS buffer (25 mM MES, 145 mM NaCl, pH 6.5, 2.0 mM $CaCl_2$).

C4BP quantification by sandwich ELISA

To compare the C4BP levels between treated and untreated sera, sandwich ELISA was performed. 10 μl/ml of rabbit anti-human C4BP antibody in 50 μl coating buffer (75 mM sodium carbonate, pH 9.6) was immobilized on 96-well Maxisorp plates (Nunc, Roskilde, Denmark) by incubating overnight at 4°C. After four washes with wash buffer (50 mM Tris-HCL, pH 8.0 supplemented with 2 mM $CaCl_2$, 0.15 M NaCl, 0.1% (v/v) Tween-20), the wells were blocked with blocking buffer (1% BSA (w/v) in TBS) at 37°C for 1 hour. Following four washes, treated and untreated sera were diluted 2000 times in blocking buffer and 50 μl was added to the wells and incubated at $37^{c}irc$C for 1 hour. After four washes, C4BP protein amount was detected with mouse anti-C4BP antibody (1 : 15000) followed by rabbit anti-mouse HRP-conjugated secondary antibody (1 : 2000). ABTS substrate (Roche Diagnostics, Mannheim, Germany) was added and the OD_{405nm} was read. Wells incubated with blocking buffer instead of serum served as a negative control.

Complement measurement by pull-down with GlcNAc- and PC- beads

Untreated serum or sera with increased or decreased C4BP from both healthy adults and patients were challenged with GlcNAc-Sepharose (Sigma-Aldrich) to initiate L-ficolin-mediated complement activation. 20 μl of GlcNAc beads was added to 500 μl of 10% serum. The beads were collected between 0.5 to 4.0 hours at intervals of

0.5 hour. For patient's serum, the beads also underwent incubation at shorter time intervals of 0, 10 and 20 minutes. For CRP-mediated pathway, PC-Sepharose (Pierce, Rockford, IL) was used in place of GlcNAc-Sepharose. Beads were washed thrice with their corresponding incubation buffer and boiled in SDS-PAGE sample buffer.

Western blot

Protein samples of the different time points obtained from the previous step was electrophoresed on 12% SDS-PAGE. The primary antibodies used were polyclonal sheep anti-C4BP, polyclonal rabbit anti-C4c and polyclonal rabbit anti-C3d at dilutions of 1 : 1000. Secondary antibodies used were rabbit anti-sheep and goat anti-rabbit at dilutions of 1 : 15000 and 1 : 2000 respectively. The fractionated proteins were transferred to PVDF membranes (Bio-Rad). Membrane blots were incubated in blocking buffer (3% skimmed milk (w/v) in TBS) overnight at $4^{c}irc$C. Primary antibodies were diluted in TBS supplemented with 3% (w/v) BSA, 0.5% (v/v) Tween-20 and reacted with the blots with gentle shaking for 2 hours at room temperature. After washing 4× for 15 minutes each with wash buffer (TBS supplemented with 0.5% (v/v) Tween-20), the blots were incubated with HRP conjugated secondary antibodies with gentle shaking for 2 hour at room temperature. Visualization was performed with the use of Super-Signal West Pico Chemiluminescent Substrate from Thermo Scientific (Rockford, IL) and exposed through X-ray. Densitometric analysis of the blots was performed using GS-800 Calibrated Densitometer (Bio-Rad). Fixed amount of pure proteins were used as positive controls and the amounts of protein on different gels were normalized to the positive control and compared with each other. Data are representative of three independent experiments.

A.3 Experimental Data

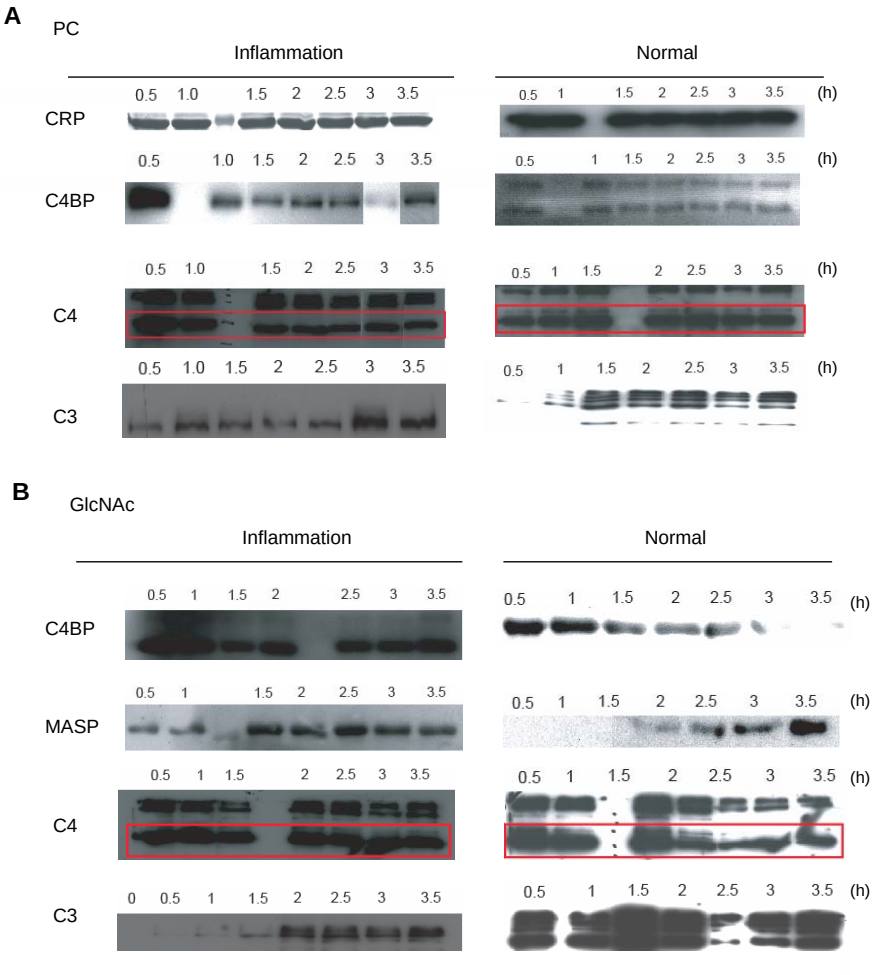


Figure A.1: Time serials experimental data under inflammation and normal conditions. (A) PC-initiated complement activation, (B) GlcNAc-initiated complement activation.

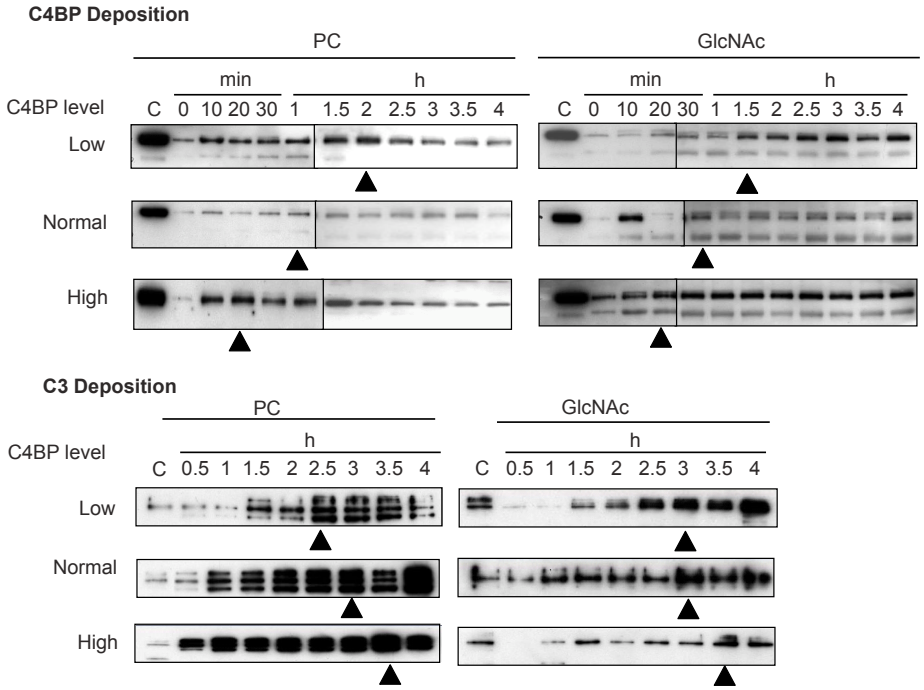


Figure A.2: Experimental verification of effects of C4BP under infection-inflammation condition. Profiles of deposited C4BP or C3 across time points of $0-4$ hours under infection-inflammation condition via classical pathway (triggered by PC beads) or lectin pathway (triggered by GlcNAc beads) in untreated or treated sera with increased C4BP or decreased C4BP, were studied. The deposited protein was resolved in 12% reducing SDS PAGE and detected using polyclonal sheep anti-C4BP. Same amount of pure protein was loaded to each of the gels as the positive control (labeled as "C" in the image). The black triangles point to the peaks of the time serials data.

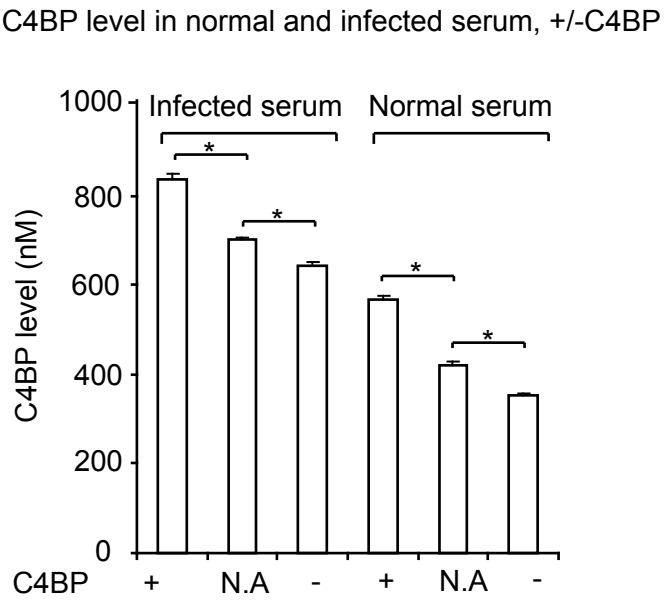


Figure A.3: C4BP levels measured by C4BP sandwich ELISA for both treated and untreated serum samples.

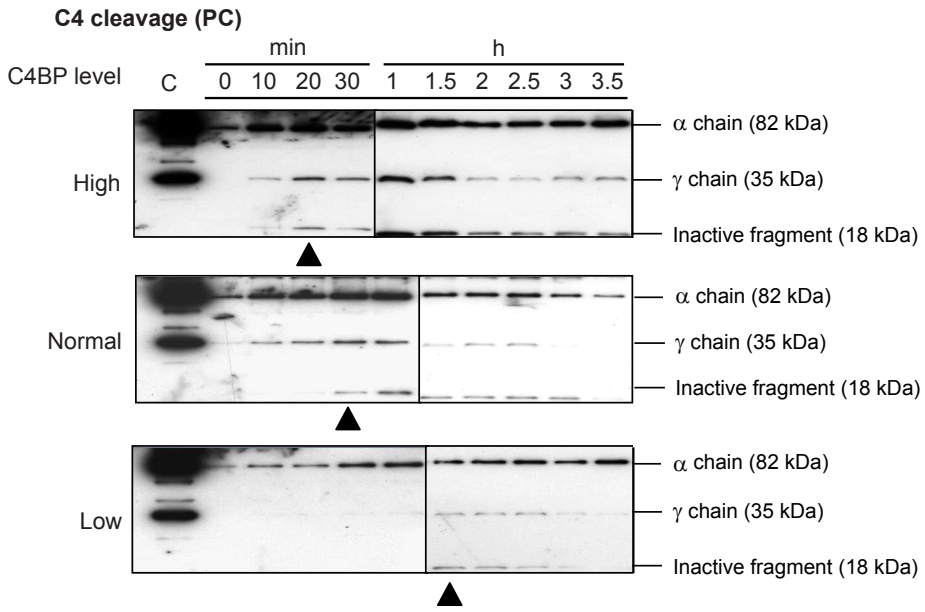


Figure A.4: (Experimental verification of the role of C4BP. Profiles of deposited cleaved/uncleaved C4 fragments across time points of $0 - 3.5$ hours under infection-inflammation condition occurring via classical pathway (triggered by PC beads) in untreated or treated sera with increased C4BP or decreased C4BP were studied. The black triangles point to the first appearance of inactive fragments.

Bibliography

Aldridge, B. B., Burke, J. M., Lauffenburger, D. A., and Sorger, P. K. (2006). Physicochemical modelling of cell signalling pathways. *Nature Cell Biology*, 8:1195–1203.

Ammann, H. (1990). *Ordinary Differential Equations: An Introduction to Nonlinear Analysis*. Walter de Gruyter.

Anderson, D. H., Radeke, M. J., Gallo, N. B., Chapin, E. A., Johnson, P. T., Curletti, C. R., Hancox, L. S., Hu, J., Ebright, J. N., Malek, G., Hauser, M. A., Rickman, C. B., Bok, D., Hageman, G. S., and Johnson, L. V. (2010). The pivotal role of the complement system in aging and age-related macular degeneration: hypothesis re-visited. *Prog Retin Eye Res*, 29(2):95–112.

Andrianantoandro, E., Basu, S., Karig, D. K., and Weiss, R. (2006). Synthetic biology: new engineering rules for an emerging discipline. *Mol Syst Biol*, 2:2006.0028.

Arkin, A., Ross, J., and McAdams, H. H. (1998). Stochastic kinetic analysis of developmental pathway bifurcation in phage lambda-infected escherichia coli cells. *Genetics*, 149(4):1633–48.

Aziz, A., Sanwal, K., Singhal, V., and Brayton, R. (2000). Model checking continuous time markov chains. *ACM Transactions on Computational Logics*, 1(1):162–170.

Babu, C. S., Yoon, S., Nam, H.-S., and Yoo, Y. (2004). Simulation and sensitivity anal-

ysis of phosphorylation of EGFR signal transduction pathway in PC12 cell model. *IEE Systems Biology*, 1(2):213–221.

Back, T., Fogel, D., and Michalewicz, Z. (1997). *Handbook of evolutionary computation*. Oxford University Press.

Banga, J. R. (2008). Optimization in computational systems biology. *BMC Systems Biology*, 2(47):1–7.

Barnum, S. R. and Dahlback, B. (1990). C4b-binding protein, a regulatory component of the classical pathway of complement, is an acute-phase protein and is elevated in systemic lupus erythematosus. *Complement Inflamm*, 7(2):71–77.

Basak, S., Kim, H., Kearns, J. D., Tergaonkar, V., O'Dea, E., Werner, S. L., Benedict, C. A., Ware, C. F., Ghosh, G., Verma, I. M., and Hoffmann, A. (2007). A fourth ikappab protein within the nf-kappab signaling module. *Cell*, 128(2):369–81.

Bell-Pedersen, D., Cassone, V. M., Earnest, D. J., Golden, S. S., Hardin, P. E., Thomas, T. L., and Zoran, M. J. (2005). Circadian rhythms from multiple oscillators: lessons from diverse organisms. *Nat Rev Genet*, 6(7):544–56.

Bentele, M., Lavrik, I., Ulrich, M., Stober, S., Heermann, D., Kalthoff, H., Krammer, P., and Eils, R. (2004). Mathematical modeling reveals threshold mechanism in CD95-induced apoptosis. *The Journal of Cell Biology*, 166(6):839–851.

Berggard, K., Johnsson, E., Mooi, F. R., and Lindahl, G. (1997). Bordetella pertussis binds the human complement regulator c4bp: role of filamentous hemagglutinin. *Infect Immun*, 65(9):3638–3643.

Birtwistle, M. R., Hatakeyama, M., Yumoto, N., Ogunnaike, B. A., Hoek, J. B., and Kholodenko, B. N. (3). Ligand-dependent responses of the ErbB signaling network: experimental and modeling analyses. *Molecular Systems Biology*, 144:1–16.

Blom, A. M., Kask, L., and Dahlback, B. (2001). Structural requirements for the complement regulatory activities of c4bp. *J Biol Chem*, 276(29):27136–27144.

Blom, A. M., Nandakumar, K. S., and Holmdahl, R. (2009). C4b-binding protein (c4bp) inhibits development of experimental arthritis in mice. *Ann Rheum Dis*, 68(1):136–142.

Blom, A. M., Webb, J., Villoutreix, B. O., and Dahlback, B. (1999). A cluster of positively charged amino acids in the c4bp alpha-chain is crucial for c4b binding and factor i cofactor function. *J Biol Chem*, 274(27):19237–19245.

Boerger, L. M., Morris, P. C., Thurnau, G. R., Esmon, C. T., and Comp, P. C. (1987). Oral contraceptives and gender affect protein s status. *Blood*, 69(2):692–694.

Bonzanni, N., Krepska, E., Feenstra, K. A., Fokkink, W., Kielmann, T., Bal, H., and Heringa, J. (2009). Executing multicellular differentiation: quantitative predictive modelling of c.elegans vulval development. *Bioinformatics*, 25(16):2049–56.

Boyen, X. and Koller, D. (1998). Tractable inference for complex stochastic processes. In *Proceedings of the 14th Conference on Uncertainty in Artificial Intelligence—UAI 1998*, pages 33–42. San Francisco: Morgan Kaufmann.

Boyen, X. and Koller, D. (1999). Approximate learning of dynamic models. In Kearns, M. S., Solla, S. A., and Kohn, D. A., editors, *Advances in Neural Information Processing Systems 11: Proceedings of the 1998 Conference—NIPS 1998*, pages 396–402. Cambridge: MIT Press.

Bozga, M. and Maler, O. (1999). On the representation of probabilities over structured domains. In Halbwachs, N. and Peled, D., editors, *Computer Aided Verification*, volume 1633 of *Lecture Notes in Computer Science*, pages 682–682. Springer Berlin / Heidelberg.

Brown, K. S., Hill, C. C., Calero, G. A., Lee, K. H., Sethna, J. P., and Cerione, R. A. (2004). The statistical mechanics of complex signaling networks: nerve growth factor signaling. *Physical Biology*, 1:184–195.

Bryant, V. (1985). *Metric Spaces: Iteration and Application*. Cambridge University Press.

Burgard, A. P., Pharkya, P., and Maranas, C. D. (2003). Optknock: a bilevel programming framework for identifying gene knockout strategies for microbial strain optimization. *Biotechnol Bioeng*, 84(6):647–57.

Calder, M., Duguid, A., Gilmore, S., and Hillston, J. (2006a). Stronger computational modelling of signalling pathways using both continuous and discrete-state methods. In Priami, C., editor, *CMSB*, volume 4210 of *Lecture Notes in Computer Science*, pages 63–77. Springer.

Calder, M., Gilmore, S., and Hillston, J. (2006b). Modelling the influence of RKIP on the ERK signalling pathway using the stochastic process algebra PEPA. *Transactions on Computational Systems Biology VII*, 4230:1–23.

Calder, M., Vyshemirsky, V., Gilbert, D., and Orton, R. (2005). Analysis of signalling pathways using the PRISM model checker. In *Proceeding of Computational Methods in Systems Biology (CMSB'05)*, pages 179–190.

Calder, M., Vyshemirsky, V., Gilbert, D., and Orton, R. (2006c). Analysis of signalling pathways using continuous time Markov chains. *Transactions on Computational Systems Biology VI*, 4220:44–67.

Cascante, M., Boros, L. G., Comin-Anduix, B., de Atauri, P., Centelles, J. J., and Lee, P. W.-N. (2002). Metabolic control analysis in drug discovery and disease. *Nat Biotechnol*, 20(3):243–9.

Chang, Y. and Sahinidis, N. V. (2005). Optimization of metabolic pathways under stability considerations. *Computers & Chemical Engineering*, 29(3):467–479.

Chattopdhyay, B. (2010). Accelerating systems biology computations using graphical processors. Bachelor's Thesis, National University of Singapore.

Chen, M. and Hofestaedt, R. (2003). Quantitative Petri net model of gene regulated metabolic networks in the cell. *In Silico Biology*, 3(3):347–365.

Chen, W. W., Schoeberl, B., Jasper, P. J., Niepel, M., Nielsen, U. B., Lauffenburger, D. A., and Sorger, P. K. (2009). Input-output behavior of ErbB signaling pathways as revealed by a mass action model trained against dynamic data. *Mol Syst Biol*, 5:239.

Cho, K. H., Shin, S. Y., Kolch, W., and Wolkenhauer, O. (2003). Experimental design in systems biology, based on parameter sensitivity analysis using a Monte Carlo method: A case study for the TNFα-mediated NF-κB signal transduction pathway. *Simulation*, 79(12):726–739.

Ciocchetta, F., Degasperi, A., Hillston, J., and Calder, M. (2009). CTMC with levels models for biochemical systems. Preprint submitted to Elsevier.

Clarke, E. M., Faeder, J. R., Langmead, C. J., Harris, L. A., Jha, S. K., and Legay, A. (2008). Statistical model checking in BioLab: Applications to the automated analysis of T-Cell receptor signaling pathway. In Heiner, M. and Uhrmacher, A. M., editors, *CMSB*, volume 5307 of *Lecture Notes in Computer Science*, pages 231–250. Springer.

Dahlback, B. (1983). Purification of human c4b-binding protein and formation of its complex with vitamin k-dependent protein s. *Biochem J*, 209(3):847–856.

Danos, V., Feret, J., Fontana, W., Harmer, R., and Krivine, J. (2007). Rule-based modelling of cellular signalling. In Caires, L. and Vasconcelos, V. T., editors, *CONCUR*, volume 4703 of *Lecture Notes in Computer Science*, pages 17–41. Springer.

David, R. and Alla, H. (1987). Continuous petri nets. In *Proceeding of 8th European Workshop on Application and Theory of Petri Nets*, pages 275–294, Zaragoza, Spain.

de Jong, H. (2002). Modeling and simulation of genetic regulatory systems: a literature review. *J Comput Biol*, 9(1):67–103.

de Pillis, L. G., Radunskaya, A. E., and Wiseman, C. L. (2005). A validated mathematical model of cell-mediated immune response to tumor growth. *Cancer Res*, 65(17):7950–8.

Dequeant, M.-L., Glynn, E., Gaudenz, K., Wahl, M., Chen, J., Musheqian, A., and Pourquie, O. (2006). A complex oscillating network of signaling genes underlies the mouse segmentation clock. *Science*, 314:1595–1598.

Do, J. H., Nagasaki, M., and Miyano, S. (2010). The systems approach to the prespore-specific activation of sigma factor sigf in bacillus subtilis. *Biosystems*, 100(3):178–84.

Doi, A., Nagasaki, M., Fujita, S., Matsuno, H., and Miyano, S. (2003). Genomic Object Net: II. Modeling biopathways by hybrid functional Petri net with extension. *Applied Bioinformatics*, 2(3):185–188.

Draper, N. R. and Smith, H. (1981). *Applied regression analysis*. Wiley series in probability and mathematical statistics. Wiley, New York, 2d ed edition.

Durrett, R. (2004). *Probability: Theory and Examples*. Duxbury Press.

Egan, L. J. and Toruner, M. (2006). Nf-kappab signaling: pros and cons of altering nf-kappab as a therapeutic approach. *Ann N Y Acad Sci*, 1072:114–22.

El-Samad, H., Kurata, H., Doyle, J. C., Gross, C. A., and Khammash, M. (2005). Surviving heat shock: control strategies for robustness and performance. *Proc Natl Acad Sci U S A*, 102(8):2736–41.

Feldman, J. (2008). Review of measurable functions. University of British Columbia.

Feng, X.-j. and Rabitz, H. (2004). Optimal identification of biochemical reaction networks. *Biophys J*, 86(3):1270–81.

Fisher, J., Piterman, N., Hajnal, A., and Henzinger, T. (2007). Predictive modeling of signaling croostalk during C. elegans vulval development. *PLoS Computational Biology*, pages 92–106.

Fogel, D., Fogel, L., and Atmar, J. (1992). Meta-evolutionary programming. In *25th Asiloma Conference on Signals, Systems and Computers.*, pages 540–545, Asilomar. IEEE Computer Society,.

Fomekong-Nanfack, Y., Kaandorp, J. A., and Blom, J. (2007). Efficient parameter estimation for spatio-temporal models of pattern formation: case study of Drosophila melanogaster. *Bioinformatics*, 23(24):3356–3363.

Fujita, T., Matsushita, M., and Endo, Y. (2004). The lectin-complement pathway–its role in innate immunity and evolution. *Immunol Rev*, 198:185–202.

Gadkar, K. G., Doyle Iii, F. J., Edwards, J. S., and Mahadevan, R. (2005a). Estimating optimal profiles of genetic alterations using constraint-based models. *Biotechnol Bioeng*, 89(2):243–51.

Gadkar, K. G., Gunawan, R., and III, F. J. D. (2005b). Iterative approach to model identification of biological networks. *BMC Bioinformatics*, 6:155.

Gallego, M., Eide, E. J., Woolf, M. F., Virshup, D. M., and Forger, D. B. (2006).

An opposite role for tau in circadian rhythms revealed by mathematical modeling. *PNAS*, 103(28):10618–10623.

Geisweiller, N., Hillston, J., and Stenico, M. (2008). Relating continuous and discrete PEPA models of signalling pathways. *Theoretical Computer Science*, 404(2):97–111.

Gigli, I. (1979). Prosser white oration 1978. the complement system in inflammation and host defence. *Clin Exp Dermatol*, 4(3):271–289.

Gillespie, D. (1977). Exact stochastic simulation of coupled chemical reactions. *Journal of Physical Chemistry*, 81(25):2340–2361.

Girolami, M. (2008). Bayesian inference for differential equations. *Theor. Comput. Sci.*, 408(1):4–16.

Goldbeter, A. and Pourquie, O. (2008). Modeling the segmentation clock as a network of coupled oscillations in the notch, wnt and fgf signaling pathways. *Journal of Theoretical Biology*, 252:574–585.

Griffin, J. H., Gruber, A., and Fernandez, J. A. (1992). Reevaluation of total, free, and bound protein s and c4b-binding protein levels in plasma anticoagulated with citrate or hirudin. *Blood*, 79(12):3203–3211.

Guldberg, C. M. and Waage, P. (1879). Über die chemische affinitat. *Prakt. Chem.*, 19:69.

Gunawan, R., Cao, Y., Petzold, L., and Doyle, 3rd, F. J. (2005). Sensitivity analysis of discrete stochastic systems. *Biophys J*, 88(4):2530–40.

Gunawan, R. and Doyle, 3rd, F. J. (2006). Isochron-based phase response analysis of circadian rhythms. *Biophys J*, 91(6):2131–41.

Gutenkunst, R. N., Waterfall, J. J., Casey, F. P., Brown, K. S., Myers, C. R., and Sethna, J. P. (2007). Universally sloppy parameter sensitivities in systems biology. *PLoS Computational Biology*, 3(10):189.

Hansson, H. and Jonsson, B. (1994). A logic for reasoning about time and reliability. *Formal Asp. Comput.*, 6(5):512–535.

Heath, J., Kwiatkowska, M., Norman, G., Parker, D., and Tymchyshyn, O. (2008). Probabilistic model checking of complex biological pathways. *Theoretical Computer Science*, 319(3):239–257.

Heinemann, M. and Panke, S. (2006). Synthetic biology–putting engineering into biology. *Bioinformatics*, 22(22):2790–9.

Heiner, M., Koch, I., and Will, J. (2003). Model validation of biological pathways using Petri nets - demonstrated for apoptosis. In Priami, C., editor, *CMSB*, volume 2602 of *Lecture Notes in Computer Science*, page 173. Springer.

Helikar, T., Konvalina, J., Heidel, J., and Rogers, J. A. (2008). Emergent decision-making in biological signal transduction networks. *PNAS*, 105(6):1913–1918.

Hillston, J. (1996). *A compositional approach to performance modelling*. University Press.

Hindmarsh, A. C. (1983). ODEPACK, a systematized collection of ODE solvers. *Scientific Computing*, pages 55–64.

Hirsch, M. W., Smale, S., and Devaney, R. L. (2004). *Differential Equations, Dynamical Systems and In Introduction to Chaos*. Elsevier, 2 edition.

Hoffmann, A., Levchenko, A., Scott, M. L., and Baltimore, D. (2002). The IκB-NF-κB signaling module: Temporal control and selective gene activation. *Science*, 298:1241–1245.

Hooke, R. and Jeeves, T. A. (1961). "Direct search" solution of numerical and statistical problems. *Journal of the Association for Computing Machinery*, 8:212–229.

Hoops, S., Sahle, S., Gauges, R., Lee, C., Pahle, J., Simus, N., Singhal, M., Xu, L., Mendes, P., and Kummer, U. (2006a). COPASI - a COmplex PAthway SImulator. *Bioinformatics*, 22(24):3067–3074.

Hoops, S., Sahle, S., Gauges, R., Lee, C., Pahle, J., Simus, N., Singhal, M., Xu, L., Mendes, P., and Kummer, U. (2006b). Copasi–a complex pathway simulator. *Bioinformatics*, 22(24):3067–3074.

Hucka, M., Finney, A., Sauro, H. M., Bolouri, H., Doyle, J. C., Kitano, H., Arkin, A. P., Bornstein, B. J., Bray, D., Cornish-Bowden, A., Cuellar, A. A., Dronov, S., Gilles, E. D., Ginkel, M., Gor, V., Goryanin, I. I., Hedley, W. J., Hodgman, T. C., Hofmeyr, J.-H., Hunter, P. J., Juty, N. S., Kasberger, J. L., Kremling, A., Kummer, U., Novre, N. L., Loew, L. M., Lucio, D., Mendes, P., Minch, E., Mjolsness, E. D., Nakayama, Y., Nelson, M. R., Nielsen, P. F., Sakurada, T., Schaff, J. C., Shapiro, B. E., Shimizu, T. S., Spence, H. D., Stelling, J., Takahashi, K., Tomita, M., Wagner, J., Wang, J., and Forum, S. B. M. L. (2003). The systems biology markup language (sbml): a medium for representation and exchange of biochemical network models. *Bioinformatics*, 19(4):524–531.

Jaeger, M. (2004). Probabilistic decision graphs c combining verification and ai techniques for probabilistic inference. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 12(Supplementary Issue 1):19–42.

James, A. B., Monreal, J. A., Nimmo, G. A., Kelly, C. L., Herzyk, P., Jenkins, G. I., and Nimmo, H. G. (2008). The circadian clock in Arbidopsis roots is a simplified slave version of the clock in shoots. *Science*, 322(5909):1832–1835.

Jha, S., Clarke, E., Langmead, C., Legay, A., Platzer, A., and Zuliani, P. (2009). A

bayesian approach to model checking biological systems. In Degano, P. and Gorrieri, R., editors, *Computational Methods in Systems Biology*, volume 5688 of *Lecture Notes in Computer Science*, pages 218–234. Springer Berlin / Heidelberg.

Ji, X. and Xu, Y. (2006). libsres: a c library for stochastic ranking evolution strategy for parameter estimation. *Bioinformatics*, 22(1):124–126.

Joshi-Tope, G., Gillespie, M., Vastrik, I., D'Eustachio, P., Schmidt, E., de Bono, B., Jassal, B., Gopinath, G. R., Wu, G. R., Matthews, L., Lewis, S., Birney, E., and Stein, L. (2005). Reactome: a knowledgebase of biological pathways. *Nucleic Acids Res*, 33(Database issue):D428–32.

Keasling, J. D. and Chou, H. (2008). Metabolic engineering delivers next-generation biofuels. *Nat Biotechnol*, 26(3):298–9.

Kennedy, J. and Eberhart, R. (1995). Particle Swarm Optimization. In *Proceedings of the Fourth IEEE International Conference on Neural Networks*, pages 1942 – 1948, Perth, Australia.

Kholodenko, B. N. (2007). Untangling the signalling wires. *Nature Cell Biology*, 9(3):247–249.

Khosla, C. and Keasling, J. D. (2003). Metabolic engineering for drug discovery and development. *Nat Rev Drug Discov*, 2(12):1019–25.

Kirjavainen, V., Jarva, H., Biedzka-Sarek, M., Blom, A. M., Skurnik, M., and Meri, S. (2008). Yersinia enterocolitica serum resistance proteins yada and ail bind the complement regulator c4b-binding protein. *PLoS Pathog*, 4(8):e1000140.

Kleinstein, S. H., Bottino, D., and Lett, G. S. (2006). Nonunifrom sampling for global optimization fo kinetic rate constants in biological pathways. In *Proceedings of the 2006 Winter Simulation Conference (IEEE)*, pages 1161–1166.

Klipp, E., Herwig, R., Kowald, A., Wierling, C., and Lehrach, H. (2005). *Systems Biology in Practice: Concepts, Implementation and Application*. Wiley-VCH.

Koch, I., Junker, B. H., and Heiner, M. (2005). Application of petri net theory for modelling and validation of the sucrose breakdown pathway in the potato tuber. *Bioinformatics*, 21(7):1219–1226.

Koch, I., Reisg, W., and Schreiber, F., editors (2010). *Modeling in systems biology: the Petri net approach*, volume 16 of *Computational biology*. Springer, New York, 1st. ed edition.

Koh, C. H., Nagasaki, M., Saito, A., Wong, L., and Miyano, S. (2010a). Da 1.0: parameter estimation of biological pathways using data assimilation approach. *Bioinformatics*, 26(14):1794–6.

Koh, G., Hsu, D., and Thiagarajan, P. S. (2010b). Incremental signaling pathway modeling by data integration. In Berger, B., editor, *RECOMB*, volume 6044 of *Lecture Notes in Computer Science*, pages 281–296. Springer.

Koh, G., Tucker-Kellogg, L., Hsu, D., and Thiagarajan, P. (2007). Globally consistent pathway parameter estimates through belief propagation. In *Proceeding of the Seventh Workshop on Algorithms in Bioinformatics (WABI)*, pages 420–430, Philadelphia.

Koh, Y. N., Teong, H. F., Hsu, D., Clement, M.-V., and Thiagarajan, P. (2005). Computational Modeling of the AKT Pathway. Unpublished.

Kwiatkowska, M. Z. and Heath, J. K. (2009). Biological pathways as communicating computer systems. *J Cell Sci*, 122(Pt 16):2793–800.

Kwiatkowska, M. Z., Norman, G., and Parker, D. (2002). PRISM: Probabilistic symbolic model checker. In Field, T., Harrison, P. G., Bradley, J. T., and Harder, U.,

editors, *Computer Performance Evaluation / TOOLS*, volume 2324 of *Lecture Notes in Computer Science*, pages 200–204. Springer.

Kwiatkowska, M. Z., Norman, G., and Parker, D. (2007). Stochastic model checking. In Bernardo, M. and Hillston, J., editors, *SFM*, volume 4486 of *Lecture Notes in Computer Science*, pages 220–270. Springer.

Kwiatkowska, M. Z., Norman, G., Parker, D., Tymchyshyn, O., Heath, J., and Gaffney, E. (2006). Simulation and verification for computational modelling of signalling pathways. In Perrone, L. F., Lawson, B., Liu, J., and Wieland, F. P., editors, *Winter Simulation Conference*, pages 1666–1674. WSC.

Langmead, C., Jha, S., and Clarke, E. (2006a). Temporal logics as query languages for dynamic Bayesian networks: application to D. Melanogaster embryo development. Technical report, Carnegie Mellon University.

Langmead, C. J., Jha, S., and Clarke, E. M. (2006b). Temporal-logics as query languages for dynamic bayesian networks: Application to d. melanogaster embryo development. Technical report, Carnegie Mellon University.

Lauffenburger, D. A. (2000). Cell signaling pathways as control modules: Complexity for simplicity? *PNAS*, 97(10):5031–5033.

Le Novere, N., Bornstein, B., Broicher, A., Courtot, M., Donizelli, M., Dharuri, H., Li, L., Sauro, H., Schilstra, M., Shapiro, B., Snoep, J., and Hucka, M. (2006). BioModels Database: A free, centralized database of curated, published, quantitative kinetic models of biochemical and cellular systems. *Nucleic Acids Research*, 34:D689–D691.

Lebiedz, D. (2005). Exploiting optimal control for target-oriented manipulationof (bio)chemical systems: A model-based approach to specific modification of self-organized dynamics. *International Journal of Modern Physics B*, 19(25):3763–3798.

Lee, D.-Y., Zimmer, R., Lee, S. Y., and Park, S. (2006). Colored Petri net modeling and simulation of signal transduction pathways. *Metabolic Engineering*, 8:112–122.

Lee, K. H., Park, J. H., Kim, T. Y., Kim, H. U., and Lee, S. Y. (2007). Systems metabolic engineering of escherichia coli for l-threonine production. *Mol Syst Biol*, 3:149.

Legewie, S., Bluthgen, N., and Herzel, H. (2006). Mathematical modeling identifies inhibitors of apoptosis as mediators of positive feedback and bistability. *PLoS Computational Biology*, 2(9):120–133.

Levenberg, K. (2). A method for the solution of certain nonlinear problems in least squares. *Quart. Appl. Math.*, 1994:164–168.

Li, C., Donizelli, M., Rodriguez, N., Dharuri, H., Endler, L., Chelliah, V., Li, L., He, E., Henry, A., Stefan, M. I., Snoep, J. L., Hucka, M., Novre, N. L., and Laibe, C. (2010). BioModels Database: An enhanced, curated and annotated resource for published quantitative kinetic models. *BMC Systems Biology*, 4(92):1–14.

Li, C., Nagasaki, M., Ueno, K., and Miyano, S. (2009). Simulation-based model checking approach to cell fate specification during caenorhabditis elegans vulval development by hybrid functional petri net with extension. *BMC Syst Biol*, 3:42.

Liu, B., Zhang, J., Tan, P. Y., Hsu, D., Blom, A. M., Sethi, S., Ho, B., Ding, J. L., and Thiagarajan, P. S. (2010). A computational and experimental study of the regulatory mechanisms of the complement system. PLoS Computational Biology, accepted.

Lodish, H. F. (2003). *Molecular cell biology*. W.H. Freeman and Company, New York, 5th ed edition.

Logan, C. Y. and Nusse, R. (2004). The wnt signaling pathway in development and disease. *Annu Rev Cell Dev Biol*, 20:781–810.

Lüdtke, N., Panzeri, S., Brown, M., Broomhead, D. S., Knowles, J., Montemurro, M. A., and Kell, D. B. (2008). Information-theoretic sensitivity analysis: a general method for credit assignment in complex networks. *J R Soc Interface*, 5(19):223–35.

Marlovits, G., Tyson, C. J., Novak, B., and Tyson, J. J. (1998). Modeling M-phase control in Xenopus oocyte extracts: the surveillance mechanism for unreplicated DNA. *Biophysical Chemistry*, 72:169–184.

Marnell, L., Mold, C., and Du Clos, T. W. (2005). C-reactive protein: ligands, receptors and role in inflammation. *Clin Immunol*, 117(2):104–111.

Marquardt, D. (1963). An algorithm for least squares estimation of nonlinear parameters. *SIAM Journal*, 11:431–441.

Materi, W. and Wishart, D. S. (2007). Computational Systems Biology in Drug Discovery and Development: Methods and Application. *Drug Discovery Today*, 12(7/8):295–303.

Matsuno, H., Tanaka, Y., Aoshima, H., Doi, A., Matsui, M., and Miyano, S. (2003a). Biopathways representation and simulation on hybrid functional petri net. *In Silico Biol*, 3(3):389–404.

Matsuno, H., Tanaka, Y., Aoshima, H., Doi, A., Matsui, M., and Miyano, S. (2003b). Biopathways representation and simulation on hybrid functional Petri net. *In Silico Biology*, 3(3):389–404.

Matsuno, H., Tanaka, Y., Aoshima1, H., Doi, A., Matsui, M., and Miyano, S. (2003c). Biopathways representation and simulation on hybrid functional petri net. *In Silico Biology*, 3:0032.

McKay, M., Beckman, R., and Conover, W. (2000). A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics*, 42(1):55–61.

Meri, T., Blom, A. M., Hartmann, A., Lenk, D., Meri, S., and Zipfel, P. F. (2004). The hyphal and yeast forms of candida albicans bind the complement regulator c4b-binding protein. *Infect Immun*, 72(11):6633–6641.

Mitchell, M. (1995). *An Introduction to Genetic Algorithms*. MIT Press.

Mold, C., Nakayama, S., Holzer, T. J., Gewurz, H., and Du Clos, T. W. (1981). C-reactive protein is protective against streptococcus pneumoniae infection in mice. *J Exp Med*, 154(5):1703–1708.

Moles, C. G., Mendes, P., and Banga, J. R. (2003). Parameter estimation in biochemical pathways: A comparison of global optimization methods. *Genome Research*, 13:2467–2474.

Murphy, K. P. (2002). *Dynamic Bayesian Networks: Representation, Inference and Learning*. PhD thesis, University of California, Berkeley.

Murphy, K. P. and Weiss, Y. (2001). The factored frontier algorithm for approximate inference in DBNs. In *Proceedings of the 17th Conference in Uncertainty in Artificial Intelligence*, pages 378–385, San Francisco, CA, USA.

Nagasaki, M., Saito, A., Jeong, E., Li, C., Kojima, K., Ikeda, Y., and Miyano, S. (2010). Cell illustrator 4.0: A computational platform for systems biology. *In Silico Biol*, 10:0002.

Ng, P. M., Le Saux, A., Lee, C. M., Tan, N. S., Lu, J., Thiel, S., Ho, B., and Ding, J. L. (2007). C-reactive protein collaborates with plasma lectins to boost immune response against bacteria. *EMBO J*, 26(14):3431–3440.

Nodelman, U., Shelton, C. R., and Koller, D. (2002). Continuous time Bayesian networks. In *Proceedings of the 18th Conference in Uncertainty in Artificial Intelligence (UAI'02)*, pages 378–387.

Nordstrom, T., Blom, A. M., Forsgren, A., and Riesbeck, K. (2004). The emerging pathogen moraxella catarrhalis interacts with complement inhibitor c4b binding protein through ubiquitous surface proteins a1 and a2. *J Immunol*, 173(7):4598–4606.

Norris, J. R. (1997). *Markov Chains*. Cambridge University Press.

Nunez, L. M. (1989). On the relationship between temporal Bayes networks and Markov chains. Master's thesis, Brown University.

Okroj, M., Heinegard, D., Holmdahl, R., and Blom, A. M. (2007). Rheumatoid arthritis and the complement system. *Ann Med*, 39(7):517–530.

Papin, J. and Palsson, B. (2004). Topological analysis of mass-balanced signaling networks: a framework to obtain network properties including crosstalk. *Journal of Theoretical Biology*, 227(2):283–297.

Papoutsakis, E. T. (1984). Equations and calculations for fermentations of butyric acid bacteria. *Biotechnol Bioeng*, 26(2):174–87.

Park, J. H. and Lee, S. Y. (2010). Metabolic pathways and fermentative production of l-aspartate family amino acids. *Biotechnol J*, 5(6):560–77.

Petri, C. A. (1962). *Kommunikation mit automaten*. PhD thesis, University of Bonn.

Petzold, L. (1983). Automatic selection of methods for solving stiff and nonstiff systems of ordinary differential equations. *SIAM Journal on Scientific and Statistical Computing*, 4:136–148.

Pnueli, A. (1977). The temporal logic of programs. In *FOCS*, pages 46–57. IEEE.

Prasadarao, N. V., Blom, A. M., Villoutreix, B. O., and Linsangan, L. C. (2002). A novel interaction of outer membrane protein a with c4b binding protein mediates serum resistance of escherichia coli k1. *J Immunol*, 169(11):6352–6360.

Press, W. H., Flannery, B. P., Teukolsky, S. A., and Vetterling, W. T. (1992). *Numerical Recipes in FORTRAN 77: The Art of Scientific Computing*. Cambridge University Press, 2 edition.

Raab, R. M., Tyo, K., and Stephanopoulos, G. (2005). Metabolic engineering. *Adv Biochem Eng Biotechnol*, 100:1–17.

Ram, S., Cullinane, M., Blom, A. M., Gulati, S., McQuillen, D. P., Monks, B. G., O'Connell, C., Boden, R., Elkins, C., Pangburn, M. K., Dahlback, B., and Rice, P. A. (2001). Binding of c4b-binding protein to porin: a molecular mechanism of serum resistance of neisseria gonorrhoeae. *J Exp Med*, 193(3):281–295.

Ramsey, S., Orrell, D., and Bolouri, H. (2005). Dizzy: Stochastic simulation of large-scale genetic regulatory networks. *J. Bioinformatics and Computational Biology*, 3(2):415–436.

Rawal, N., Rajagopalan, R., and Salvi, V. P. (2009). Stringent regulation of complement lectin pathway c3/c5 convertase by c4b-binding protein (c4bp). *Mol Immunol*, 46(15):2902–2910.

Reisig, W. and Rozenberg, G. (1998). Lecture on Petri nets I: Basic models. In *Lecture Notes in Computer Science*, volume 1491. Springer-Verlag.

Rodriguez-Fernandez, M. and Banga, J. R. (2008). Global sensitivity analysis of a biochemical pathway model. In Corchado, J. M., de Paz, J. F., Rocha, M., and Riverola, F. F., editors, *IWPACBB*, volume 49 of *Advances in Soft Computing*, pages 233–242. Springer.

Rodriguez-Fernandez, M., Egea, J. A., and Banga, J. R. (2006a). Novel metaheuristic for parameter estimation in nonlinear dynamic biological systems. *BMC Bioinformatics*, 7(483):1–18.

Rodriguez-Fernandez, M., Mendes, P., and Banga, J. R. (2006b). A hybrid approach for efficient and robust parameter esimation in biochmeical pathways. *BioSystems*, 83:248–265.

Rodriguez-Fernandez, M., Mendes, P., and Banga, J. R. (2006c). A hybrid approach for efficient and robust parameter estimation in biochemical pathways. *Biosystems*, 83(2-3):248–65.

Ross, S. M. (2002). *Probability models for computer science*. Harcourt Academic Press, San Diego.

Rullmann, J. A. C., Struemper, H., Defranoux, N. A., Ramanujan, S., Meeuwisse, C. M. L., and van Elsas, A. (2005). Systems biology for battling rheumatoid arthritis: application of the entelos physiolab platform. *Syst Biol (Stevenage)*, 152(4):256–62.

Runarsson, T. and Yao, X. (2000). Stochastic ranking for constrained evolutionary optimization. *IEEE Transactions on Evolutionary Computation*, 4:284–294.

Russell, S. J. and Norvig, P. (2003). *Artificial intelligence: a modern approach*. Prentice Hall, 3 edition.

Ruths, D., Muller, M., Tseng, J. T., Nakhleh, L., and Ram, P. T. (2008). The signaling Petri net-based simulator: a non-parametric strategy for characterizing the dynamics of cell-specific signaling networks. *PLoS Computational Biology*, 4(2):1–15.

Sahle, S., Mendes, P., and andU. Kummer, S. H. (2008). A new strategy for assessing sensitivities in biochemical models. *Phil Trans R Soc A*, 366:3619–3631.

Saltelli, A. (2008). *Global sensitivity analysis: the primer*. John Wiley, Chichester, England.

Salter, M., Knowles, R. G., and Pogson, C. I. (1994). Metabolic control. *Essays Biochem*, 28:1–12.

Sasagawa, S., Ozaki, Y., Fujita, K., and Kuroda, S. (2005). Prediction and validation of the distinct dynamics of transient and sustained ERK activation. *Nature Cell Biology*, 7(4):365–372.

Sato, Y., Hashiguchi, Y., and Nishida, M. (2009). Evolution of multiple phosphodiesterase isoforms in stickleback involved in camp signal transduction pathway. *BMC Syst Biol*, 3:23.

Scharfstein, J., Ferreira, A., Gigli, I., and Nussenzweig, V. (1978). Human c4-binding protein. i. isolation and characterization. *J Exp Med*, 148(1):207–222.

Schilling, M., Maiwald, T., Hengl, S., Winter, D., Kreutz, C., Kolch, W., Lehmann, W. D., Timmer, J., and Klingmüller, U. (2009). Theoretical and experimental analysis links isoform-specific erk signalling to cell fate decisions. *Mol Syst Biol*, 5:334.

Schmidt, H., Madsen, M. F. M. F., Dano, S., and Cedersund, G. (2008). Complexity reduction of biochemical rate expressions. *Bioinformatics*, 24(6):848–854.

Schoeberl, B., Eichler-Jonsson, C., Gilles, E. D., and Müller, G. (2002). Computational modeling of the dynamics of the map kinase cascade activated by surface and internalized egf receptors. *Nat Biotechnol*, 20(4):370–5.

Sheskin, D. J. (2004). *Handbook of parametric and nonparametric statistical procedures*. Chapman & Hall/CRC, 3 edition.

Sjoberg, A. P., Trouw, L. A., and Blom, A. M. (2009). Complement activation and inhibition: a delicate balance. *Trends Immunol*, 30(2):83–90.

Sjoberg, A. P., Trouw, L. A., McGrath, F. D., Hack, C. E., and Blom, A. M. (2006). Regulation of complement activation by c-reactive protein: targeting of the inhibitory activity of c4b-binding protein. *J Immunol*, 176(12):7612–7620.

Sobol, I. M. (2001). Global sensitivity indices for nonlinear mathematical models and their monte carlo estimates. *Mathematics and Computers in Simulation*, 55(1-3):271–280.

Spencer, S. L., Gaudet, S., Albeck, J. G., Burke, J. M., and Sorger, P. K. (2009). Non-genetic origins of cell-to-cell variability in trail-induced apoptosis. *Nature*, 459(7245):428–32.

Stryer, L. (1988). *Biochemistry*. New York: W. H. Freeman.

Swameye, I., Muller, T. G., Timmer, J., Sandra, O., and Klingmuller, U. (2003). Identification of nucleocytoplasmic cycling as a remote sensor in cellular signaling by databased modeling. *Proc Natl Acad Sci U S A*, 100(3):1028–33.

Swann, W. (1972). Direct search methods. *Numerical methods for unconstrained optimization*, pages 13–28.

Tasaki, S., Nagasaki, M., Kozuka-Hata, H., Semba, K., Gotoh, N., Hattori, S., Inoue, J., Yamamoto, T., Miyano, S., Sugano, S., and Oyama, M. (2010). Phosphoproteomics-based modeling defines the regulatory mechanism underlying aberrant egfr signaling. *PLoS One*.

Teusink, B. (2000). Can yeast glycolysis be understood in terms of in vitro kinetcs of the constituent enzymes? testing biochemistry. *Eur. J. Biochem.*, 267:5313–5329.

Thakar, J., Pilione, M., Kirimanjeswara, G., Harvill, E. T., and Albert, R. (2007). Modeling systems-level regulation of host immune responses. *PLoS Comput Biol*, 3(6):e109.

Thern, A., Stenberg, L., Dahlback, B., and Lindahl, G. (1995). Ig-binding surface proteins of streptococcus pyogenes also bind human c4b-binding protein (c4bp), a regulatory component of the complement system. *J Immunol*, 154(1):375–386.

Truedsson, L., Bengtsson, A. A., and Sturfelt, G. (2007). Complement deficiencies and systemic lupus erythematosus. *Autoimmunity*, 40(8):560–566.

Valk, R. (1978). Self-modifying nets, a natural extension of petri nets. In *Lecture Notes in Computer Science*, volume 62, pages 464–476. Springer-Verlag.

van Riel, N. A. (2006). Dynamic modelling and analysis of biochemical networks: mechanism-based models and model-based experiments. *Briefings in Bioinformatics*, 7(4):364–374.

van Stiphout, R. G. P. M., van Riel, N. A. W., Verhoog, P. J., Hilbers, P. A. J., Nicolay, K., and Jeneson, J. A. L. (2006). Computational model of excitable cell indicates atp free energy dynamics in response to calcium oscillations are undampened by cytosolic atp buffers. *Syst Biol (Stevenage)*, 153(5):405–8.

Vaseghi, S., Macherhammer, F., Zibek, S., and Reuss, M. (2001). Signal transduction dynamics of the protein kinase-A/phosphofructokinase-2 system in Saccharomyces cerevisiae. *Metabolic Engineering*, 3(2):163–172.

Veerhuis, R., Boshuizen, R. S., and Familian, A. (2005). Amyloid associated proteins in alzheimer's and prion disease. *Curr Drug Targets CNS Neurol Disord*, 4(3):235–248.

Vital-Lopez, F. G., Armaou, A., Nikolaev, E. V., and Maranas, C. D. (2006). A computational procedure for optimal engineering interventions using kinetic models of metabolism. *Biotechnol Prog*, 22(6):1507–17.

Vives, J., Juanola, S., Cairó, J. J., and Gòdia, F. (2003). Metabolic engineering of apoptosis in cultured animal cells: implications for the biotechnology industry. *Metab Eng*, 5(2):124–32.

von Dassow, G., Meir, E., Munro, E. M., and Odell, G. M. (2000). The segment polarity network is a robust developmental module. *Nature*, 406(6792):188–92.

Voss, K., Heiner, M., and Koch, I. (2003). Steady state analysis of metabolic pathways using Petri nets. *In Silico Biology*, 3(3):367–387.

Walport, M. J. (2001a). Complement. first of two parts. *N Engl J Med*, 344(14):1058–1066.

Walport, M. J. (2001b). Complement. second of two parts. *N Engl J Med*, 344(15):1140–1144.

Weng, G., Bhalla, U. S., and Iyengar, R. (1999). Complexity in biological signaling systems. *Science*, 284:92–96.

Wilkinson, D. J. (2006). *Stochastic modelling for systems biology.* Chapman & Hall/CRC mathematical and computational biology series. Taylor & Francis, Boca Raton.

Yoshida, R., Nagasaki, M., Yamaguchi, R., Imoto, S., Miyano, S., and Higuchi, T. (2008). Bayesian learning of biological pathways on genomic data assimilation. *Bioinformatics*, 24(22):2592–601.

Zadura, A. F., Theander, E., Blom, A. M., and Trouw, L. A. (2009). Complement inhibitor c4b-binding protein in primary sjogren's syndrome and its association with other disease markers. *Scand J Immunol*, 69(4):374–380.

Zhang, J., Koh, J., Lu, J. H., Thiel, S., Leong, B. S. H., Sethi, S., He, C. Y. X., Ho, B., and Ding, J. L. (2009). Local inflammation induces complement crosstalk which amplifies the antimicrobial response. *PLoS Pathogens*, 5(1):e1000282.

Zhang, Y. and Rundell, A. (2006). Comparative study of parameter sensitivity analyses of the tcr-activated erk-mapk signalling pathway. *Syst Biol (Stevenage)*, 153(4):201–11.

Zi, Z., Cho, K. H., Sung, M. H., Xia, X., Zheng, J., and Sun, Z. (2005). In silico identification of the key components and steps in IFN-γ induced JAK-STAT signaling pathway. *FEBS Letters*, 579(5):1101–1108.

Zweig, G. (1996). A forward-backward algirthm for inference in Bayesian networks and an empirical comparison with HMMs. Master's thesis, U.C. Berkeley.