Bayesian Network Approaches for Refining and Expanding Cellular and Immunological Pathways

by

Andrew P. Hodges

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Bioinformatics)
in the University of Michigan
2011

Doctoral Committee:

Assistant Professor Yongqun He, Co-Chair
Peter J. Woolf, FoodWiki, Co-Chair
Professor Brian D. Athey
Professor Matthias Kretzler
Research Investigator James D. Cavalcoli

To Patrick, touring the cosmos on sister Emily’s lightship,

And to all who sacrifice to define and improve life for all.

Acknowledgements

To my family, thank you for your continuous support over many years. I'd like to foremost thank my mother, Lynne Hodges, who has supported me and my brother since birth. You've sacrificed so much so that we could pursue our dreams and aspirations. And to my brother Steve, thank you for putting up with my crazy ways and means.

I wish to thank my thesis advisors and committee for their gracious help and unswerving passions for science. Drs. Peter Woolf and Yongqun "Oliver" He have served as effective co-advisors throughout my thesis. I will never forget our discussions and debates in Bayesian networks, modeling concepts, writing etiquette and style, and philosophical stances in science. The three of us rarely agreed on the same positions for any particular topic, yet our combined forces and continuous debates enabled the facilitation and publication of several interesting studies. Thank you so much for your support in many ways throughout these several years. Additional thanks go out to Drs. Brian Athey, James Cavalcoli, and Matthias Kretzler for serving on my thesis committee and contributing to the development of this thesis (as well as earlier participants Inhan Lee and Steve Qin). Chuanwu Xi and Dongjuan Dai along with Felix Eichinger are also acknowledged for their help with the experimental biofilm and progressive kidney disease dataset analyses and consultation, respectively.

To my friends, thank you for being there and for being you. To the staff at the University of Michigan, I sincerely address these thanks. Julia, Sandy, Denise, Yuri, and

Janet have kept my dreams of finishing the Ph.D. alive with their tireless assistance and patience over the years. Allen, Bruce, Alex, Jeff, and Jerome are acknowledged for their technical support and insightful discussions.

To many wonderous staff and faculty at other universities, I address these thanks. My earlier career progression is attributed to an outstanding educational experience at Manchester College. I wish to acknowledge my undergraduate honors thesis advisor and friend, Dr. James Brumbaugh-Smith. You inspired me to pursue academic research and higher callings, and were an excellent advisor. Other faculty who impacted my philosophical and scientific perspectives include Drs. Susan Klein, David Kreps, Mark Angelos, Andy Rich, and John Planer. Drs. Wilma Olson and Patrick DeLeenheer served as my REU mentors at Rutgers University during my undergraduate studies: I thank you both for your mentoring and training. And to my long-time friend and undergrad cross-country coach Brian Cashdollar, thank you for instilling a passion for excellence, perseverance, and camaraderie in all aspects of my life.

Preface

When looking back on this thesis, I am reminded of one of my favorite readings as a young adult. This well-known scientific novel/novella, “Flatland”, depicts the adventure of a triangle who is whisked out of his two-dimensional reality into three dimensions by an ambitious and instructive sphere. The actions and thoughts of the triangle are bounded within the two-dimensional reality of his plane (via both geometric and philosophical constraints). However, the perceptions of his existence change as he views his life and family from a new perspective, within a third dimension.

This book is such a fascinating view about the limitations put on ourselves through our life experiences and, more pervasively, the boundaries of our minds. For example, I often debate matters such as the existence of God, or other deities, and the boundaries of our existence. Is it possible that the presence of a higher level of existence (e.g. post-mortem) may be entirely hidden from our current reality as due to the set rules and foundations of our existence? The ways that we conceptualize our reality may be flawed or, at the very least, simplistic compared to the actual reality. As a toy example, the number 5 itself does not exist as 5 in a modulus 3 rule, but rather as the number 2 (e.g. 5%3=2). It is interesting to question how such mathematical abstractions and conceptions relate to existence. Our reality likely exists as a defined set of rings and fields within, adjacent to, or even overlapping other realities and as yet inconceivable domains. With this in mind, our ideal goal is to somehow extend beyond these rules and expand our perceptions and understanding of the universe(s) despite such limitations.

This thesis attempts to tackle comparatively simpler problems within defining reality. We will embark on an excursion through the realms of machine learning, systems biology, and pathway analysis, with the hopes of furthering our understanding within biology and medicine. A novel set of computational approaches are introduced which attempt to draw us, the triangles (and sometimes squares and circles, puns as necessitated) away from traditional approaches within our dogmatic plane to achieve new perspectives.

One of the foundational questions giving rise to our specific aims is how we can conceptualize the reality of biology using a set of computational (and more or less philosophical) rules. How do we identify as yet uncharacterized biological interactions and molecular entities which interact with sets of known molecular entities and their biological roles and interactions? How might we generate such predictions using minimal knowledge in order to save some knowledge for later comparison and validation (e.g. non-circular reasoning)? Finally, how are our computationally-predicted representations of the biological reality similar to and/or different from existing biological knowledge, and how do they both compare to the “real” underlying biology?

Table of Contents

List of Figures

List of Tables

Abstract

This thesis focuses on the computational analysis of cellular and immune pathways of living cells in response to molecular signals using machine learning approaches such as Bayesian networks. Bayesian networks (BN) have been applied to the reconstruction of these pathways (e.g. gene regulatory and protein signaling pathways) as network models using existing biological data. However, many biological interactions and molecular entities (e.g. genes and proteins) are not yet known which participant in the pathways of interest. For example, understanding the key biological interactions and participants of Jak/Stat pathway members in progressive kidney disease, a complication of diabetes, is necessary for refined understanding of the disease as well as future drug development. In order to resolve this issue, two major Bayesian network approaches are presented and applied in this thesis to allow refinement and expansion of known biological pathways to identify new interactions and molecular entities involved in the pathway model for future experimental analysis.

In Chapters One and Two, an overview of modeling approaches and assumptions for pathway refinement and expansion, including Bayesian network analysis, is presented. I introduce the major assumptions used when generating computational models such as Bayesian networks for known biological pathways from existing knowledge repositories. The major pathways analyzed in the thesis, including synthetic, the reactive oxygen species (ROS) pathway in *E. coli*, B cell receptor signaling pathway

in *Mus musculus*, and Jak/Stat signaling pathway in the *H. sapiens'* progressive kidney disease, are also introduced. Chapter Two specifically focuses on the Bayesian network theory implemented in the thesis and two major developed approaches.

In Chapter Three, the issue of how to refine existing Bayesian networks to identify the well-supported interactions predicted using underlying biological data and also remove false positive interactions is explored. I introduce a refinement algorithm called EdgeClipper which was developed to identify the most well-supported network edges in a distribution of saved Bayesian networks. The EdgeClipper algorithm implements a posterior weighting-based approach to prioritize these hypothesized interactions, and includes methods to remove poorly-supported interaction hypotheses. The approach was tested using synthetic and *Escherichia coli* reactive oxygen species (ROS) pathways and shown to faithfully identify many of the known interactions, as well as improve specificity with some sensitivity loss. The algorithm was demonstrated to have comparable performance to bootstrapping approaches with significantly faster computational time, and is effective for Bayesian network modeling with small datasets.

In Chapter Four, I introduce an effective expansion approach to identify yet unknown though potentially novel pathway members which likely influence the biological activities of the pathway. I developed an algorithm called BN+1 which can prioritize and identify which unknown pathway entities (e.g. genes, proteins) are involved in the biological pathway functions and activities. BN+1 was applied to the expansion of several synthetic, prokaryotic, and eukaryotic pathways. Major findings included the identification of genetic interactions between genes *gadX* and *uspE* and their direct regulation of biofilm activities in *E.coli*, all of which were verified experimentally. Other novel findings were achieved for a B cell receptor (BCR) signaling pathway using eukaryotic murine data.

In Chapter Five, the expansion and refinement algorithms were combined to achieve powerful predictions in both prokaryotic and eukaryotic pathways. As a test example, a small ROS pathway sub-network generated by EdgeClipper and later expanded by BN+1 recovered a known acid fitness island and new putative acid fitness regulators in the ROS pathway. This finding established the combinatorial approach of both methods. The EdgeClipper and BN+1 approaches were then applied in tandem towards understanding the roles of Jak/Stat pathway regulation during progressive kidney disease in two kidney compartments in *H. sapiens*. Our results revealed that Jak/Stat pathway shows relatively low overlap in supported interactions for the glomerular and tubule compartments, though the expanded pathway genes identified through BN+1 reflect the appropriate biological functions and stages of disease progression for the respective kidney compartments.

In Chapter Six, a novel web infrastructure (MARIMBA, http://marimba.hegroup.org) developed to facilitate Bayesian network, EdgeClipper refinement, and BN+1 expansion analysis is discussed. This tool, developed and used exclusively for the previous thesis chapters, allows researchers to freely execute all of the BN expansion and refinement methods and visualize and interpret results. The web-based tools are being improved and updated for increased public use.

In Chapter Seven, a summary of the major findings and results are discussed along with future directions. The refinement and expansion methods and their applicability to other next-generation and high-throughput datasets are discussed.

Overall, my results demonstrate that it is possible to refine and expand a protein-level signaling pathway representation using transcriptional microarray data, Bayesian network-based expansion and refinement algorithms, as well as other relevant bioinformatics approaches. The overlap between the generated and computational networks may vary according to the extent and type of biological data and type of selected pathway, though novel pathway members and interactions are discoverable using these approaches and underlying assumptions. The methods have been applied to a variety of biological systems with varying biological complexity, and are applicable to a wide variety of other biological and computational systems as well as high-throughput datasets.

Chapter 1

Introduction and Overview

1. Introduction

This thesis focuses on novel computational strategies for investigating biological pathways in a variety of living organisms. A set of powerful techniques will be explored which can identify likely interactions amongst sets of molecular components and elucidate novel mechanisms of disease and biological change. First, the increased need for computational models in pathway analysis is discussed. Second, common terminology and ideas will be introduced. Third, after defining some of the major dogmatic concepts throughout the thesis, the major questions and aims are introduced. Finally, an overview of the upcoming chapters is presented.

1.1 Driving Philosophical Questions

How are biological and computational networks (e.g. Bayesian networks) similar to and different from each other? Which entities in the biological networks are predicted by the computational networks (and vice versa), and which are missed? How can we identify novel interactions and interactors via computational analysis which are not yet included in existing biological models and knowledge repositories, or is this even feasible?

1.2 Defining computational and biological models of reality

1.2.1 Common Terminology and Definitions

In this section, some of the major terms and definitions which appear throughout the thesis are defined.

Dataset: a collection of data which represent selected experimental or synthetic data collected under specified conditions. Mathematically, the dataset can be considered a matrix with one dimension defined by experimental conditions or observations, and the other defined as the variables or nodes of interest (e.g. genes, proteins).

Network: a graphical model or concept which includes nodes that may be connected by edges or arrows. Networks are often used to model biological or other processes such as genetic regulatory and protein signaling pathways.

Edge: usually a directed or undirected arrow, this will represent some type of relationship between two nodes or variables in a network model. In the Bayesian network dogma, edges reflect a statistical influence from one node towards another node. These influences can also be represented via conditional probability tables.

Nodes: sometimes referred to as variables, these are both mathematical and graphical representations of some biological entity. Nodes are commonly used to represent selected genes, probes/probesets, proteins, various phenotypic measurements, or other entities. The actual type of node in each network may vary, and will be defined within the respective study.

Biological model: a network model generated using a combination of curated literature, database, and experimental data which reflect experimentally-derived results and conclusions. Biological models are often defined within specific experimental contexts or studies, and as such are considered to be more contextual.

Computational model: a variety of network models which incorporate information from a variety of experimental, literature, and other sources to study and infer behavior amongst network nodes and variables. Computational models require some prior or starting information as well as a set of mathematical rules to predict biological behaviors.

1.2.2 Defining Computational Models Based on Existing Biological Knowledge

A common approach for computational modeling is the generation of mathematical and programmatic models which can simulate some known set of variables and their interactions. These models require certain assumptions about the biological system, such as what biological entities (e.g. genes, proteins, miRNA) to represent as variables in the models, the types of interactions, as well as the type(s) of data included as either training or test information.

Figure 1.3 provides one such example of a biological model, the Nf-κB signaling pathway, and the assumptions imposed to construct a computational model. Figure 1.3A represents the sequence of events required for gene Nfkb1 to regulate some target gene in the pathway. After transcription of mRNA, mRNA translocation to the cytosol, protein translation and post-translational modifications (a good example of the classical central dogma in biology), the Nfkb1 protein can heterodimerize with other proteins. Activation of the Nfkb1 to induce transcription and translation of downstream target genes ("target") requires extracellular signaling through a cascade of kinases and other molecules to eventually phosphorylate the attached IκB monomer and allow translocation of the freed Nfkb1 protein into the nucleus. There, Nfkb1 acts as a transcription factor and initiates the transcription of mRNA at a variety of target binding sites for various genes.

In terms of computational modeling, this large sequence of signaling, transcription, translation, and translocation events can be simplified into a relatively smaller representation (Fig. 1.3B). Namely, the Nfkb1 gene influences target, which is represented as Nfkb1→target. This relationship which is represented graphically as a grey box in (Fig. 1.3A-B) is assumed by computational models which associate and sometimes correlate the mRNA expression levels for the two genes obtained via gene expression microarray studies and other related approaches.

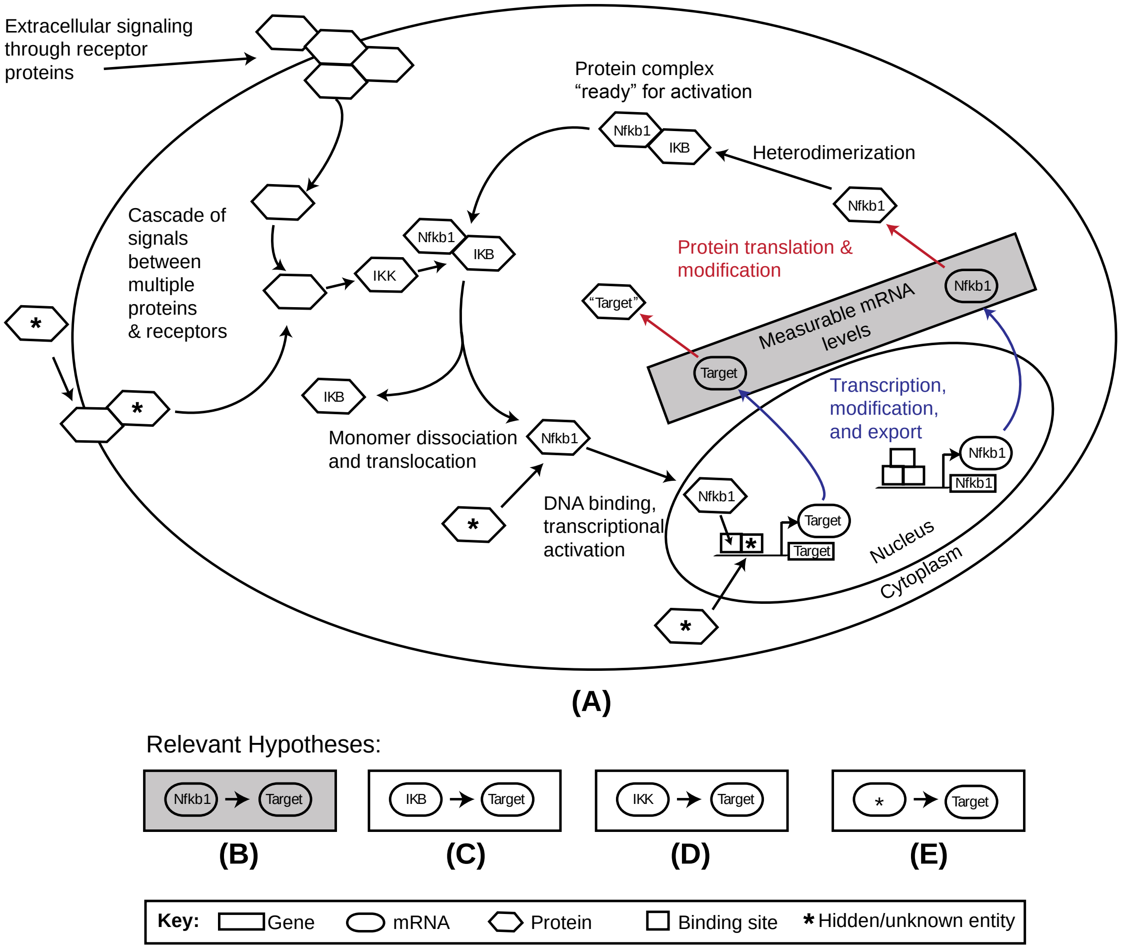

Extracellular signaling through receptor proteins
Protein complex "ready" for activation
Nfkb1
IKB
Heterodimerization
Nfkb1
Cascade of signals between multiple proteins & receptors
Nfkb1
IKB
IKK
Protein translation & modification
"Target"
Measurable mRNA levels
Nfkb1
Target
*
*
IKB
Transcription, modification, and export
Monomer dissociation and translocation
Nfkb1
Nfkb1
Nfkb1
DNA binding, transcriptional activation
Nfkb1
*
Target
*
Target
Nucleus
Cytoplasm
*
(A)
Relevant Hypotheses:
Nfkb1
Target
IKB
Target
IKK
Target
*
Target
(B)
(C)
(D)
(E)
Key:
Gene
mRNA
Protein
Binding site
* Hidden/unknown entity

Figure 1.1 Defining hypotheses and computational representations for biological networks. (A) Schematic representation of molecular events linking transcription factor Nfkb1 to mRNA regulation of selected target genes. (B) Major hypothesis that Nfkb1 influences or regulates target variable at the transcriptional level, as might be represented in a Bayesian network or other approach. (C-D) Known members of the Nf-□B protein signaling pathway which can be hypothesized to regulate the target gene and its mRNA expression. (E) Hidden or unknown factors which may also influence the regulation of the target gene expression. Note that not all interactions and entities shown in KEGG [1]or literature are represented in this conceptual model.

However, the figure illustrates other important interactions. For example, various signaling proteins may also affect the mRNA expression of the target if they are perturbed or absent (e.g. transmembrane proteins, various cytosolic proteins and macromolecules, other kinases and signaling molecules, transcriptional and translational machinery, etc). Hence, the possibility of other interacting proteins and their underlying mRNA expression profiles guiding the target expression must be considered (Fig. 1.3C-E). Thus, a major question is how to identify the most relevant interactions guiding target gene expression.

1.2.3 Comparing Computational and Biological Models

Defining computational and knowledge-based biological models can be problematic and relatively biased. In general, biological models are reductionist representations of underlying biology based upon experimental studies and traditional experimental biology approaches. These classical approaches are prevalent in literature and the various biomedical repositories in NCBI (http://ncbi.nlm.nih.gov), the European Molecular Biology Laboratory EMBL (www.ebi.ac.uk/embl/), and other online resources. A major strength of the knowledge-based biological models is that they can generate strongly-supported interactions within controlled environments.

However, such reductionist and controlled studies can be problematic when attempting to identify novel interactions or behaviors, such as in different environmental contexts. For example, predicting whether those interactions will occur in different experimental conditions not yet studied may prove problematic without computational analysis. Furthermore, generating biological models for much larger phenomena, such as those interactions spanning multiple tissues, organs, and even organisms are generally too large and complex for integrated experimental analysis.

Computational models allow additional insight into existing and potentially novel interactions. The computational models, which encompass a variety of different assumptions regarding biological behaviors and rules, can be used to model existing interactions, predict what effects may be observed if the existing rules, parameters, and entities in the model change (e.g. over time), and even bridge the analysis of changes across multiple biological compartments and hierarchical scales of organization.

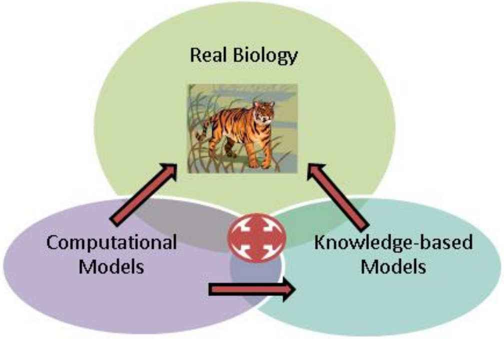


Figure 1.2 Venn diagram of computational versus knowledge-based networks and their relationship with underlying 'real' or existing networks. The target in this thesis is to expand the central (red) region in the figure for more comprehensive and integrative biological understanding. To achieve this goal, computational methods are introduced to identify new biological knowledge not yet present in existing repositories.

Ideally, the biological and computational modeling approaches (Fig. 1.1) can be bridged to generate comprehensive models of biological phenomena and change. This, however, cannot be accomplished until a detailed understanding of how the computational and biological models are similar to and different from each other. Figure 1.4 illustrates a conceptual overlap of biological and computational models and how they relate to their biological target realm. In both approaches, the computational and biological models might focus upon understanding the roles of components of the Nf-kB signaling pathway (or another of the hundreds of known pathways in *H. sapiens* and other species).

Computational and biological models may sometimes disagree. In some cases, the biological models will uncover interactions not predicted by the computational approaches. Likewise, in some situations the computational approaches may predict or infer novel interactions not yet seen in existing biomedical studies or literature. As mentioned earlier, Figure 1.1 lists common methods and resources for computational and biological modeling. A common theme in computational biology and bioinformatics is in determining the extent to which one can recover known interactions. This type of analysis, often referred to as benchmarking, assumes that a known gold standard is available for comparison. However, whether or not these disparate interactions are actually ‘real’ or purely a false positive result generated by the respective modeling technologies is itself another problem that surfaces in contemporary research, since the underlying biology is often hidden. Furthermore, the absence of some entities (e.g. ‘hidden players’ or ‘hidden variables’) may influence the results of either or both the computational and biological models and enhance such disparities. These issues give rise to the major questions and aims present in this thesis.

1.3 Specific Aims

Specific Aim 1: Develop the EdgeClipper algorithm to identify the most well-conserved or supported interactions from Bayesian networks trained on high-throughput data.

Specific Aim 2: Develop the BN+1 algorithm to identify novel hidden factors which are involved in the regulation of specified pathway entities using high-throughput data.

Specific Aim 3: Integrate the EdgeClipper and BN+1 algorithms to compare the mechanisms of genetic regulation in two kidney compartments during progressive kidney disease.

The three major aims and their final products are illustrated in Figure 1.5. Given some known biological information from existing biological pathway and public microarray repositories, starting networks are generated using Bayesian network analysis. These Bayesian networks are then refined and/or expanded to identify the most well-supported interactions and novel factors, respectively, which relate to the pathway activities.

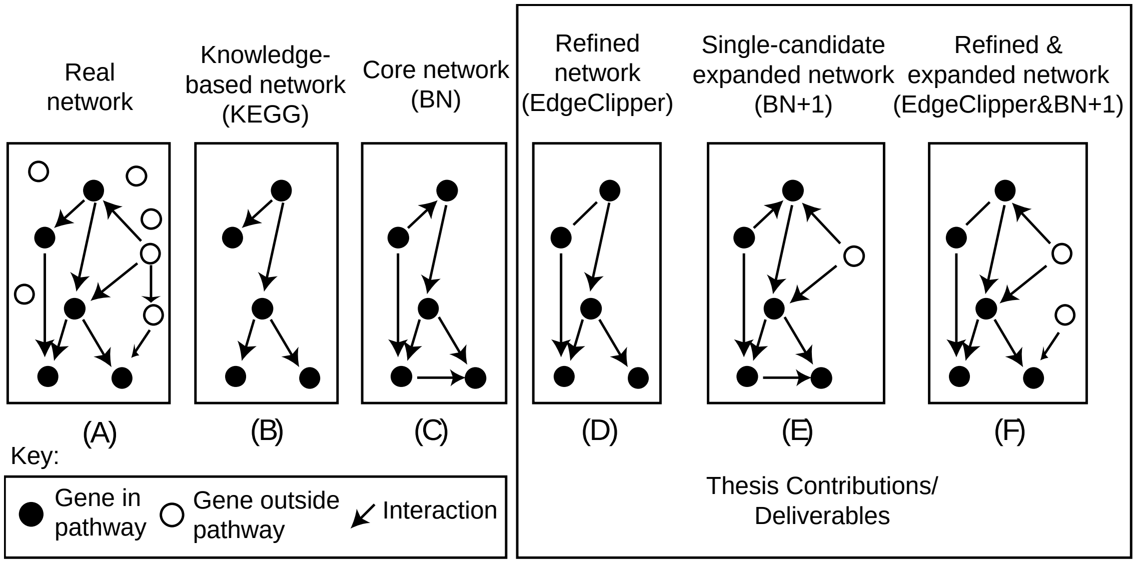

Real network
Knowledge-based network (KEGG)
Core network (BN)
Refined network (EdgeClipper)
Single-candidate expanded network (BN+1)
Refined & expanded network (EdgeClipper&BN+1)
(A)
(B)
(C)
(D)
(E)
(F)
Key:
Gene in pathway
Gene outside pathway
Interaction
Thesis Contributions/ Deliverables

Figure 1.3 Developed approaches for BN refinement and expansion. An existing, underlying biological pathway is assumed to be present (A). A priori, the set of known pathway components (B) are included as variables in a Bayesian network model (C). This core network can then be refined via EdgeClipper algorithm to identify the most well-supported interactions including novel testable interaction hypotheses (D), expanded via BN+1 algorithm to identify the most relevant and influential entities not yet 'known' in the pathway (E), and refined and expanded using EdgeClipper and BN+1 to filter down to the most well-supported core network interactions and then identify novel factors (F).

Two major algorithms are introduced in this thesis. The first approach is an algorithm which can prioritize sets of interactions from Bayesian network analyses from the most to the least likely and reduce false positive prediction rates. This novel approach is comparable to E-value methods in BLAST analysis [2], and can be used to identify interactions which do not appear in literature yet have a high probability or affinity as predicted by the Bayesian networks. Second, we introduce a powerful and novel expansion algorithm called BN+1 which can identify novel sets of interactors and their regulatory roles within a specified pathway context. These approaches allow both the comparison of computational and biological networks, the initial aim of this thesis, as well as to extend the achievable knowledge in the computational and biological models to increase the amount of overlap with yet undiscovered biological reality.

1.3.1 Selecting Biological Pathways for Analysis

The definition of a pathway depends greatly upon dogmatic and contextual views. One common method of defining pathways is identifying and associating the sets of molecular entities which are involved in specified biological processes. For example, the NF-κB signaling pathway includes a set of cytoplasmic proteins which, following a series of protein-level interaction events, induce a series of transcriptional regulatory changes in the nucleus. The NF-κB signaling pathway has many effects on cellular behavior and survival, and has been associated with many unfavorable effects on cellular survival when perturbed. The naming of this pathway reflects the major molecular entities or ‘players’ which have a major role in the biological outcomes and not all of the possible biological roles. Other pathways are named specifically for their biological outcome(s), such as the apoptotic pathway.

Classical definitions of pathway assume that a pathway has some input of information, a sequence of steps or processes, and some eventual outcome. This definition is more generalized for a variety of computational and biological pathways, and parallels some of the definitions for information flow and Shannon entropy ({Ma'ayan, 2006 #250;{Lungarella, 2006 #251};Gatenby, 2007 #229}). In this thesis, the majority of pathways discussed are protein-level signaling pathways which interact to some measurable degree with an underlying transcriptional regulatory pathway. This is an important distinction, since transcriptional regulatory and protein signaling pathways may have differential regulation and activities. Several studies have focused on modeling the interactions and flow of information between the transcriptional regulatory and protein signaling pathways [3];[4]. Interestingly, in many of the traditional gene expression microarray studies, researchers have assumed that the expression levels of selected genes were sufficient to predict protein activities and phenotypic outcomes [5]. As I will show later in this thesis, such assumptions are partially biased and naïve. Other studies have shown that this assumption may not be valid [6,7].

The selection of pathways in this thesis was, admittedly, biased according to publicly or internally available datasets. We selected an interesting set of such data from the Many Microbes Microarray (M3D) repository [8] which combined data from multiple published studies. In their papers, oxidative stress pathway genes were a few of several types of genes which were differentially regulated either up or down in some of the studies. Our selection of the ROS pathway reflected our individual interests in the oxidative stress pathway and its roles in bacterial survival and defense.

Many other approaches are available for the selection of pathways. A common method for selecting pathways involves the identification of the most highly differentially expressed mRNA transcripts [9] across a set of microarray experiments. In these studies, a set of control experiments are conducted along with some perturbation or other experimental modification. The log difference between the two conditions is calculated to determine the extent of up- or down-regulation of the gene with respect to the control experiment. These studies have often assumed that the most highly up or down-regulated genes were the most meaningful biologically.

Other approaches have not even assumed pathway-specific contexts for sets of genes. Instead, these approaches have either attempted to reconstruct entire networks starting with only a single seed gene or variable (e.g. bottom-up approach in [10,11,12]), or built entire global interactions networks (e.g. top-down ‘interactome’ or ‘exome’ analysis [12]) based on specified computational assumptions and then mined into local pathway-like subnetworks. These approaches often do not assume any starting prior knowledge about what constitutes a pathway, though tend to rely on computationally naïve assumptions and often miss more complex hidden interactions and factors.

However, we admit our bias towards studying known biological pathways of documented biomedical relevance and instead use this knowledge to our advantage. The initial studies with synthetic and *E. coli* ROS pathways are used as gold-standard references to test our Bayesian network approaches. Later applications of the approaches are shown in the murine B cell receptor signaling and human progressive kidney disease studies. By studying our approaches in the context of known biological pathways, it was possible to benchmark and validate the approaches, as well as offer insights towards their application to other biological pathways and high-throughput datasets.

1.4 The Need for Innovative Computational and Experimental Approaches

1.4.1 Extending beyond the genome sequence

One of the most significant developments in biomedical research was the advent of the genome sequencing era. This advent began with the whole-genome sequencing of *Haemophilis influenza* in 1995 [13], and after progressing through multiple species, resulted in the sequencing of the human genome. Major competitive efforts between the Human Genome Consortium and Venter groups to generate the first human genome map resulted in a wealth of new knowledge for genomic analysis [14,15,16] at the onset of the new millennium. One result of sequencing the various genomes was the applicable integration of approaches in genetics, comparative genomics, and bioinformatics to analyze health and disease in a variety of organisms [14].

However, a major challenge following the generation of the genome sequences was how to interpret and understand the genome, such as characterization of the exome or functional sequences in each genome. Collins *et al*. stated in 2003 that new technologies would be needed to catalogue all of the components in the human genome, interpret how those components interact to perform biological functions, as well as understand how genomes might change their components and/or functionality over (evolutionary) time [14]. At that time, a variety of new and existing experimental approaches were implemented to assign biological functions to predicted genomic components, including microarray analysis, RNA-seq and other high-throughput sequencing technologies, mass spectrometry, cloning, PCR, microfluidics, and other relevant technologies [14]. Each of these technologies would later play important roles in establishing biological functions of the components and implicate them in various disease and pathway models, as well as provide potential links between the molecular components, their functional roles, their involvement in various pathways, and, most desirably, their interplay and potential causative roles in targeted diseases [14].

Two grand challenges were posed by Collins *et al*. in the context of genomics and biomedical research [14]. The first goal was to comprehensively identify and functionally characterize components of the human genome. One major initiative launched in this regard was the ENCODE project, which sought to characterize all of the genetic components for a targeted 1% of the human genome [14,17,18]. The second goal was to elucidate the organization and roles of protein pathways and genetic networks in the context of cellular and organismal phenotypes [14]. This was initially achieved by assigning genes and proteins to pathways given evidence from knockout or knockdown, gain of expression, and targeted small molecule experiments. Figure 1.1 lists several databases which store these types of information in pathway-related contexts. Computational methods would also serve as an important method for achieving both goals. Given that many high-throughput datasets would be acquired from different biological scales (e.g. molecular, tissue, organ) using a variety of experimental approaches and technologies, the need for computational approaches to analyze this data was envisioned quite early. This need is even more important today, as described in the future directions section of Chapter Seven.

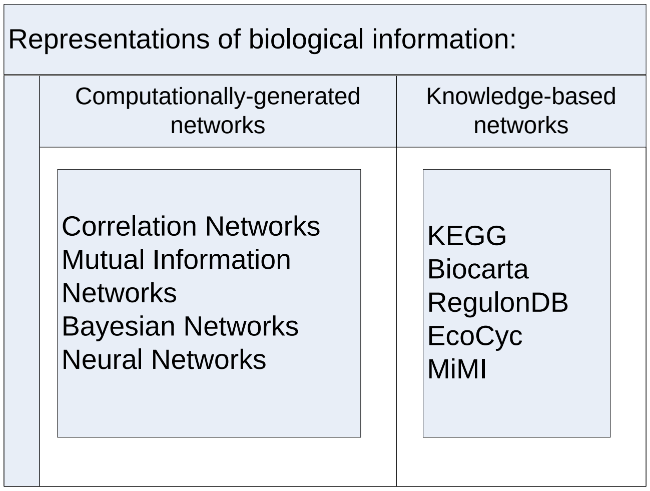


Figure 1.4 Methods and data resources for computational network analysis. Listed knowledge-based pathway databases include Kyoto Encyclopedia of Genes and Genomes [1,19,20], Biocarta [21], RegulonDB [22], EcoCyc [23], and the Michigan Molecular Interactions (MiMI) portal [24]. These pathway repositories incorporate a variety of information from different biological levels and experimental methodologies, as well as some inferred information from existing computational approaches.

These early driving goals in the human genome highlight the major concepts in this thesis. First, the definition and refinement of a biological pathway given certain types of biological data is necessary and important in understanding organismal biology, health, and disease. Pathways are defined using selected types of interactions and molecular entities, though they are often incomplete and require additional investigation. The identification of novel interactions and components could be achieved computationally, though more work is needed to both develop and verify such approaches.

1.4.2 Reactive oxygen species (ROS) detoxification pathway in *E. coli*

We hypothesized that Bayesian networks derived from microarray gene expression data are largely consistent with known pathway models and can be used as a basis to predict novel factors and interactions that influence a given pathway. In this study, the hypothesis was examined using the *Escherichia coli* reactive oxygen species (ROS) pathway. The *E. coli* ROS pathway has been well studied [25,26,27,28] and includes a variety of catalases and superoxide dismutases which are regulated at the transcriptional level by several known transcription factors and are involved in the processing of oxygen stressors such as oxygen ions, superoxides, and peroxide which are harmful to bacteria and living cells.

This particular pathway which was identified using the EcoCyc database [23], a BioCyc database designed specifically for *Escherichia coli* annotation and other knowledge. This particular pathway is especially interesting, since it relates protein-level interactions directly to transcriptional information. The model provides a more simple transition when comparing transcriptional regulatory networks generated by the Bayesian networks and microarray data to the pathway represented in EcoCyc. Twenty-seven variables or nodes were identified at the time of analysis, which included the five catalases and superoxide dismutases and twenty-two transcription factors represented on the corresponding gene expression microarray platform.

1.4.3 B cell receptor signaling pathway

As another example of the challenge of merging a pathway model and gene expression data, one study in this thesis focuses on the B-cell receptor pathway (BCR) as described by KEGG [1,19]. The BCR pathway is an integral component of the adaptive immune response mechanism by which B cells respond to foreign antigens [29]. While the KEGG pathway database includes a manually curated BCR pathway, this pathway is still considered incomplete [29].

A subset of genes was selected from the BCR pathway and studied using our developed BN+1 expansion algorithm. One question we addressed was whether the BN+1 algorithm could recover all components of the BCR pathway given the selected subset of genes.

1.4.4 Progressive Kidney Disease and Diabetes

One of the major focuses of systems biology and bioinformatics is on the analysis of complex biomedical phenomena such as diabetes. Roughly $2.3x10^8$ humans (and an estimated 5.1% of the global population) currently have Type 1 or 2 diabetes [30,31], establishing diabetes types 1 and 2 as major and prevalent diseases in the world. Furthermore, despite multiple treatment and preventative initiatives, there is no cure yet available for either form of diabetes [31].

Many biological processes in different tissues and organs, such as the immune system, are perturbed in diabetes. For example, both forms of diabetes involve the loss of beta-cells (differing in cause and rate of loss) with some concurrent inflammatory processes in the pancreatic islet [31]. It has been proposed that various regenerative and anti-inflammatory treatments which target the beta-cells could benefit patients with types 1 and 2 diabetes [31]. Other important complications following from diabetes include diabetic nephropathy and progressive kidney disease (a major focus of this thesis), as well as diabetic neuropathy.

Progressive kidney disease, a major complication of diabetes, includes a sequence of detrimental effects to the afflicted human patient. Progressive kidney disease is defined into classes based upon histological markers which represent the respective stage of disease progression. One significant aspect of the kidney disease progression is the order of histological changes in the distinct kidney microarchitecture. Major changes to the glomerular compartment are observed followed by changes to the interstitial tubule architecture [32]. Diabetes nephritis is considered a major cause of progressive kidney disease [32].

In terms of the progressive kidney disease (PKD), we are most interested in how the kidney compartments change their mRNA expression and regulation during the disease. Do the glomerular and tubulointerstitial compartments share the same predicted interactions between Jak/Stat pathway genes and hence not perturbed as a function of disease state, or are they different and reflective of the different stages of progressive kidney disease? What additional biological entities or factors, such as genes and proteins, are likely involved in the regulation or downstream activities of the Jak/Stat signaling pathway for each compartment and are yet unknown or not implicated in the disease? Figure 1.2 illustrates these two questions.

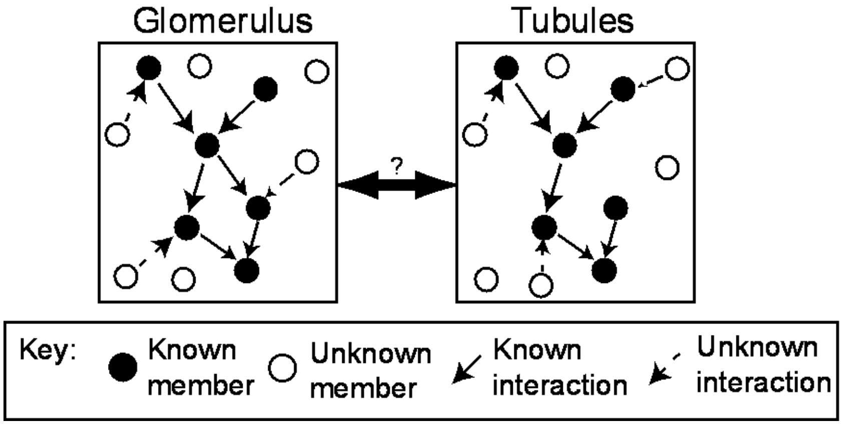


Figure 1.5 Jak/Stat regulation in two compartments during PKD.

In order to understand these questions and provide a systematic means of analysis, a set of computational approaches will be introduced and implemented to study the selected biological system. One major challenge is the selection of an appropriate set of molecular entities such as genes which can be compared fairly and methodologically between the two compartments. Another challenge is the identification of both novel entities and interactions may be important for either of both compartment. I will introduce two major methods which can be used independently or in combination to refine and/or expand the selected computational networks and allow comparison across the two compartments. Before that final analysis, simpler synthetic, prokaryotic, and eukaryotic networks are used initially in order to benchmark the developed approaches and to identify their advantages and caveats.

1.5 Major Biological Studies in This Thesis

The driving philosophical questions listed in Section 1.2 are studied in the context of several biological and conceptual studies. First, synthetic networks, a set of computational models, are used to test or benchmark some of our derived algorithms. Second, we acquired several publicly-available datasets from prokaryotic and eukaryotic studies involving gene expression microarray data. The most widely-studied biological system in this thesis is that of the prokaryotic *Escherichia coli* reactive oxygen species (ROS) detoxification pathway. This pathway, which was relatively well-studied yet still missing important interactors, was used as a representative system to test our developed algorithms. In a more complex organism, *Mus musculus*, the BN+1 algorithm was implemented to better understand mechanisms of genetic and protein-level regulation of the Nf-κB subnetwork in B cell receptor signaling. Finally, the approaches were combined and used to study an important and complex system, two major kidney microenvironments which change during progressive kidney disease. This disease is a major complication of Diabetes types I and II.

In all of the four studies, we asked which molecular entities not yet appearing in the known literature or knowledge repositories were most likely interacting with the selected biological pathways. In the synthetic, ROS, and progressive kidney disease pathways, we asked which known interactions were recovered by the Bayesian networks, and which disparate edges were best supported by the Bayesian networks and worthy of further investigation. We also asked whether the two microcompartments within the kidney showed similar genetic regulation for a selected pathway, and whether the predicted lists of novel interactors for each compartment were similar or different. In this regard, we were able to test whether the two compartments undergo similar or perturbed and different genetic regulation during progressive kidney disease. Our results suggest that not only are the two compartments different in terms of their gene regulatory network for the same set of pathway genes, but that their predicted interactors are also fundamentally different and follow known biological roles and functions already listed in the literature.

1.6 Summary of Chapters

In Chapter 2, the notion of Bayesian networks is introduced along with the two developed major algorithms, EdgeClipper and BN+1 approaches. Major assumptions and computational formulas are introduced, though described more formally in later chapters.

In chapter 3, EdgeClipper network refinement is described in detail and applied to the *E. coli* ROS pathway and developed synthetic networks. A novel equation which incorporates both posterior-based and frequency-based methods for edge weighting and prioritization was developed which allows direct comparison across these traditionally disparate methods. The approach was shown to be significantly faster computationally and comparable in performance to bootstrapping analysis.

In Chapter 4, the BN+1 algorithm is described in detail for three of the major biological studies in this thesis. BN+1 was benchmarked using synthetic networks. Then, BN+1 was used to expand and identify novel factors regulating the prokaryotic reactive oxygen species (ROS) and later the Nf-κB subnetwork in murine B cell receptor signaling. Novel findings included the identification and validation of genetic interactions between genes *uspE* and *gadX*, as well as their involvement in biofilm formation and activities.

In Chapter 5, the EdgeClipper and BN+1 approaches were combined to revisit previous models of the ROS detoxification pathway and then study progressive kidney disease in the two compartments of *H. sapiens'* kidney. The effective combination of these two approaches was established through an EC refinement and BN+1 expansion of ROS pathway genes which resulted in identification of an entire known acid fitness island.

In Chapter 6, the online implementation of the EdgeClipper and BN+1 approaches in MARIMBA is described. MARIMBA was implemented for all of the previous approaches, including three published BN+1 papers and an upcoming EdgeClipper paper. Furthermore, the two approaches are being developed as open-source Python code for greater public used.

In Chapter 7, a discussion of the major approaches and findings from the thesis appears. The various methods are then described in the context of their future applications and directions. References follow chapter 7.

Chapter 2

Introduction to Bayesian Networks

2.1 Introduction

One exciting development in bioinformatics research was the advent and application of Bayesian networks (BN) in biological research. Basically, BNs are graphical representations of statistical interdependencies amongst sets of nodes. BNs model interactions amongst sets of variables (*e.g.* genes, proteins) as probabilistic dependencies or influences. Judea Pearl introduced the notion of Bayesian networks in 1985 [33,34] to emphasize three aspects: (i) Often subjective nature of the input data information; (ii) Reliance on Bayes's conditioning as the basis for information updating; and (iii) Distinction between causal and evidential modes of reasoning. Bayesian networks were later implemented by Heckerman et al, Friedman *et al,* and various other research labs towards biological research [35,36,37].

Specifically, a BN for a set of variables X = {X_1, X_2, ...,X_n} consists of (1) a network structure S that encodes a set of conditional independence assertions about variables in X, and (2) a set P of conditional probability distributions associated with each variable [38]. Together, these components denote the joint probability distribution for X. The BN structure S is a directed acyclic graph, meaning that the network is hierarchical and has both top-level and terminal nodes and no directed paths which eventually return to them. We use Pa_i to denote the parents of node X_i in S as well as the variables corresponding to those parents. Given structure S, the joint probability distribution for X is given by

$$p(\mathbf{x}) = \prod_{i=1}^{n} p(x_i|\mathbf{pa}_i) \quad . \qquad (2.1)$$

However, the scoring for the overall network can vary depending upon the input data and assumptions used to generate the conditional probabilities for the child nodes and their parents. This concept as well as the implementation to handle this issue is described later in the chapter. In the next section, different methods developed to learn BN structures are introduced in detail.

2.2 Learning Bayesian networks (BNs)

The problem of learning a Bayesian network (BN) can be stated as follows: given a training dataset of independent instances, find a network that best matches the dataset. The common approach to this problem is to introduce a statistically sound scoring function that evaluates each network with respect to the training dataset and to search for the optimal network based on this score.

To dissect the processes of learning BNs, we summarize five major steps as follows:

- Data selection and pre-processing
- Prior definition (including variables and edges)
- Network searching strategy selection (e.g., simulated annealing, greedy)
- BN execution with a specific scoring method
- Results output and analysis

These steps will be introduced in detail here for gene expression data analysis.

2.2.1 Data selection and preprocessing

BN analysis is a powerful tool for analyzing high throughput data, e.g., DNA microarray data. Pre-processing is usually required to normalize raw data and possibly filter out those genes that do not show significant changes over all conditions. Some probes or probesets appearing in the microarray dataset may be considered uninformative if their signal-to-noise ratio is especially low, such that no significant changes in the overall expression of the biological entity (here, expressed and measured mRNA abundance) are observed across the set of microarray experiments or samples.

One method to filter out such uninformative microarray probes or probesets is by using a coefficient of variation (c.v.) [39] greater than at least 1.0. The coefficient of variation is generally defined as the absolute value of the standard deviation divided by the mean of the expression values for the microarray set. It has been demonstrated and assumed that the variation of transcripts when compared to other transcripts across the genome is relatively fixed, and that the c.v. is appropriate in this situation when considering signal-to-noise levels [39]. The inclusion of such cutoff criteria is important when later considering the discretization of the same datasets, since a faithful and accurate binning of the data cannot be achieved for data assumed to be ordered at random (e.g. not extending beyond the noise). Other filtering approaches assume minimum allowable values for log fold expression changes when comparing control and experimental groups in the data.

In this thesis, we analyze a variety of static datasets. Static datasets are assumed to be independent of each other, even if temporal data are present. One reason for this assumption is that we generate static Bayesian networks (BNs) which do not infer temporal relations between genes. More amenable approaches for temporal modelling include dynamic Bayesian networks (DBNs) and neural networks (NNs) [40].

2.2.2 Prior definition (including variables and edges)

After selecting appropriate data and variable sets for investigation, settings for the BN simulation must be chosen. Initially, assumptions must be made as to whether structural priors (e.g. the requirement of certain interactions to appear in a model) should be included or not in the BN analysis. It is not necessary to assume any structural priors for the initial set of variables. However, structural priors can be implemented, especially in cases where the biological interactions to be represented are well-established and also fully represented in the underlying biological data used for modelling.

2.2.3 Set up network searching strategy

Once the prior is specified, the BN learning becomes finding a structure that maximizes the BN score according to a BN scoring function. This problem is proven to be NP-hard [41]. Thus heuristic search is needed. The decomposition of the score is crucial for the optimization problem. For example, a local search procedure that changes one edge at a time can efficiently evaluate the gains of a specified score made by adding, removing, or reversing an edge. An example of such a procedure is a greedy random search algorithm with random restarts. Although this procedure does not necessarily achieve a global maximum, it reaches a local maximum and does perform well in practice [36]. Another commonly used method is simulated annealing search algorithm with a temperature schedule that allows for “reannealing" as the temperature is lowered [37]. Other BN searching strategies include stochastic hill-climbing and genetic algorithm [36].

2.2.4 Bayesian network scoring approaches

The key part of BN learning is to determine a scoring metric that compares networks and identifies the most likely or ‘best supported’ networks. Bayesian network scoring is based upon conditional probabilities. One commonly used scoring method is the Bayesian Dirichlet (BDe) score [35,37], which is a posterior probability defined as:

$$P(M \mid D) \propto \prod_{i=1}^{n} \prod_{j=1}^{q_i} \frac{(r_i - 1)!}{(N_{ij} + r_i - 1)!} \prod_{k=1}^{r_i} N_{ijk}!\,, \tag{2.2}$$

where n is the number of variables, q_i is the number of parent configurations for given variable i, r_i is the arity of variable i, N_{ij} is the number of observations with selected parent configuration q_i, N_{ijk} is the number of observations of child in state k with parent configuration q_i [35]. The calculation of this score is implemented in many software programs such as BANJO [40].

Another BN scoring method is the Bayesian Information Criterion (BIC), which was specifically designed to compensate for overfitting [42]. In the BIC method, the data is exponentially distributed, and the BIC is computed as:

$$-2 \ln p(x|k) \sim -2 \ln L + k \ln(n) \tag{2.3}$$

Where x is the observed data, n is the number of observations or data points, L is the maximized likelihood for the model, and k is the number of parameters to estimate. The

BIC is closely related to other information criterion such as Akaike, deviance, and Hannan-Quinn information criterion [43,44].

2.2.5 BN result output and analysis

To visualize BN results, different methods can be performed. For example, BANJO uses DOT type of BN result output. MARIBMA uses DOT and can also export networks as *.sif format for use in Cytoscape (http://www.cytoscape.org). Since different BNs are available, it is crucial for a user to select 'best-scoring' networks and/or generate consensus networks. Often methods are also needed to build weighted networks based on computational analysis or from literature and other database queries.

2.3 Bayesian network refinement methods

Although the BN provide are robust framework for biological pathway modeling, many factors such as insufficient or noisy data [19] and the influence of hidden variables (*e.g.*, unknown miRNAs and genes) can still generate missed or erroneously-included interactions in Bayesian and other network models [12,19,45]. To generate a reliable network given noisy data, consensus networks can be generated to increase the modeling specificity. A consensus network is defined as a network topology with edges that are conserved based on a list of calculated networks. Two basic approaches can be used to generate consensus BNs: (i) bootstrapping-based BN method with resampling of the original data [46,47,48], and (ii) identification of conserved edges within top ranked BN networks without data resampling [49]. The bootstrapping-based BN method obtains a consensus network by generating the best BN networks via sampling the original data with replacement and calculating the frequency of an edge present in the best networks obtained from BN modeling with different resampled datasets. This bootstrapping method has been proven to be reliable and consistent in consensus network generation and edge prioritization [46,50,51,52]. However, the central disadvantage of bootstrapping is that it is computationally intensive. This restriction limits the wide usage of bootstrapping in consensus network generations.

The other approach that uses the same data without resampling requires saving more than one top network. In this strategy, consensus networks include those variables and edges which appear at or above some selected frequency or weighting cutoff across a set of stored “top-scoring” networks with posterior probability scores. While this method has been frequently used as an *ad hoc* technique, the details of why it works and how it can be optimized with specific cutoffs have not been thoroughly studied. For example, by scoring 8 billion possible networks, a consensus network of the Her2-Neu signaling pathway was obtained by analyzing top 500 networks using proteomics and protein–protein interaction data [49]. Those edges that were conserved in >400, >300, and >200 of the 500 highest scoring networks were then recorded in the final consensus network. However, this method has not been justified rigorously. In addition, these numbers were chosen empirically, and each interaction edge in the consensus network was not assigned any score to indicate the prediction accuracy.

2.4 Bayesian network expansion methods

Bayesian network (BN) expansion is an approach that is built on the BN method and aims to identify new pathway elements that participate in a specified network. In this section, we will introduce basic BN expansion methods and then focus on describing our internally developed BN+1 algorithm and its implementation. Compared to the other network expansion methods described above, Bayesian network-based expansion methods provide distinct advantages, such as prediction of both linear and nonlinear functions, robustness in noise data analysis, and identification of causal or appearly causal influences representing interactions among genes. In general, Bayesian network expansion can be defined as the addition of new variables to an existing network, followed by rescoring and ranking of those variables.

BN-based expansion has been used for gene expression data analysis [11,53]. For example, Pena *et al.* reported an algorithm AlgorithmGPC that also grows BN models from seed genes [11]. This approach starts with one single gene and builds networks around this gene through expansion and pruning with a set number of genes. Gat-Viks *et al*. also generated a Bayesian network-based refinement and expansion method [53]. A main limitation of this approach is that it requires high-quality prior knowledge on the signaling pathways. The topology of the biological pathways may not be consistent with networks learned from transcriptional gene expression data obtained via DNA microarray studies. Therefore, a fixed topology as initial seed network may not be appropriate for robust network expansion simulations. Other BN expansion methods have also been published [54,55]. These approaches differ from each other but all showed different levels of success in identifying new pathway elements. In the following two sections, we will introduce our BN+1 algorithm [56,57], and how it can be implemented in the MARIMBA software.

Recently, we developed an algorithm termed "BN+1" which implements Bayesian network expansion to predict new factors and interactions that participate in a specific pathway. Broadly, the BN+1 algorithm iteratively tests to see if any single variable added to a given pathway will significantly improve the likelihood of the overall network. This approach is based on the observation that those variables which are hidden and regulate or are regulated by a network are more likely ranked with high posterior probability scores. Using a compendium of microarray gene expression data obtained from *Escherichia coli*, the BN+1 algorithm predicted many novel factors that influence the *E. coli* reactive oxygen species (ROS) pathway. Some of the predicted new ROS and biofilm regulators (*e.g., uspE* and its interaction with *gadX*) were further experimentally verified [56]. In another study, a synthetic network was also designed to further evaluate this algorithm. Based on the synthetic data analysis, the BN+1 method is able to identify both linear and nonlinear relationships and correctly identify variables near to the starting network [57].

Two major assumptions are included in my implementation. These include:

(1) The selection of seed (or called core) genes is an important step. The seed genes can be selected from an existing pathway database, from literature survey, or from internal experimental results. Since it is computationally expensive to calculate BNs using a large number of variables, it is often necessary to filter out some genes from an initial list using different criteria, for example, filtering out those genes that do not have significant changes among all microarray chips.

(2) While we use a top network structure generated from initial core gene simulation as prior, we prefer not to fix the core network structure for next network expansion. This preference makes our approach differ from a commonly used method of fixing the prior structure. One argument is that the prior structure is often determined by many layers of studies, including DNA, RNA and protein data analyses. When only RNA transcriptomic data are used, such prior structure may not hold. The fixture of a prior structure would result in obtaining suboptimal networks that do not match the datasets used for BN simulation.

These assumptions have important ramifications for some of the biological entities and behaviors predicted by our system. Comparison to some of the existing approaches may be limited due to the nature of these imposed assumptions.

2.5 Bayesian network refinement and expansion

The designed refinement and expansion algorithms were designed to be independent approaches to answer separate questions about how to refine the BN models generated for selected pathway entities (EdgeClipper, EC) or to expand the network representation to include novel hidden factors (BN+1 expansion). In this regard, our combination of refinement followed by expansion algorithms presents an approach similar to the Pena *et al.* AlgorithmGPC that also grows BN models from seed genes [11], with several distinct advantages.

First, our approach allows a target analysis of the pathway of interest. The Pena *et al.* algorithm [11] can continually contract and constrict to change the core network for expansion. However, this approach may also lose some sense of biological meaningfulness or function. An extreme case could involve losing all of the original core network genes after multiple iterations of refinement and expansion. A second advantage of our approach includes a more thorough analysis of the neighborhoods around the core network, which includes many of the top predicted expansion genes for each core network. This type of approach can be used to verify conserved biological functions and activities using annotation information and a naive natural language processing (NLP) technology. Such considerations were vital to the identification of novel genetic regulatory mechanisms and their directed biological funcions and later experimental validation.

In order to tackle the progressive kidney disease question of differential Jak/Stat pathway regulation in two compartments assuming minimal data (described in detail in Chapter 5), the EdgeClipper and BN+1 approaches would need to be combined in sequence (refinement first followed by expansion). The choice of ordering was selected to first establish the set of most conserved interactions shared in the pathway models for the two compartments, and then use those models as well-supported core networks for subsequent expansion.

Chapter 3

The EdgeClipper Algorithm for BN Refinement

3.1 Overview

To increase the specificity of Bayesian network (BN) modeling, consensus networks are often generated to identify the best supported edges in an empirically-generated set of top-scoring BNs. For better identification of consensus BNs and prioritization of predicted edges (*i.e.*, interactions), we developed an algorithm called EdgeClipper that sorts and analyzes the posterior distribution of high scoring BNs and identifies the most well-supported influences or edges across the posterior distribution. The EdgeClipper algorithm includes a unique B-value for network selection and a separate C-value for edge or interaction weighting and ranking.

As a cutoff for selecting the number of top BNs for inclusion in consensus network generation, a B-value score was defined as the right-tail cumulative density of the distribution of weighted posterior probabilities of selected top networks when considering all saved networks. Since some edges may not appear with 100% frequency in all of the saved Bayesian networks and/or may not appear in the top-scoring networks, we devised three versions of the EdgeClipper algorithm (EC-L, -R, and –F) with different criteria for network selection and edge inclusion. The loose EdgeClipper (EC-L) approach assumes all networks are included (B-value = 0) and no cutoff regarding edge frequency, whereas a restrictive EdgeClipper (EC-R) approach introduces more stringent assumptions (e.g. sliding B-values). The EC methods assign a C-value metric to each edge, based upon the weights and/or frequencies of the edges assigned during the respective approaches. This C-value is then used to rank the respective edges. EdgeClipper was tested and validated using synthetic data and *E. coli* microarray data analyses.

Our results indicate that decreasing B-values result in increased specificity and decreased sensitivity of predicting edges in consensus networks. Furthermore, the developed formulas can also represent the existing frequency cutoff methods. The edge ranking by C-values largely correlates with and is sometimes superior to the rankings produced by bootstrapping. EdgeClipper provides a systematic method for defining consensus Bayesian networks and assessing the relative support for edges in the network.

3.2 Introduction

In this study, we generalize the posterior probability-based method and develop a new algorithm called EdgeClipper for calculating consensus BNs and prioritizing edges in a network. The EdgeClipper algorithm can be adjusted to be more or less strict according to the number of BNs included in the consensus generation, as well as by defining the minimum frequency of edge occurrence in that set of networks. A B-value is used to define how many top networks (*e.g.*, 500 networks in the above example) to include in consensus network generation based upon the posterior distributions of networks.

Two major implementations of the EdgeClipper algorithm for posterior-based weighting were designed. The first method, the loose EdgeClipper (EC-L) approach, assumes all networks are considered or included during consensus network generation

(B-value = 0) and no minimum cutoff for edge occurrence is present. In the restrictive EdgeClipper (EC-R) approach, a subset of networks can be selected using the B-value metric (sliding-window selection of networks) as a cutoff, followed by selection of only those edges appearing with 100% frequency. The EC-R method also assumes that the set of considered interactions or edges is obtained from the top networks with the same best score. To achieve edge ranking and prioritization, a novel equation was generated which is robust and can incorporate both posterior probability-based and frequency-based weighting methods. The equation is significant since it can incorporate existing frequency methods into our EdgeClipper framework (as an EC-F function). All versions of the EC algorithm generate a C-value which weights the edges in the selected Bayesian networks and can be used for edge prioritization or ranking and as a cutoff criteria for consensus network generation.

Using synthetic network and *E. coli* pathway analyses with a compendium of *E. coli* microarray data, the EdgeClipper algorithm was verified to successfully predict consensus networks and conserved edges. The EC-L and EC-R approaches were compared to the prevalent bootstrapping approach in both synthetic and biological cases to benchmark and understand the algorithm, as well as to better understand the predictions generated for the ROS detoxification pathway.

3.3 Methods

3.3.1 Bayesian network scoring and top network search

In our study, the probability of a particular Bayesian network given a set of data was scored using log of the BDe score [35,37] which is the natural log of posterior probability

($S = \ln(P(M \mid D))$) and is listed in Equation 2.2 in Chapter 2. A random sampling approach (*e.g.*, simulated annealing) can be used to provide a broad search of possible networks during the BN analysis [56,57]. In the reported study, the calculation of the BDe score and network sampling by simulated annealing were implemented using the open source software BANJO [40]. Other searcher approaches have been used in other studies though were not explored in this analysis.

3.3.2 Derivation of B-value metric for constructing consensus BNs

The Bayes factor describes the relative improvement or loss of score for one network relative to another network. In this regard, the Bayes factor is represented as the ratio of posterior probabilities for two models, M_m and M_n. Given that $Score_x = \ln(P(M_x|D))$, the Bayes factor (BF) can be represented as follows for two candidate models:

$$BF_{m,n} = \frac{P(M_m \mid D)}{P(M_n \mid D)} \tag{3.1}$$

$$= \frac{e^{Score_m}}{e^{Score_n}} = e^{Score_m - Score_n} = e^{\Delta Score_{m,n}} \tag{3.2}$$

After saving a set of top-scoring Bayesian networks, the set of unique (non-redundant) scores $\{S_i \mid i=1..x\}$ are saved and then sorted from the highest to the lowest posterior probability. A Bayes factor is then calculated for each score to the top score (*i.e.*, $e^{\Delta Score_{m,n}} = e^{\Delta Score_{1,j}}$) in the saved set. Each Bayes factor is marginalized to give a weighted probability, such that for each score k,

$$P_k = \frac{e^{\Delta S_{1,k}}}{\sum_{i=1}^{x} e^{\Delta S_{1,i}}} \tag{3.3}$$

where x is the number of unique scores in the saved set.

A unique B-value is introduced by considering all considered networks using each of the weighted probabilities. The B-value represents the right-tail cumulative density of the distribution of weighted posterior probabilities:

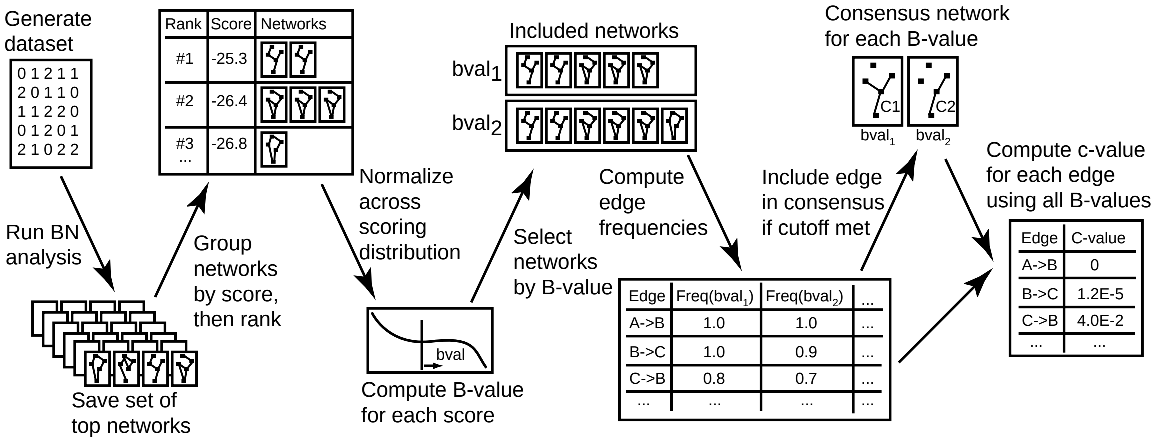
$$B-value = 1-\sum_{k=1}^{j} P_k = 1-\sum_{k=1}^{j}\left(\frac{e^{\Delta S_{1,k}}}{\sum_{i=1}^{x} e^{\Delta S_{1,i}}}\right). \qquad (3.4)$$

Here *j* is the number of top unique scores (natural log of posterior probability) chosen for inclusion in the consensus network calculation, while *x* is the number of all unique scores saved for network analysis. S_k is the natural log of posterior probability for a unique score *k* that appears for at least one saved network. *P* = 1-Bval is the sum of posterior probabilities for the top *j* scores normalized across all unique posterior probabilities (scores); *i.e. P* is a cumulative density function (CDF) value that represents the coverage of the best networks relative to all possible networks. P_k is defined above in Equation 3.3. The *B*-value measures the strictness of a “top” network compared to the total networks stored.

3.3.3 The EdgeClipper Algorithm

The EdgeClipper algorithm is shown in Figure 3.1, with pseudo-code presented in Figure 3.2. First, the algorithm requires as input a set of top-scoring networks from some Bayesian network analysis results. The networks are grouped according to identical log posterior scores, and then the scores are ranked from best to worst (where best is defined as closest to the value zero). This set of networks is used in total for the loose analysis

(EC-L), whereas a subset of networks is used for the restrictive approach (EC-R). The set of networks with log posterior score mapped to B-values greater than some specified input B-value is the subset obtained for the EC-R approach. The classical frequentist approach for edge selection is represented as a third method, EC-F, and also appears in the figure (more details described later in this chapter). After network selection, the set of all edges or interactions appearing in the selected networks is determined, as well as their frequency of occurrence in the set and their overall C-value weight. C-values are only assigned to those edges appearing in a selected network set. Finally, following C-value assignment, a consensus network is derived by including all edges with C-value above some selected threshold.


Generate dataset
Run BN analysis
Save set of top networks
Group networks by score, then rank
Normalize across scoring distribution
bval
Compute B-value for each score
Select networks by B-value
Included networks
bval$_1$
bval$_2$
Compute edge frequencies
Include edge in consensus if cutoff met
Consensus network for each B-value
C1
C2
bval$_1$
bval$_2$
Compute c-value for each edge using all B-values

0	1	2	1	1
2	0	1	1	0
1	1	2	2	0
0	1	2	0	1
2	1	0	2	2

Rank	Score	Networks
#1	-25.3	
#2	-26.4	
#3 ...	-26.8	

Edge	Freq(bval$_1$)	Freq(bval$_2$)	...
A->B	1.0	1.0	...
B->C	1.0	0.9	...
C->B	0.8	0.7	...
...	...	...	...

Edge	C-value
A->B	0
B->C	1.2E-5
C->B	4.0E-2
...	...

Figure 3.1 Schema for the EdgeClipper algorithm.

```
EdgeClipper

Input: Networks $N$ from a BN analysis with unique scores $S$, method: EC-L,R, or F, B-value cutoff $B$ (B-value = 0 for EC-L method), c-val cutoff

Data Preprocessing (Optional)

Group networks by unique scores in S

Computing C-values for edges in BNs

Save edges appearing in all included networks $N$ as edge set $E$.

For each edge $e$ in $E$,

    Compute C-value using N and $e$ using

        Equations 3.6 and 3.7 if EC-L

        Equations 3.6 and 3.8 if EC-R

        Equations 3.6, 3.9, and 3.10 if EC-F

(Optional, e.g. EC-R): For each edge $w$ not_contained_in $E$,

    C-value = “NA” or not defined

Generating consensus networks

For each edge $e$,

    Include edge in consensus if C-value > cutoff

Output: C-values defined for selected edge sets, consensus network generation
```

Figure 3.2 Pseudocode for the BN+1 expansion algorithm

3.3.4 Defining C-values for representation of edge consensus level

B-values were defined previously to select sets of best-scoring networks from large-scale Bayesian network simulations. However, another approach was needed to prioritize edges given the support from the best-supported networks. C-values were designed to show the overall weight for each network edge given certain assumptions on the set of networks and posterior probability distributions. The C-value for a given edge is generalized for the loose and strict assumptions as follows:

(3.5)

where *i* represents the indexes into the respective scores for included networks (with maximum index *Nmax*), W_i is a weight calculated for each edge, and function *f* generates a binary value of zero or one depending on the presence of the edge ( ) in the networks with same score.

The EC-R, EC-L, and EC-F approaches define W_i and differently (Equations 3.6-9 and Figure 3.3). In EC-L, W_i is defined as the normalized probability P_i (Equation 3.3) for a selected score. This weight is then added to the cumulative weighting if the edge does not appear in one of the networks with that score, as determined by Boolean function :

if , else 1. (3.6)

This procedure is repeated for all scores (and hence mapped networks) to give a cumulative reverse weighting for an edge given the set of saved networks. Those edges with defined C-values closest to zero have the most support. This formulation allows direct comparison to the EC-R approach and B-value metric (which attempts to minimize the right-tail distribution of normalized posterior scores).

In the EC-R formulation, W_i is also defined as the normalized probability for a given score. However, the Boolean function is defined iteratively as follows:

if (, and ( )), else 1. (3.7)

Here, the Boolean function incorporates all previous decisions about an edge's presence when traversing from highest log posterior score (best supported) to lowest (least supported). The weights in Equation 3.5, representative of the right-tail cumulative density function, accumulate after encountering the first network set lacking the edge of interest. Those interactions not appearing in the top-scoring network are currently not defined (e.g. "NA"), since we assume that the set of edges to consider come from the top-scoring networks (another possibility is to assume that the Boolean function always gives value 1). This formulation allows direct comparison of the EC-R and EC-L approaches.

The above methods and Equation 3.5 are also applicable towards describing the predominant frequency-based edge selection. We define this method as EC-F, and generate a simpler representation of the W_i and as follows:

if , else 0 (3.8)

W_i = (3.9)

Hence, assuming that =1 for all I, . Then, the C-values generated are bounded between 0 and 1 and contain equal weights for all of the networks. Thus, the EC-F method is exactly the frequency of edge occurrence across the set of uniquely-scoring networks assuming equal weights of networks (independent of posterior distribution).

After computation of the C-value for each method, the edges can be ranked according to their computed C-values. C-values are sorted from 0 to 1 (reflecting the weights of networks either not including an edge or not deemed significant in weight such as in the EC-R design) since it is easier to compare C-values such as 1x10^{-6}, 1x10^{-7}, etc. and not 0.999999 and 0.9999999 (though they related by subtracting each from the value 1.0). Those edges which have C-values greater than some cutoff can then be included as undirected edges in the consensus networks.

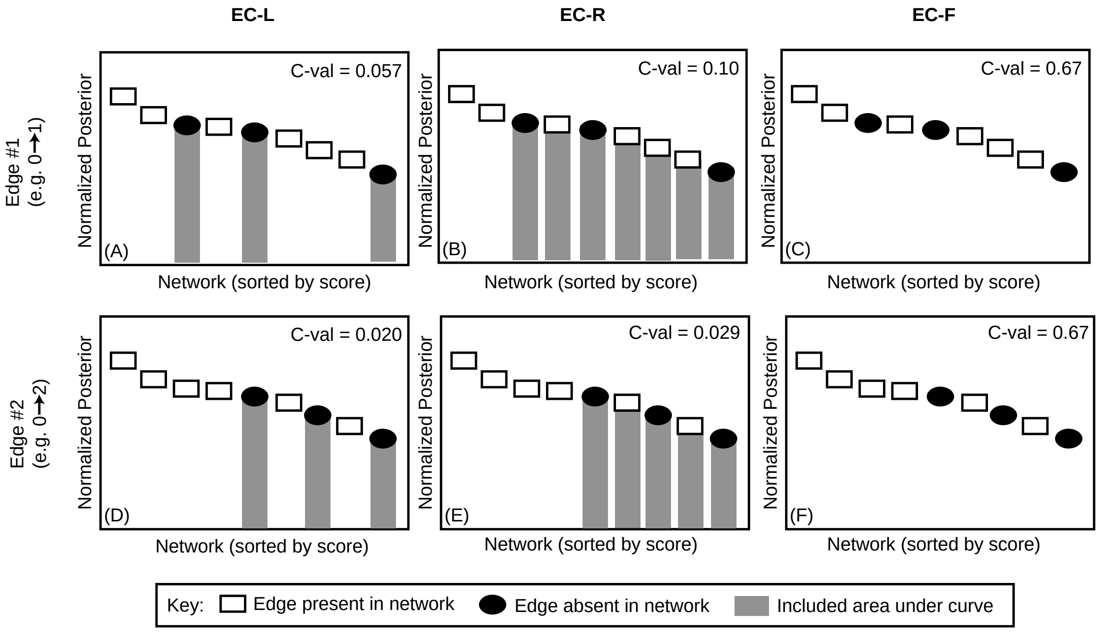


Figure 3.3 Comparison of EC-L, -R, and –F methods for two edges. Plots are shown for nine hypothetical networks sorted by log posterior scores. Two edges are selected from the networks, such that an edge may either appear in a network (box) or not (ellipsoid). Then C-values are calculated for each edge (A-C and D-F, respectively) using the three EC methods and listed in the top-right corner of each plot. For EC-L and EC-R, the C-value is computed using the sum of the area under curve for the indicated plot regions. For EC-F, this is instead the frequency of edge presence (or number of boxes divided by nine). In this example, interactions (or edges) #1 and #2 share the same frequency (EC-F) yet differ greatly in EC-R and EC-L values. Edge #2 will be ranked higher than #1 due to its smaller C-value in EC-R and EC-L.

3.3.5 EdgeClipper software

We have developed an EdgeClipper software package in Python to interpret Bayesian network simulation results and generate both B- and C-values for networks. This software uses a BANJO (http://www.cs.duke.edu/~amink/software/banjo/ [40]) output file as the input to the EC algorithms, and calculates C-values for each possible edge. BANJO provides both static and dynamic Bayesian network analysis. Both EC-L and EC-R methods are implemented for the static Bayesian network analysis. The Python source code for the EdgeClipper software program is available at: http://code.google.com/p/edgeclipper/. In addition, the MARIBMA program (http://marimba.hegroup.org) implements both EC-L and EC-R methods in PHP code. This program is open-source software with the Apache License version 2.0.

3.3.6 Synthetic data generation

A synthetic network with nine variables was designed for simulating microarray gene expression data. The synthetic data were generated based on a previous study by Luo *et al.* [58] with modifications. Specifically, the following mathematical formulae were used:

$$A = \mathrm{N}(0, 1) \tag{3.10}$$

$$B = \mathrm{N}(10, 5) \tag{3.11}$$

$$C = \mathrm{N}(0, 10) \tag{3.12}$$

$$D = A^3 + \mathrm{N}(0, 0.1) \tag{3.13}$$

$$E = A + \mathrm{N}(0, 0.1), \text{ while } (A+10>=B);\ E = B/10 + \mathrm{N}(0, 0.1) \text{ otherwise.} \tag{3.14}$$

$$F = (B-C)/(B+10) + \mathrm{N}(0, 0.1) \tag{3.15}$$

$$G = A + \sin(C) + \mathrm{N}(0, 0.1), \text{ while } (A+10>=B);$$

$$\text{else } E = B/10 + \sin(C) + \mathrm{N}(0, 0.1). \tag{3.16}$$

$$\mathrm{H} = \log(e^{A} + e^{F}) + \mathrm{N}(0, 0.1) \tag{3.17}$$

$$\mathrm{I} = (D + H) * (F/2) + \mathrm{N}(0, 0.05) \tag{3.18}$$

Separate datasets with 10, 50, 100, 250, 500, and 1,000 observations were sampled independently using the synthetic network topology and rules encoded in R [59]. These data were used in the subsequent synthetic network analysis. The five synthetic datasets with different numbers of observations or conditions were used in separate Bayesian network simulations. The EdgeClipper algorithm was applied towards refining the network results from each of these five BN analyses. Sensitivity and specificity were plotted as a function of the B-value cutoff variable for the different simulations.

3.3.7 *E. coli* ROS pathway data analysis using EdgeClipper

A compilation dataset comprising 305 gene expression microarray observations and 4,217 genes from *Escherichia coli* MG1655 was obtained from the M3D database [8]. A coefficient of variation threshold (c.v. ≥ 1.0) was used to select 4,205 genes for analysis. Twenty-seven genes were identified from the EcoCyc ROS detoxification pathway (downloaded on March 26, 2008) and matched to unique features found in 305 available gene expression microarray chips. Expression profiles for each gene were discretized using a maximum entropy approach that uses three equally-sized bins. To maximize the network search space, 4,000 independent simulations with random starts were used to search 2.5 x 10^{7} networks per start for a total of 1 x 10^{11} networks. Five top networks were saved from each run, thereby generating a final list of 20,000 top-scoring networks.

To reduce the large 27 gene network down to medium and small networks, we trimmed the networks using different B-values.

3.3.8 Analysis of selected EcoCyc pathways using EdgeClipper algorithm

All EcoCyc pathways were checked for the number of genes or corresponding proteins which successfully mapped to genes on the microarray platform. Seven pathways were then randomly selected from sets of pathways with 5, 10, 15, and 20 genes. These genes were then included as variables in the EdgeClipper analyses.

3.3.9 Bootstrapping analysis of consensus networks and edge prioritization

The standard bootstrapping method [60] was used to generate multiple datasets given some starting datasets from the synthetic and *E. coli* datasets. Specifically, bootstrapping with replacement was used to generate multiple data files with the same number of conditions or observations as the starting dataset. Each condition had an equal chance of being selected (uniform probability across all of the conditions), with the possibility that each condition could be selected zero, one, or more times and represented in the bootstrapped data file.

For synthetic data bootstrapping simulations, five major analyses were conducted. A set of 1,000 bootstrapped datasets (bootstrap with replacement) was generated for each synthetic datasets (10, 25, 50, 100, 250, and 500 conditions) to give 5,000 total bootstrapped datasets. Each set of 1,000 observation data files with identical numbers of observations was used in independent Bayesian network simulations. In each simulation, 5 x 10^7 networks were searched using simulated annealing for the nine variables and one of the 1,000 data files. The top 1,000 networks from the simulated annealing approach were then saved. Bootstrap results were compiled for the 1,000 independent simulations with selected data size, giving a total of 1x10^{6} saved networks per data size.

A bootstrapping test was also used to compare the result obtained from EdgeClipper for the analysis for *E. coli* ROS pathway. For this bootstrapping analysis, a “re-shuffled” dataset was first generated from the original dataset using the method of resampling with replacement. The new dataset was used for BN analysis by simulating 2.5 x 10^{7} networks. In each of the independent BN simulations with resampled data, the top one BN model was saved. This procedure was also repeated 1,000 times. Confidence in a particular edge is defined as the frequency of how often an edge actually appears in the set of reconstructed top BN models [60].

3.3.10 Comparison between EdgeClipper and bootstrapping in consensus network generation and edge prioritization

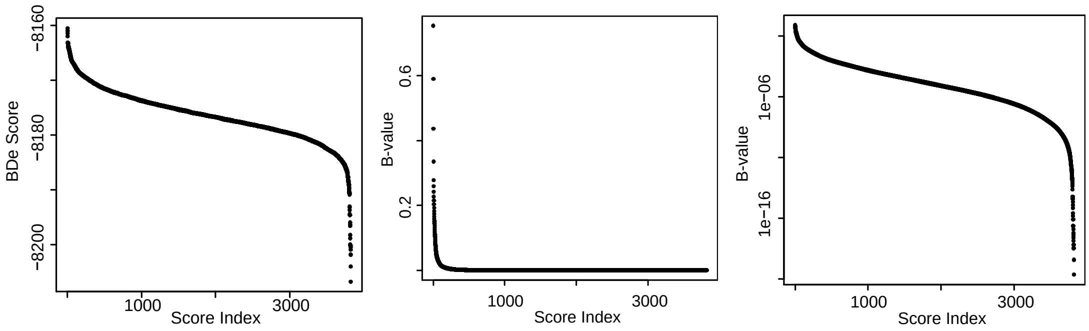
The correlation between results obtained from EdgeClipper methods EC-L and EC-R, and the bootstrapping was measure via Spearman rank correlation analysis [61]. Specifically, the cor.test function in the R ‘stats’ library was used [59]. Approximate P-values were recovered since some ranking ties were observed for the top results in both EdgeClipper C-values and bootstrap rankings.

3.4 Results

3.4.1 EdgeClipper algorithm

EdgeClipper was developed to increase the specificity (*i.e.*, reduce false positive rate) in the prediction of edges appearing in a selected consensus network while retaining as many true positive edges as possible. Furthermore, a key question in biological modeling is which of the many interactions predicted by the Bayesian network are likely to be verified experimentally, and which networks should be included when assigning these interaction weights and priority. At the start of the EdgeClipper workflow, a standard BN analysis is first used. Specifically, high throughput data (*e.g.*, microarray data) are pre-processed, various network topologies are searched (*e.g.*, by simulated annealing), and the posterior probability of each network topology given the data is calculated (*e.g.*, BDe score) (Figure 3.4). Instead of selecting only one network with the best score, EdgeClipper requires the storage of a large number of top ranked BNs to generate a posterior probability density (Figure 3.4).

Based on the set of non-redundant ranked scores from those saved networks, a B-value is computed as a normalized probability that gives a relative weighting for a BN score. The B-value can be considered as the relative or normalized weighting of networks scoring worse than a selected score. Specifically, the B-value represents the right-tail cumulative density of the distribution of weighted posterior probabilities, *i.e.*, the cumulative weighting from the best score (and hence best-scoring networks) to a BN score is subtracted from one to give a unique B-value. The primary reason of the selection of the right-tail instead of the left-tail cumulative density is that the right-tail density is more sensitive to the change in the number of networks selected for consensus network generation. Figure 3.4 illustrates the concept of B-value as compared to the original BDe distribution. When B-value = 0, all saved networks are selected for consensus network generation. A B-value of 1 represents that no saved network is selected. In practical application, at least one top network score is selected. In this case, the B-value is directly associated with the weighting of the first top network score. In a typical sorted BN result, the top networks have much higher posterior probabilities than the networks with low scores.



(A) (B) (C)

Figure 3.4 Comparison of B-value and BDe distributions. (A) BDe score distribution from a set of BN simulations sorted from best to worst score with score index i. (B-C) B-value distributions for the same scores with index i plotted using standard (B) and semi-log (C) y-axis. Here, (B) illustrates the severe drop-off of B-values (<<0.1), while (C) shows the close relationship of the B-value distribution with the original log posterior distribution.

After the set of networks is selected by B-value for inclusion in consensus network computation, conserved edges among the saved top networks will be identified and kept in the final consensus network. Edges which have an accepted C-value are specifically included in the consensus network. Other approaches, including the implemented bootstrapping method, have used edge frequency cutoffs which do not directly incorporate the log posterior distribution as support or weight.

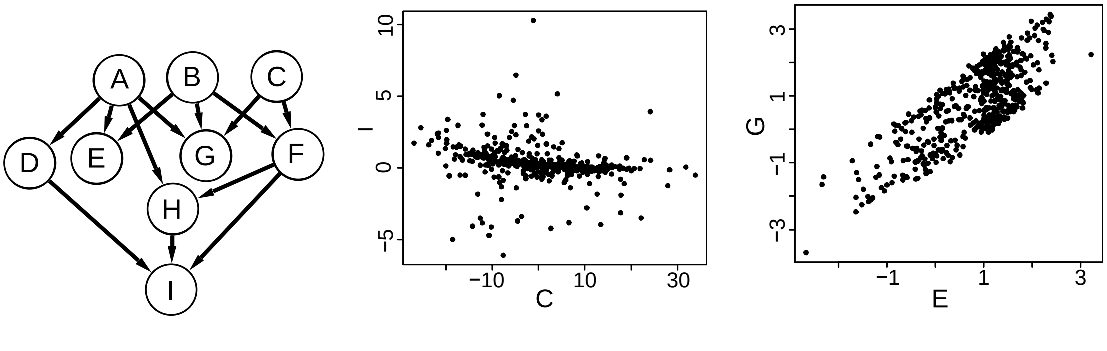
One important assumption in the approach is the inclusion of both edge directions cumulatively as equal representation of an interaction (hence influences A→E and E→A are assumed equal mathematically). One reason why we consider both directions for an edge is that many BN networks are often score equivalent, *i.e.*, there are multiple equivalent toplogies differing only in edge direction that have identical probabilities given an observational dataset [62]. Two equivalent BN structures with the same scores and edge topology may have different directions in many edges. An equivalence class of network structures can be uniquely represented by a *partially directed graph* (PDAG), where a directed edge X→Y denotes that all members of the equivalence class contain the arc X→ Y, and an undirected edge X—Y denotes that some members of the class contain the arc X→ Y and the others contain the arc Y→X [36]. This PDAG representation is applied in EdgeClipper. Specifically, EdgeClipper defines directed edges in the consensus network as those edges that appear with 100% frequency in one direction in all stored networks for the *B*-value selected above [63]. Undirected edges represent those edges appearing 100% of the time in both directions in all stored networks. Other approaches such as that in Bose *et al*. [49] will define an intermediate value between 0 and 1 for the frequency of edge occurrence.

Once a consensus network is generated, a C-value is defined to rank the edges in the consensus network according to their level of support from the data. The C-value of an edge represents a minimal B-value at which this edge disappears from the consensus network. Edges are next ranked according to their C-values. Those edges (representing interactions) with C-values closer to 0 are expected to be more conserved and specific than other edges with C-values closer to 1. Any C-value of an edge in a consensus network is always greater than or equal to the B-value used for generation of the consensus network. A C-value of zero for an edge indicates that all saved networks contain the edge, and hence the cumulative weight of zero networks lacking the edge is zero. Those top-ranked edges with their C-values being zero are the most conserved and well-supported interactions.

Based on the filtering ability of the B-value for network refinement, two EdgeClipper methods have been developed: EdgeClipper-Loose or EC-L (loose cutoffs with B-value = 0 a) and EdgeClipper-Restrictive or EC-R (restrictive cutoffs with B-value sliding) (Figure 1). The EC-L method contains all save networks (*i.e.*, B-value = 0) and does not have any restriction on the accumulative frequency of edge occurrence in all selected networks. The consensus network in EC-L includes all possible edges that have ever present in any saved BN. Therefore, these conditions are the loosest as we can ever expect when including posterior-based weighting. One advantage of EC-L is that after C-value calculation, every possible encountered edge will have a C-vale prioritization score. One disadvantage of this approach is that it is computationally expensive compared to EC-R. In contrast, EC-R requires that any edge in the consensus network should be present in all networks selected by the B-value cutoff. In EC-R, the differences between different consensus networks will indeed be determined by B-value. If the B-value is zero, the consensus network will be a PDAG that is formed using all save networks. When the B-values are increased from zero, a decreasing number of networks from the right-tail distribution will be used for consensus network generation, leading to more and more edges present in the consensus network.

3.4.2 Evaluation of EdgeClipper using synthetic networks

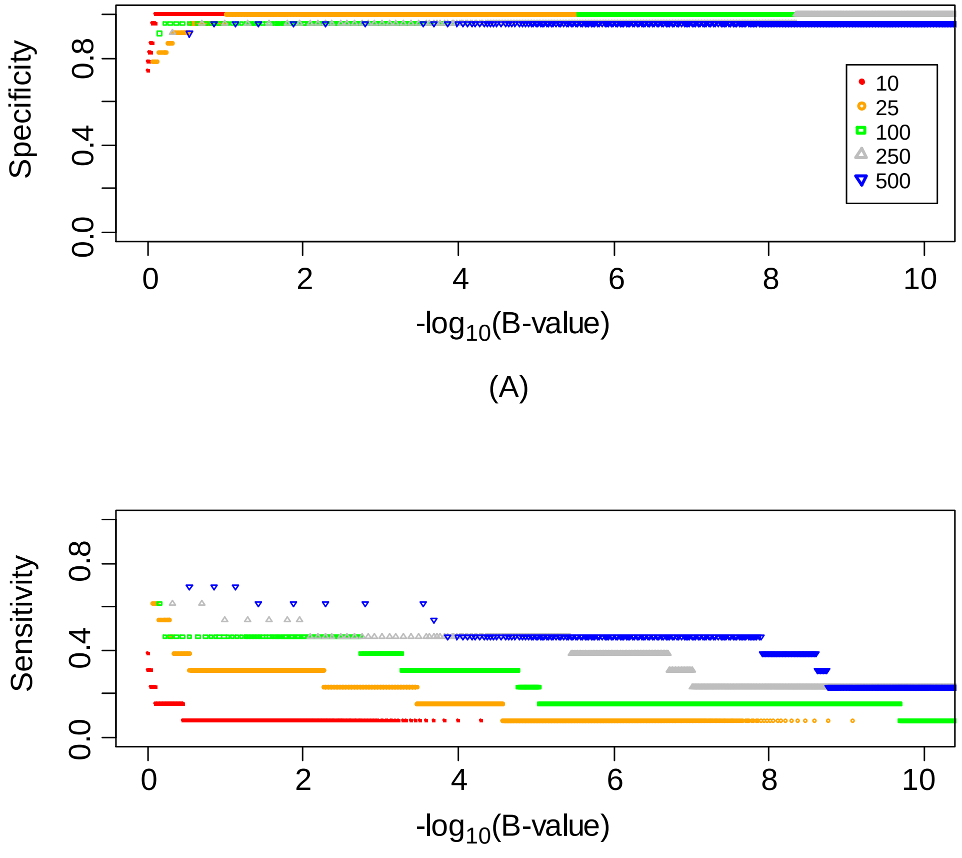
To benchmark the overall performance of the EdgeClipper algorithm, the algorithm was first applied to a synthetic network dataset generated with different data sizes (Equations 3.10-18). Figure 3.5A illustrates the synthetic network used in the analysis. Our initial analysis was focused on the impact of B-value in the final consensus network generation. Because EC-R has sliding B-values (e.g. a unique B-value cutoff mapped to each edge), it is natural to test the B-value impact using the EC-R method. In the synthetic data analysis, different data sizes ranging from 10 to 500 were used. With decreasing B-values from 1 to 0.01 (or increasing the X-axis –$\log_{10}$(B-value) value from 0 to 2 in Figure 3.6) for all data sizes, the usage of the EC-R method resulted in increasing gains in specificity in terms of edge prediction in the network. Meanwhile, the sensitivity decreases for each of the data sizes as the B-values decreases and approaches zero (or X-axis value approaching infinity).



(A) (B) (C)

Figure 3.5 Synthetic data analysis for benchmarking the EdgeClipper algorithm.

(A) The topology of the synthetic network. (B) A false positive interaction predicted by bootstrapping but ignored by EC-L and EC-R. (C) A false positive interaction predicted by all three methods.



(A)

(B)

Figure 3.6 Performance benchmarking of EdgeClipper for a range of dataset sizes. The performance of specificity (A) and sensitivity (B) of edge predictions in predicted consensus networks based on the EdgeClipper EC-R method was studied. The tested data sizes include 10 (red), 25 (orange), 100 (green), 250 (grey), and 500 (blue).

Besides the B-value, the data size is another factor that influences the specificity gain and sensitivity loss. With the same pattern of decreasing B-value cutoffs, the EdgeClipper analysis using smaller data sizes tend to have more gains in specificity.

However, the EdgeClipper consensus network analysis will also have quicker loss in sensitivity at the same time. In contrast, the EdgeClipper analysis with larger data sizes will have slower gain in specificity but also less loss in sensitivity when the B-values decrease. These results suggest that BNs trained using smaller data sizes will benefit the most in terms of finding specific edges using the EdgeClipper algorithm. In the synthetic analysis case, no noticeable gains in specificity were identified for B-value < 0.01 in all cases, suggesting an optimal B-value range could be determined in specific cases (e.g. 0.1-0.01 as a cutoff in Figure 3.6).

Using the same synthetic datasets, we also compared the performances of EC-R, EC-L, and bootstrapping in edge ranking (Table 3.1). In general, the results obtained from all three methods correlate well (P-value < 0.01). The three methods all predicted eight of the top nine edges with almost identical order. The one edge (E-G) was not predicted by any of the three methods. Because EC-R requires an edge to be present in all retained networks (including the top scoring network), those edges in the final consensus network will have to be in the top scoring network. Therefore, it is reasonable that only a portion of the edges were predicted by EC-R. In addition, we found that bootstrapping but not EdgeClipper sometimes predicted spurious or false positive interactions (Figure 3.5B-C). Sometimes a spurious or false positive interaction, such as G-E, was predicted by all three methods. Therefore, the synthetic data analysis indicates that EdgeClipper methods are equally good or better than bootstrapping in prediction of edges in consensus networks.

Table 3.1 Comparison of bootstrapping, EC-L, and EC-R in edge ranking in the synthetic data analysis. The analysis of 500 observation was used. Only those edges appearing in both the bootstrapping and EdgeClipper EC-L and EC-R results are ranked in this table. Key: NA- not defined, Y – edge exists as direct connection in synthetic network.

Edge	Bootstrap Frequency	EC-L C-value	EC-R C-value	Real?
A-D	1000000	-5.36E-16	0	Y
B-E	1000000	-5.36E-16	0	Y
C-F	1000000	-5.36E-16	0	Y
E-G	977309	-5.36E-16	0	
F-I	964218	6.83E-11	2.42E-09	Y
B-F	922296	8.54E-10	1.23E-08	Y
A-H	916495	7.28E-11	1.74E-09	Y
H-I	825251	8.16E-05	0.000137	Y
D-I	753223	9.28E-05	0.000208	Y
C-H	540636	0.230169	0.14175	
D-E	495874	0.949087	NA	
F-H	471996	0.769831	NA	Y
A-E	465201	0.050913	0.037012	Y
A-B	242656	0.999974	NA	
B-C	231985	1	NA	
B-D	228231	0.999997	NA	
A-G	225637	0.999986	NA	Y
A-I	204972	0.99999	NA	
D-H	151304	1	NA	
A-C	108918	0.999999	NA	

C-G	98479	0.99995	NA	Y
C-D	76659	0.999998	NA	
C-E	67706	0.999997	NA	
D-G	65834	1	NA	
C-I	61692	1	NA	
A-F	47598	0.999999	NA	
B-G	34076	1	NA	Y
D-F	26489	0.999999	NA	
E-F	21164	1	NA	
B-H	17537	0.999996	NA	
E-H	12737	1	NA	
F-G	6889	1	NA	
B-I	4706	1	NA	
G-H	3674	1	NA	
G-I	1770	1	NA	
E-I	131	NA	NA	

3.4.3 Results of *E. coli* ROS pathway analysis using EdgeClipper

To evaluate EdgeClipper using biological data, we first tested the *E. coli* reactive oxygen species (ROS) detoxification pathway using a compendium of microarray gene expression data from the M3D database [8]. The existing ROS pathway from EcoCyc models five *E. coli* enzymes important for the resistance against ROS toxicity, as well as 22 transcription factors that bind to targeted DNA sequences at the protein level. Using all 27 genes contained in the *E. coli* ROS pathway, a previous study was conducted to construct a consensus network with only the top one scoring networks among 20,000 saved networks [56]. We hypothesized that with more restrictive B-values (*i.e.*, more top-scoring networks used), more conserved networks could be generated with more conserved edges detected.

To measure the relation between B-values and the specificity of predicted consensus networks, EC-R was first used. Based on three distinct B-values, three consensus networks were identified using the EC-R method (Figure 3.7). The large network with a B-value of 0.247 contains all 27 genes from the original ROS detoxification list in EcoCyc. The predicted topology of the large network is basically the same as the one shown in previous work [56], which was generated by using all equivalent networks with the same BDe score. By comparing all the EcoCyc, RegulonDB, and literature data, a 42% correlation was observed between the predicted and known edges [56]. The medium consensus network had a B-value of 10^{-3}, which

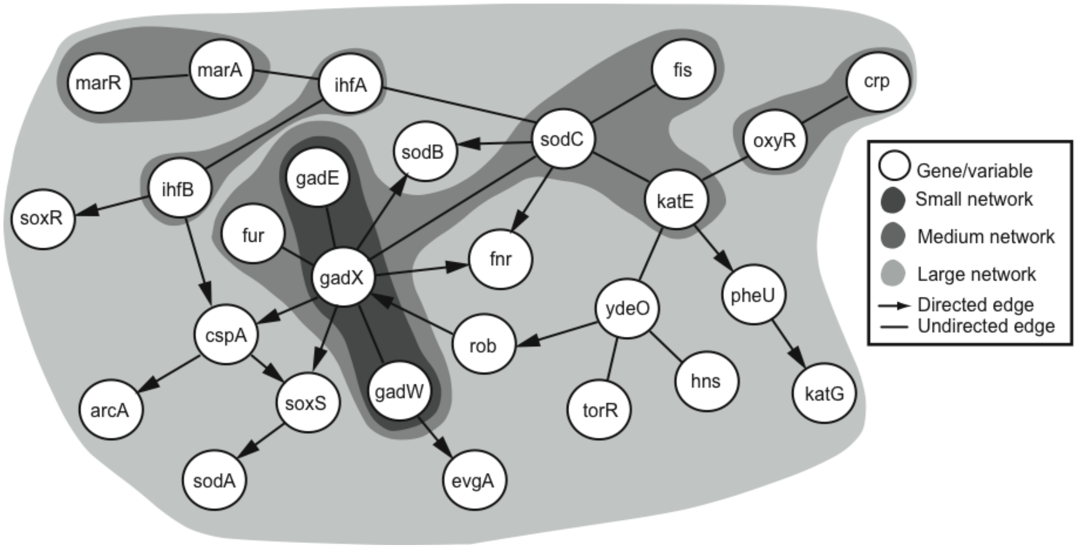


Figure 3.7 ROS consensus networks generated by B-values. Three successive consensus networks were generated using different B-values, as described in-text. The smallest core network was selected using the largest connected set of genes with B-value = 0.0. These networks were later used as core networks in Chapters 4 and 5 for BN+1 expansion and hidden factor identification.

corresponds to the selection of the top 3,644 simulated networks. By calculating the curated results from Supplemental Table 1 in Reference [56], this network contains 10 edges, and of them were verified to be true. The remaining edges are well supported hypotheses and deserve further investigation. When all 20,000 saved networks were used (*i.e.*, B-value = 0) in consensus network generation, all seven edges were supported based on existing knowledge (Supplemental Table 1 in Reference [56]). This study also found that all edges shown in a more conserved network are also found in a less conserved network. For example, all edges shown in the small network (B-value = 0) exist in the medium network (B-value = 10^{-3}) and large network (B-value = 0.247). In summary, the EC-R method can refine the consensus network down to the best-supported interactions appropriate to the underlying dataset used in the analysis. The EC-L method provides a nearly identical list of ranked interactions, including those interactions which do not appear initially in the top-scoring network.

One question was whether the bootstrapping or EdgeClipper predicted interactions most closely reflect the underlying biological interactions. Spearman rank correlation testing revealed that the edges in the *E. coli* ROS pathway network ranked according to EC-L and EC-R show a significant negative correlation (P-value < 0.01 in both cases) with the ranked bootstrap frequencies for those edges. Table 3.2 shows the major interactions and their predicted weights and rankings according to the bootstrap, EC-L and EC-R methods. Specifically, C-values are ranked from zero to one with zero being a score for the most conserved and specific edges. In contrast, the bootstrapping confidence values are ranked from one to zero, where zero for an edge represents no data

Table 3.2 ROS BN interactions predicted using three methods.

Interaction	Bootstrap Freq. (2-way)	EC-L	EC-R	Rank: Bootstrap	Rank: C-value (EC-L)	Rank: C-value (EC-R)
gadX-gadE	0.999	4.40E-15	0	1	1	1
marA-marR	0.999	4.40E-15	0	1	1	1
gadX-gadW	0.998	4.40E-15	0	3	1	1
katE-sodC	0.996	4.40E-15	0	4	1	1
fis-sodC	0.988	4.40E-15	0	5	1	1
ihfA-ihfB	0.986	4.40E-15	0	6	1	1
crp-oxyR	0.863	1.96E-07	6.55E-05	7	7	7
sodA-soxS	0.756	0.0195	0.145	8	22	23
cspA-ihfB	0.733	0.0118	0.0773	9	19	20
gadX-fur	0.719	4.61E-06	0.000939	10	8	9
ihfA-sodC	0.718	0.00482	0.0507	11	18	17
katE-pheU	0.661	0.000831	0.0262	12	12	13
evgA-gadW	0.66	0.423	0.437	13	30	30
sodB-sodC	0.65	0.0399	0.192	14	26	26
fnr-gadX	0.631	0.000108	0.00586	15	10	10
gadX-rob	0.592	0.00207	0.0221	16	15	12
ihfA-marA	0.564	0.0219	0.121	17	24	22
hns-ydeO	0.557	0.00128	0.0422	18	13	16
cspA-soxS	0.555	0.0441	0.215	19	28	28
gadX-sodC	0.533	1.44E-5	0.000693	20	9	8
rob-ydeO	0.532	0.0211	0.166	21	23	24
torR-ydeO	0.53	0.00137	0.0344	22	14	14
katE-ydeO	0.425	0.00429	0.0537	23	16	18
cspA-gadX	0.394	0.0167	0.0773	24	21	20
gadX-soxS	0.383	0.00432	0.0404	25	17	15
arcA-cspA	0.358	0.000418	0.0141	26	11	11

katG-pheU	0.352	0.0855	0.259	27	29	29
gadX-sodB	0.348	0.0417	0.192	28	27	26
katE-oxyR	0.291	0.0333	0.174	29	25	25
ihfB-soxR	0.28	0.434	0.590	30	31	31
fnr-sodC	0.276	0.0128	0.0710	31	20	19

support for this edge. Interestingly, all of the consensus edges appear in the top 57 interactions listed by the ranked bootstrap results (top 57 out of 50,295 total interactions saved). However, one interaction, *sodA-sodB*, was missed by EdgeCliper but was highly ranked by bootstrapping. Because the *sodA-sodB* interaction did not appear in the top-scoring Bayesian network, no C-value was assigned according to EC-R. Furthermore, EC-L ranked the *sodA-sodB* as 35th with a C-value of 0.935.

3.4.4 Analysis of selected EcoCyc pathways using EdgeClipper algorithm

Ten additional EcoCyc pathways were selected for additional analysis. The pathways were selected for having 5 to 25 genes which matched the microarray platform. BN analyses were conducted for each of the ten pathways, followed by subsequent refinement with the EdgeClipper EC-R method. The EC-R method was tested for sensitivity and specificity using B-value cutoffs of 0.1 and 0.01. Our analysis results indicate that the lower B-value a cutoff was used, the more specific interactions it predicted. The results are similar to the ones found in the *E. coli* ROS pathway analysis and further confirm that EC-R is able to increase the prediction specificity of edges in consensus networks.

3.5 Discussion

This study reports the development and evaluation of EdgeClipper, a posterior probability-based algorithm for generation of consensus networks and prioritization of edges in a BN. Our synthetic and *E. coli* data analyses indicate that EdgeClipper improves the specificity of consensus network generation and provides an effective way to rank edges.

EdgeClipper is a posterior probability-based algorithm developed to systematically construct consensus networks and rank the support for each edge in the network. The concept of consensus network generation based on analysis of a set of stored 'top-scoring' networks has been conceived before. For example, many scientists have used such a method *ad hoc* [49]. The partial directed acyclic graph (PDAG) method, proposed to summarize the networks with equivalent classes [36], is a type of consensus network built on saved networks with the best posterior probability score. However, the PDAG method has not been associated with any systematic and quantitative measures. Hartemink also proposed an approach for BN edge prioritization by computing cumulative posterior probabilities based on all saved top-scoring networks [64]. This proposed method is similar to the C-value calculation based on EC-L. However, Hartemink's method does not consider equivalent networks, and the approach has not been tested. The major contribution of EdgeClipper is that instead of directly using posterior probabilities, we are the first to use the right tail of accumulated density in the posterior probabilities for consensus BN analysis. If posterior probabilities are directly used, it would be difficult to compare different scores (Figure 2). The switch of using the right tail of accumulative posterior probabilities (*i.e.,* B-value) allows us to use the posterior probabilities to generate consensus network. In addition, the introduction of B-value in EdgeClipper allows us to consider a whole or a portion of saved top networks with a B-value cutoff. Furthermore, because many BNs are equivalent, the use of posterior probabilities in EdgeClipper allows us to group these equivalent networks with the same scores using the PDAG approach. An ignorance of those networks with the same scores due to equivalent or nearly equivalent network structures, may bias the results of edge prioritization.

The EC-L and EC-R methods are two EdgeClipper methods with extreme settings (Figure 3.1). When we calculate consensus networks, EC-R only uses the top scoring networks based on B-value cutoff, while EC-L includes all networks ranging from high scoring to low scoring networks. Since EC-R uses the Fe-value of 100%, EC-R will not rank those edges that do not exist in the best scoring networks. In contrast, EC-L does not have any restriction of the Fe-value cutoff, so EC-L will assign a score for any edge that exists at least once in any of the saved networks (Table 1-2). However, for those edges present in the best scoring networks, these two methods correlate well. Those edges that are absent from the best scoring networks usually have low C-value scores. Therefore, if we are only interested in finding those most specific and conserved edges, EC-R is sufficient. While both EC-R and EC-L are computationally faster than bootstrapping because of the lack of the data resampling step, EC-R is slightly faster to compute than EC-L due to its restrictive settings.

In our synthetic and *E. coli* real data analyses, we found that the results of EdgeClipper and bootstrapping largely correlate, and in some cases EdgeClipper behaves better than bootstrapping in predicting an edge. We note that there are fundamental differences between bootstrapping and EdgeClipper. First, the underlying datasets used in the two approaches are different. The original underlying dataset was used as a whole in the EdgeClipper approach. However, the bootstrapping method uses reshuffled data. The reshuffling may change the binning assignment from one sample to another. It is important to recognize this difference, since for relatively small data sizes, the bootstrapping approach may be extremely sensitive to the implemented sampling approach and loss/gain of selected data vectors. However, EdgeClipper uses all the original data and thus does not have this problem.

Due to the differences in data processing, the primary questions addressed by these two methods become different. Based on the B-value derived from the posterior probability, EdgeClipper focuses on answering the question "How well does it fit in with the model with the data?" The B-value fits in line with density, statistical P-value, and BLAST E-value analyses in that all these values consider the weighting or significance of a set of results within a selected probability distribution. Significance is used loosely here since our B-value is based on the observed probability distribution. In contrast, bootstrapping answers the question "How sensitive is the fit of the model to specific data, or how robust is it?" Here, the bootstrapping confidence value can be grouped together with q-value and cross validation results. The bootstrapping can be used to identify whether the BNs are sensitive to certain data, such as in cases where datasets are small or sparse, or selected data are over-represented in the set and bias the overall model. Different bootstrapped data can give rise to unique posterior probabilities in the Bayesian networks, though we would expect similar overall distributions if the model is robust given the data.

EdgeClipper has basically two uses: consensus network creation, and edge ranking. Our studies indicate that EdgeClipper generates more specific and conserved consensus networks and the edges can be ranked accordingly with C-values, with EC-R giving the most specific interactions. Therefore, EdgeClipper can be used to confirm known interactions and identify new interactions with high specificity. For example, we identified many unknown but specific and conserved interactions (*e.g.*, *sodC* – *gadX*) in the *E. coli* ROS pathway (Figure 6). Since BN modeling attempts to predict many new interactions, it is too expensive and most likely impossible for a wet-lab to test all possible interactions. Therefore, it is crucial to identify those most promising interactions for experimental verification. The EdgeClipper consensus network approach allows us to focus on a small network with high confidence. The C-value ranking provides a way for a research to experimentally investigate those predicted interactions with the best chance of success.

Chapter 4

BN+1 Algorithm for Identification of Novel Pathway Members

4.1 Overview

Signaling and regulatory pathways that guide gene expression have only been partially defined for most organisms. However, given the increasing number of microarray measurements, it may be possible to reconstruct such pathways and uncover missing connections directly from experimental data. To achieve the identification of novel pathway members and their biological roles in selected pathways, we developed a novel algorithm called BN+1 which incorporates Bayesian network computations to expand networks to include potentially important biological interactors. BN+1 analysis enables the prediction of the most likely molecular interactors given some initial set of molecular entities (e.g. genes) and an existing biological dataset (e.g. gene expression microarray data).

The BN+1 approach was tested and characterized using synthetically-derived networks which can mimic some biological interactions. This approach was also applied to the analysis of the ROS pathway in *Escherichia coli* (partially described in Chapter 2) and B cell receptor signaling pathway in *Mus musculus*. This expansion procedure predicted many stress-related genes (e.g., dusB and uspE), and their possible interactions with other ROS pathway genes. A simple yet novel term enrichment method identified that biofilm-associated microarray data usually contained high expression levels of both uspE and gadX. The predicted involvement of gene uspE in the ROS pathway and interactions between uspE and gadX were confirmed experimentally using E. coli reporter strains. Genes gadX and uspE showed a feedback relationship in regulating each other's expression. Both genes were verified to regulate biofilm formation through gene knockout experiments. Furthermore, the approach was successful in identifying known and putative interactors with the Nf-κB subnetwork within the larger B cell receptor signaling pathway.

These data suggest that the BN+1 expansion method can uncover hidden or unknown genes for a selected pathway with significant biological roles. Our results demonstrate the power of BN+1-based pathway augmentation or expansion in synthetic, prokaryotic, and eukaryotic systems. Thus, the presently reported BN+1 expansion method is a generalized approach applicable to the characterization and expansion of other biological pathways and living systems.

4.2 Introduction

In this study, we explore how a biological pathway can be defined, and identify a set of methods to automatically learn a pathway from experimental data. Although many biological pathways have been described in the literature, these pathways likely represent only a small portion of the known underlying network of interactions. Recently, such pathway representations have been systematized in databases such as EcoCyc [23], RegulonDB [65], and KEGG [20]. The pathways represented in these databases are commonly used as a starting point (seed network) to analyze gene expression data and identify pathway activity using computational tools such as GSEA [66] and DAVID [67].

However, when an annotated pathway is used to analyze microarray gene expression data, the assumption is made that the ideal microarray derived network will be the same as that in the literature. This assumption may not hold since many pathways are defined based on observed protein-protein and protein-DNA interactions, metabolic fluxes, and subsets of particularly well-studied genes. Each of these factors may contribute to the substantial inconsistency between RNA-level microarray-based networks and currently defined pathways. Furthermore, the selected pathway representation may be incomplete and not include relevant regulator or effector molecules, thus necessitating computational prediction and subsequent validation. To address this issue, we introduce a method to systematically expand a pathway by identifying new genes that, from a gene expression perspective, better define the pathway itself.

Biological pathways have been constructed from the existing literature and annotation information using a wide range of methods [12,36,45,68,69,70,71,72,73]. One method of pathway reconstruction uses Bayesian networks (BNs) to learn and model relationships between variables (e.g., genes). Bayesian networks are graphical models that describe causal or apparently causal interactions between variables. In this study, a Bayesian network is defined as a set of interactions (edges or arrows) between variables (nodes) selected from a set of known pathway genes. High scoring BN topologies are learned from data based on scoring metrics such as the BDe scoring metric introduced by Cooper et al. in 1992 [35], that incorporates the joint probabilities for variables connected to one or more other variables. In this context, the Bayesian model is a multinomial model with a uniform Dirichlet prior. Bayesian networks such as these have been used to identify relationships from gene expression data [36,46], protein-protein interactions[74,75], and the regulation of phosphorylation states [49]. Due to their flexibility, reliability, ability to model multi-variable relationships, and human interpretability, Bayesian networks are well suited for network modeling using high-throughput data such as gene expression microarrays.

Networks learned from datasets such as gene expression data can be used to expand our knowledge about a known pathway, by independently testing the effects of added genes or variables on the overall scores of the corresponding expanded networks. A general network expansion framework to predict new components of a pathway was suggested in 2001 [76]. Many of the pathway expansion methods use correlation or Boolean functions [10,76,77,78]. Compared to these methods, Bayesian network-based expansion methods provide distinct advantages, including prediction of both linear and nonlinear functions, identification of causal influences representing interactions among genes. Bayesian network-based expansion was also used for gene expression data analysis [11,53]. However, these expansion approaches are module-based methods that focus on identifying modules (or groups) of additional genes to one gene [11] or a group of genes with a fixed topology [53]. The mRNA-based networks were also merged with protein data which often do not agree with each other [53]. The topology of the biological pathways may not be consistent with networks learned from transcriptional gene expression data obtained via DNA microarray studies [77].

4.3 The BN+1 Algorithm

```
BN+1 Algorithm

Input: $N$ variables (e.g., genes) from a dataset (e.g., microarray dataset) with $L$ observations each.

Data Preprocessing (Optional)

Filter out $m$ variables (e.g., via coefficient of variation (c.v.) <= 1.0).  Number of possible variables for analysis: $N= N-m$.

BN Core Network Searching

Select $K$ variables from the set of $N$ variables (e.g. from a pathway database).

Construct matrix data file $D$ with $K*L$ observations using $K$ variables and $L$ observations.

Select settings for BN simulation, including data discretization (e.g. q3 quantization), searcher strategy (e.g. simulated annealing), and structural priors.

Execute BN simulation (e.g. using BANJO).

Save top BN network topology $C$

Iterative Core Expansion

Assign the core topology $C$ as unfixed structural prior for BN searching

For each variable $a$ in the set $\{N-K\}$, do:

        Generate new data file $D*$ by concatenating $L$ observations for $a$ to data file $D$

        Select settings for BN simulation.

        Execute BN simulation.

        Save top network and its posterior probability for $a$.

Rank each variable according to posterior probability.

Output: Rank-ordered BN+1 results.
```

Figure 4.1 Pseudocode for the BN+1 expansion algorithm.

The pseudocode for the BN+1 algorithm is represented in Figure 4.1 and its implementation is described as follows. First, Bayesian networks are generated from discretized microarray data and ranked according to log posterior score. A consensus network was then generated from the top-scoring networks and used for comparison with known pathway. Next, a top network used to generate the consensus network was randomly selected as a seed network for subsequent expansion. Each gene not included in the top network yet appearing in the microarray dataset was independently tested for its ability to acquire the best log posterior score versus the other tested expansion genes. BN+1 variables were ranked according to the best posterior score of their respective networks as compared to the other BN+1 variables. This approach was repeated for several distinct biological cases studies and is described below.

4.4 Case Study #1: Synthetic Network Analysis Using BN+1

4.4.1 Summary

To further establish the validity and evaluate potential pitfalls of the algorithm, a synthetic regulatory network was developed for testing the BN+1 algorithm. In terms of the previous ROS pathway analysis, the second most highly-ranked BN+1 gene appearing in the PLoS ONE paper, formate dehydrogenase *fdhE*, is further elucidated. Finally, cutoff criteria for selecting significant BN+1 genes and methods to improve the algorithm are discussed.

4.4.2 Method

A synthetic network was constructed by generating a set of mathematical functions which define the relationships amongst a set of variables (Fig. 4.2). In this model, eight variables are linked together in tandem (Fig. 4.2A) by the following functions:

$$\mathrm{A} = N(0,5) \tag{4.1}$$

$$\mathrm{B} = abs(10\log(abs(A)) + N(0,0.3)) \tag{4.2}$$

$$\mathrm{C} = abs\left(5e^{\left(\frac{-B}{15.0}\right)} + N(0,0.3)\right) \tag{4.3}$$

$$\mathrm{D} = abs(6.0/(C+1) + N(0,0.3)) \tag{4.4}$$

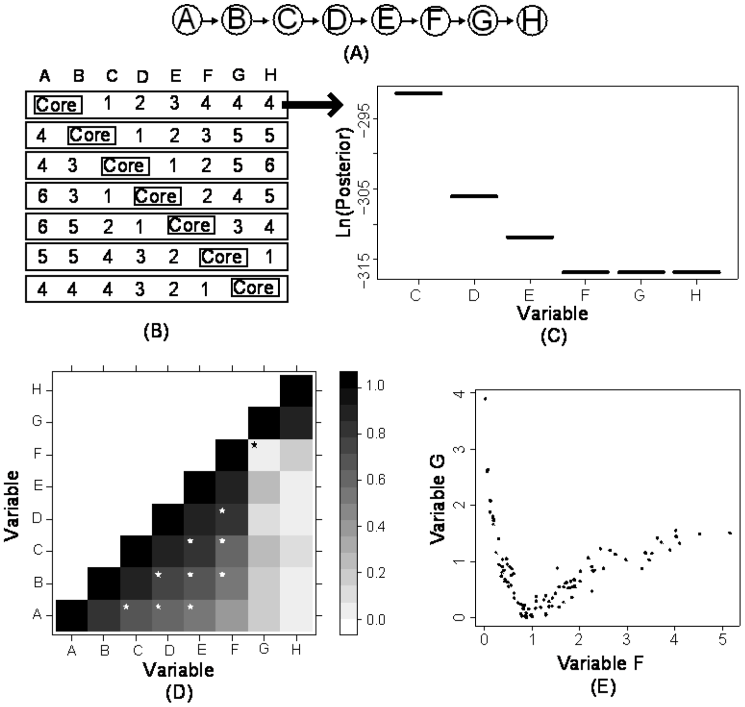
$$\mathrm{E} = abs(\log(D) + N(0,0.15)) \tag{4.5}$$

$$\mathrm{F} = abs(E^3 + N(0,0.3)) \tag{4.6}$$

$$\mathrm{G} = abs(\log(F) + N(0,0.17)) \tag{4.7}$$

$$\mathrm{H} = abs(6.0/(G+1) + N(0,0.3)) \tag{4.8}$$

where $N(\mu,\sigma)$ represents normally-distributed noise with μ as the mean and σ as the standard deviation. Biological data frequently include noise which can reduce the predictive capability of BNs and other modeling approaches. To reflect this reality, various levels of noise are added to the functional relationships. The function *abs*( ) is the absolute value of the enclosed quantity. Synthetic data were generated from these functions by sampling from the Gaussian-distributed variable *A* and subsequently sampling corresponding data values for subsequent variables in the pathway based on the above functions (a similar approach appears in [58]). This particular synthetic network contains different types of relationships amongst variables, e.g., nonlinear polynomial and biphasic relationships.



A	B	C	D	E	F	G	H
Core		1	2	3	4	4	4
4	Core		1	2	3	5	5
4	3	Core		1	2	5	6
6	3	1	Core		2	4	5
6	5	2	1	Core		3	4
5	5	4	3	2	Core		1
4	4	4	3	2	1	Core	

Figure 4.2 Synthetic network and BN+1 results for two-variable core expansion. (A) A synthetic eight-variable network. (B) Seven distinct core networks composed of two adjacent variables were used for BN+1 expansion analysis. In each row, integers identify the ranks of the BN+1 variables (where 1=top scoring gene, etc). (C) The posterior score distribution of BN+1 variables identified in the first row of Fig. 1A. (D) Plot of absolute values of pair-wise Pearson correlations for all variables. The black star denotes a relationship (between F and G) that has a poor Pearson correlation (coefficient = 0.056). White stars denote good relations between variables with correlation coefficient □ 0.5 and separated by at least one variable in the synthetic network (Fig. 1A). (E) A nonlinear relationship between variables F and G.

To further evaluate the BN+1 algorithm, a series of BN+1 simulations were designed and analyzed (Fig. 4.2B). In each simulation, two adjacent variables from the synthetic network are selected as a ‘core’ network (i.e., a known seed subnetwork) and used to identify the other six variables in terms of their roles in the overall network. The predicted variables, which are coined the BN+1 variables, are ranked according to their best log posterior scores obtained for the network containing a BN+1 variable and core network variables. This experiment was repeated for each pair of core variables in the model (Fig. 4.2B).

4.4.3 Results

When the core sub-network is located at the end of the synthetic network (i.e., A→B or G→H), the BN+1 successfully identified those variables that are closely associated to a core network in sequential order (Fig. 4.2B). For example, when the core network is A→B, BN+1 identifies the top four variables that are associated with this core network are C, D, E, F, in correct order. It is interesting that the last two variables G and H have the same score as F when they are individually added to the core network (Fig. 4.2C). A further examination indicates that none of the three variables F, G, and H is connected to A or B in the final BN network containing A, B, and one of the three variables. The disconnection of these three variables from the core A→B makes it possible for the posterior probabilities to be the same.

When the core subnetwork is located in the middle of the synthetic network (e.g., B→C or C→D), the variables identified by BN+1 are ranked in sequential order in either side of the core network. For example, for the core network C→D, the BN+1 variables on the right side are ranked 1 (E), 2 (F), 5 (G), and 6 (H), and the BN+1 variables on the left side are ranked 3 (B), and 4 (A) (Fig. 4.2B). It is interesting that top 2 (F) is located on the same side with top 1 (E) instead of direct association with C in the C→D network. Despite the direct link between B and the core network, F has stronger association (with higher posterior probability) with the core network than B. This asymmetric pattern suggests that top ranked BN+1 variables are ranked based on their extent of associations with the core network instead of physical closeness to the core network.

4.4.4 Discussion

One advantage of BN+1 over many linear correlation-based methods is that our Bayesian network based approach is able to identify those interactions that show nonlinear correlations with core variables. Pearson correlation is a typical method for defining the extent of a monotonically increasing or decreasing relationship between the variables [79]. The correlation coefficients between all possible pairs in the original dataset were calculated using Pearson correlation method. Although all of the functions are nonlinear, over the rage of parameters tested, some may be approximately linear, while others may be more strongly nonlinear and deviate from monotonicity. Figure 4.2D shows a matrix representation of the Pearson correlations observed for each pair of variables and their synthetically-generated data. In general, Pearson correlation coefficients decrease as the distance between variables (or the distance of one variable from the diagonal of the matrix) increases. Overall, Pearson correlations can not only detect those variables directly associated with one specific variable, but also identify those that are remotely associated with sequential order (white stars in Fig. 4.2D). However, Pearson correlation failed to identify the association between F and G. A further examination indicates that F and G share a clear nonlinear relationship (Fig. 4.2E). Such a nonlinear relationship is correctly detected by BN+1 (Fig. 4.2B).

In our synthetic data simulation, we found that the disconnected variables share the same score (Fig. 4.2). This cutoff shows that all subsequent BN+1 genes will be disconnected from the core gene network. Similar results were also observed in the ROS pathway simulation (further described in Section 4.5). The last 1,457 genes in the sorted BN+1 gene list were all disconnected from the core gene network. This suggests that these 1,457 genes have no relationship with the ROS pathway based on the selected microarray data and selected core network. However, the cutoff based on the loss of connection between a BN+1 gene and a core network is loose and may result in too many genes being included for further testing.

4.5 Case Study #2: BN+1 Analysis of the *E. coli* ROS Pathway

4.5.1 Summary

We hypothesize that Bayesian networks derived from microarray gene expression data are largely consistent with known pathway models and can be used as a basis to predict novel factors that influence a given pathway. In this study, the hypothesis was examined using the *Escherichia coli* reactive oxygen species (ROS) pathway. Because *E. coli* and the ROS pathway had been well studied [25,26,27,28], we were able to test the effectiveness of our network expansion algorithm and to assess the ability to reconstruct and expand an accepted pathway using microarray data. We identified many stress-related genes potentially involved in the ROS pathway and predicted their interactions with known ROS genes. Our prediction was confirmed experimentally for one example gene, *uspE*. Our single-gene expansion approach, termed ‘BN+1’, was successful in predicting unknown stress interactions that can be verified through experimental analysis, and could demonstrably be applied to other biological systems of interest.

4.5.2 Methods

4.5.2.1 Data preprocessing

A compilation dataset comprising 305 gene expression microarray observations and 4,217 genes from *Escherichia coli* MG1655 was obtained from the M3D database [8]. A coefficient of variation threshold (c.v. ≥ 1.0) was used to select 4,205 genes for analysis. Twenty-seven genes were identified from the EcoCyc ROS detoxification pathway (downloaded on March 26, 2008) and matched to unique features found in 305 available gene expression microarray chips. Expression profiles for each gene were discretized using a maximum entropy approach that uses three equally-sized bins (q3 quantization).

4.5.2.2 Learning Bayesian network pathway models

Given the set of 27 genes, Bayesian network analysis was used to learn the structure of the model which served as our core starting topology. To maximize the network search space, 4000 independent simulations with random starts were used to search $2.5\text{x}10^{7}$ networks per start for a total of $1\text{x}10^{11}$ networks. The five top networks were saved from each run, thereby generating a final list of 20,000 top-scoring networks. These networks were used to estimate the posterior distribution. During the search, each network was scored using log of the BDe score [35,37] which is the natural log of posterior probability ($S = \ln(P(M \mid D))$) and is defined previously in Equation 2.2 in Chapter 2 using the software package BANJO [40].

A consensus network was generated using 33 networks which shared the maximum or best log posterior score (ln(P(D|M)). Specifically, directed edges in the consensus networks represent those edges that appear with 100% frequency in one direction in all of these top networks. Undirected edges represent those edges appearing 100% of the time in both directions in all stored networks (Figure 4.3).

4.5.2.3 Network expansion using BN+1

To expand an existing network, a top network used to generate the consensus network was used as a starting topology for the BN+1 algorithm (Figure 4.3). A set of 4,178 genes (4,205-27), not included in the top BN, were tested for their ability to improve score of the initial core BN when added to the initial gene set. In each iteration of the BN+1 simulation, the current BN+1 gene was added to the original data file. This was followed by a simulated annealing search of 1x10^7 networks for the top network expansion. Although the top network was selected as a starting point or seed, during the learning round all edges could be modified such that the addition of genes could change the backbone structure of the resulting model (i.e., unfixed structural prior). Genes were sorted based on their log posterior scores. BN+1 searches for each of the top 200 genes recovered from the initial top network were rerun (2.5x10^7 networks/simulation with 150 replicate simulations) to allow sufficient convergence.

All calculations, including the network expansion, were implemented in a publicly available, internally developed software program MARIMBA (available at http://marimba.hegroup.org/, described in Chapter 6).

4.5.2.4 Term enrichment for identifying relevant experimental observations

A term enrichment program was developed to identify which descriptive terms in the experimental conditions show significant enrichment in selected regions of the microarray data. A 'term' here is defined as any individual word appearing in the names or descriptions for each microarray sample. For two selected genes, a p-value was introduced to determine the chance of observing a selected term in a selected bin. The p-value was calculated using the Fisher's exact test for appearance of 'term' and 'non-term' data observations in a specific bin [80]. The bins used for microarray BN analysis were adopted in this text enrichment analysis. For example, the q3 quantization was used for the expression levels of *gadX* and *uspE*.

4.5.3 Results

4.5.3.1 Microarray-based Bayesian network overlapped with known ROS pathway

Using a compendium of microarray gene expression data from the M3D database [8], networks were constructed for the 27 genes contained in the ROS pathway as defined by the EcoCyc database [23] (Figure 4.3). *E. coli* uses a complex detoxification pathway to protect against the oxidative stress posed by reactive oxygen species (ROS), including oxygen ions, free radicals, and peroxides [27]. The 27 genes identified in the EcoCyc ROS pathway include five ROS-processing enzymes (i.e., *katE, katG, sodA, sodB, sodC*) and 22 transcriptional factors that regulate transcription of these ROS-related enzymes. This *E. coli* expression dataset incorporates a variety of experimental conditions including time course studies, cell stress-inducing environments, over-expression, and single and double knockout strains. These conditions perturb the ROS pathway and provide a reasonable data set for the evaluation of our hypothesis. To include all results predicted from the top Bayesian networks, a consensus network was derived using the 33 top networks that shared the best identical posterior probability. The consensus network contains all 27 genes from the original ROS detoxification list in EcoCyc.

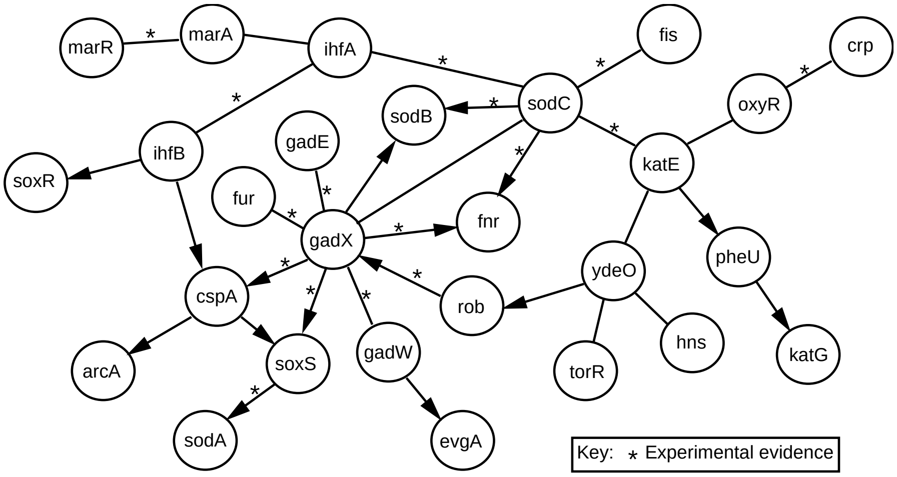


Figure 4.3 Consensus network for the ROS detoxification pathway. Bayesian networks were generated using twenty-seven genes from the reactive oxygen species (ROS) detoxification pathway as variables or nodes and 305 gene expression microarray observations per variable. Edges which appear in the consensus and are supported by external data (e.g. EcoCyc, RegulonDB, and/or literature) are indicated (*).

A comparison of the consensus network to EcoCyc revealed that 29% of the edges in the consensus are supported by corresponding edges in EcoCyc [23] or RegulonDB [81]. However, inclusion of literature information in the comparison revealed that approximately 42% of the edges found in the consensus network were confirmed. The difference suggests that some new literature results have not been collected in current databases such as EcoCyc and RegulonDB.

4.5.3.2 BN+1 pathway expansions predict ROS-related genes and gene interactions.

An expansion algorithm termed BN+1 was developed to identify those genes that provide the best network score when added to an existing core network topology (Figure 4.1). Each gene not yet included in the core network is individually added to the set of variables for the Bayesian network simulation (hence Bayesian network plus one gene, or 'BN+1'). The edges in the initial core network topology are used as a 'structural prior' or starting point, and are allowed to change over the course of the BN simulations. The added node is initially disconnected from the existing core network and can become connected to other variables over the course of the simulation. Those genes which best improve the network score when added to the existing core are expected to have the most direct biological influence and/or relevance to the core network genes.

The BN+1 expansion algorithm was used to identify additional potential members of the ROS detoxification pathway. The top-ranked results from these analyses are shown in Table 4.1. The algorithm identifies whether a gene is strongly associated with a particular network (e.g., the ROS detoxification pathway) and which genes in the network may influence or be influenced by the newly predicted gene. The predicted influences between core genes and the top "+1" genes (including *dusB* and *uspE*) identified by BN+1 expansion are shown in Figure 4.4.

Expansion of the consensus network revealed that many top predicted genes have known relationships with ROS and stress regulation (Table 4.1). The tRNA-

Table 4.1 Top 10 genes identified by BN+1 expansion of core network.

Rank	Top BN+1 gene hits	Posterior BN score

1	***dusB*** (tRNA-dihydrouridine synthase B)	S=-8295.81
2	***fdhE*** (formate dehydrogenase formation protein)	S=-8298.44
3	***uspE*** (stress-induced protein);	S=-8310.63
4	***yohF*** (predicted oxidoreductase with NAD(P)-binding Rossman-fold domain)	S=-8312.24
5	***yncG*** (predicted enzyme);	S=-8313.04
6	***msyB*** (predicted protein);	S=-8318.20
7	***yedP*** (conserved protein);	S=-8320.30
8	***sra*** (30S ribosomal subunit protein S22)	S=-8323.97
9	***ydcK*** (predicted enzyme);	S=-8325.91
10	***ynhG*** (conserved protein);	S=-8326.20

Note that the numbers shown after gene names are negative logs of posterior probabilities for each top network containing the respective predicted gene.

dihydrouridine synthase B gene (*dusB* or *yhdG*) was predicted to be the top-scoring BN+1 gene and to interact with *fis* and *sodC* (Figure 4.3A). *Fis* is an important regulator of oxidative stress [82]. Because all of the known enterobacterial *fis* genes are preceded by *dusB* (also called *yhdG*) within the same operon [82], it is reasonable that *dusB* is positioned as a parent of *fis* in our prediction. Both *fis* and *sodC* are crucial to bacterial defense against the deleterious effects of reactive oxygen species (ROS) [83,84]. The interaction between *sodC* and *dusB* is likely important for bacterial antioxidant reactions. The second top predicted gene *fdhE* encodes an *E. coli* formate dehydrogenase accessory protein that regulates the activity of catalytic sites of aerobic formate dehydrogenases and their redox activities [85]. A third gene, the universal stress protein *uspE*, is a known major regulator of motility factors and cell aggregation under stress conditions [86]. Several other predicted enzymes (*yncG* and *ydcK*) and proteins (*msyB*) found in the

BN+1 search have no currently known functions related to the ROS pathway and stress response.

Pair-wise plots of the expression of BN+1 genes versus ROS pathway genes show simple (*dusB* vs *fis,* Figure 4.3A) or complex relationships (*uspE* vs. *gadX*, Figure 4.4B-C). The plots show that the relationships between these genes may be nonlinear. For example, a “V” shaped pattern is observed between the expression profiles of *gadX* and *uspE,* where *gadX* is down-regulated at moderate levels of *uspE* and up-regulated in either increased or decreased levels of *uspE* (Figure 4.4C). This special non-linear gene interaction pattern was not clearly demonstrated in a traditional hierarchical clustering heatmap (Supplemental Figure 1 in [56]). Gene *gadX* is a transcriptional regulator of glutamic acid decarboxylase system, which enables *E. coli* to overcome acidic stress, while *uspE* is a universal stress-induced protein. A term enrichment method was generated to identify words that are preferentially grouped and reflect most significant features of the interactions between two genes (e.g., *gadX* and *uspE*) as predicted by our BN method.

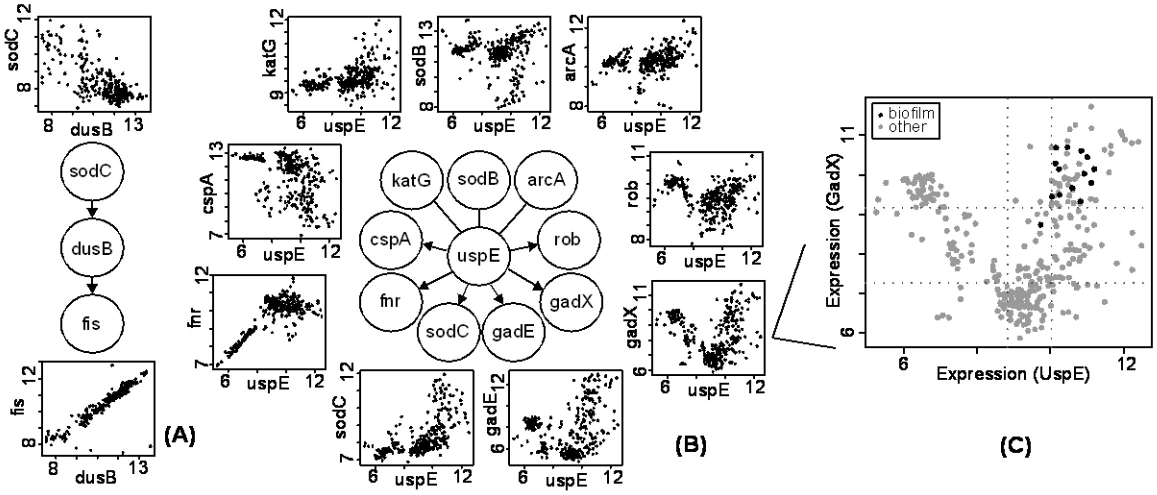


Figure 4.4 Top BN+1 predictions and their relationships with core network genes. Genes dusB(A) and uspE (B) were top results for large network expansion. (C) Scatter plot for uspE versus gadX highlighting experiments with the word "biofilm" in the experiment title and/or description. High levels of uspE and gadX were observed for all conditions mapped to 'biofilm'. The dotted lines indicate boundaries for binning used in network learning. A similar profile was observed for gene gadE (not shown).

Based on our term enrichment analysis of *gadX* and *uspE*, one term that clustered the data particularly well was "biofilm", which was demonstrated in the annotated scatter plot (Figure 4.4C). High expression of *gadX* was correlated with high expression of *uspE* in biofilms. Biofilms are aggregates of microorganisms that attach to and grow on a surface in contact with liquid, such as water or media. Induced expression of stress response genes, e.g., a universal stress regulater *uspA*, was a general feature of biofilm growth [87,88]. In fact, the biofilm microarray data used in the term enrichment were obtained from two studies. One study analyzed stress-oriented gene expression profiles of *E. coli* biofilm at various time points [89]. A second biofilm microarray study examined biofilm responses to acid resistance and oxidative stress using wild type and single gene knockout mutant strains of *E. coli* [90]. Our combined analysis of microarray gene expression and term enrichment indicated that *uspE* and *gadX* were both up-regulated in many samples (chips) where ‘biofilm’ was mentioned in the sample title and/or description (Figure 4.4B-C). These suggested a potential role of the *uspE* and *gadX* in the formation of *E.coli* biofilm.

To further evaluate the interactions between *uspE* and *gadX* and their regulatory roles in ROS stress and biofilm formation, several wet-lab experiments were conducted by Dongjuan Dai and Chuanwu Xi. These results, appearing in [56], verified (1) the interactions predicted between *uspE* and *gadX* do exist in *E. coli*, (2) their responses to hydrogen peroxide stress and further implication in ROS activities, and (3) their direct control of biofilm-related activities. Thus, the BN+1 approach is successful and can identify novel pathway members, biological interactions, as well as functional relevance.

4.5.3.3 The challenge of identifying meaningful BN+1 cutoffs

After all genes are ranked by the BN+1 simulation, what cutoff should be used to select the top ranked BN+1 genes for further analysis? While the top few BN+1 genes prove important in the ROS pathway, many more shown in the list of top BN+1 genes are also related to ROS pathway (not shown). Our Gene Ontology (GO) enrichment analysis of the top 100 genes in the sorted BN+1 gene results (~2.4% of the total genes on the microarray) showed that they were enriched for ROS-related activities or functions (results and related discussion appear in [56]). This means that a certain number of top-scoring BN+1 genes are all related to the core gene pathway. However, if only the posterior scores are considered, the scores for the selected pathway tend to decline or drop off quickly before smoothing out after a small number of the top genes (Fig. 4.5A).

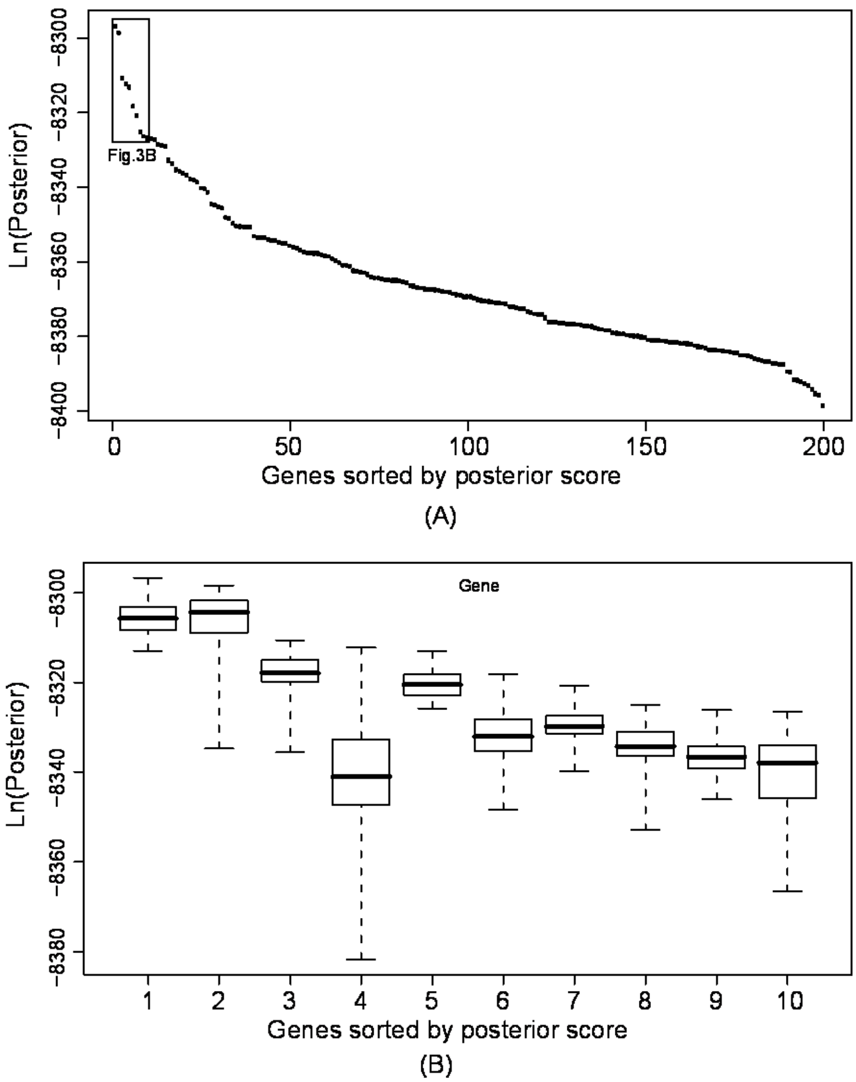


Figure 4.5 Analysis of top BN+1 genes in the ROS use case. (A) Generic plot of best score for top 200 BN+1 genes. (B) Variation in scores for top 10 genes. The BN+1 genes are ranked by maximum scores of all networks containing the core genes plus one additional gene. Genes sorted by posterior scores are shown in horizontal axis. Box plots for the set of scores pertaining to each gene are displayed. The variations are calculated based on various simulations in different computers. To perform each simulation, a simulated annealing approach was used with an unfixed structural prior (i.e. the core network edges) with multiple replicates and moderate simulation time to allow a comprehensive though non-exhaustive search.

One feasible criterion is based on the possible loss of connection between a BN+1 variable and the core network. In our synthetic data simulation, we found that the disconnected variables share the same score (Fig. 4.2). This cutoff shows that all subsequent BN+1 genes will be disconnected from the core gene network. Similar results were also observed in the ROS pathway simulation. The last 1,457 genes in the sorted BN+1 gene list were all disconnected from the core gene network. This suggests that these 1,457 genes have no relationship with the ROS pathway based on the selected microarray data and selected core network. However, the cutoff based on the loss of connection between a BN+1 gene and a core network is loose and may result in too many genes being included for further testing. For example, in our ROS example, 2,760 genes remain after the last 1,457 genes are excluded. While the loose cutoff removes roughly a third of the genes, there are still many genes which may or may not closely relate to the ROS pathway network.

To make a tighter and possibly more useful cutoff, we analyzed the distribution of sorted posterior scores. In the ROS analysis, the sorted posterior scores of BN+1 genes quickly drop across the first ten variables, followed by a slowdown of score dropping (Fig. 4.5A). Therefore, it is possible to suggest a cutoff in the beginning of the slowdown of score dropping. However, these cutoffs are still artificial because we do not know which one(s) would be optimal for maintaining the real biological predictions. Furthermore, the “best” posterior probabilities of BN+1 variables’ networks often have variations across large amounts of simulations in different computers (Fig.4.5B). Current variable rankings are based on the highest log posterior scores among all simulated networks for the selected BN+1 variable and core variables. Multiple scores may be obtained and saved for a selected BN+1 variable and core variable set. If the median scores for each set of BN+1 results were used instead, the rankings of BN+1 genes could change (e.g. the 4th and 5th genes in Fig. 3B). It is unlikely the median scores would ever be used since the BN optimization approach always seeks the best (or most optimal) result. The resulting variation is probably due to the failed achievement of convergence. To achieve a final convergence, more execution time will be needed. More compute time will reduce the variation in scores for each individual BN+1 variable and improve our overall confidence in the rankings of the BN+1 variables. Because our synthetic data use case only have eight variables, it is relatively easy to achieve convergence. For example, Fig.1C shows no score variation in replicates for each of the BN+1 variables in our synthetic network (hence the box plots appear as lines denoting the median score), suggesting sufficient convergence was achieved by the algorithm.

To make the experimental testing more meaningful, an empirical cutoff such as the top 10% of the score distribution or top 100 genes may be helpful. Although this type of cutoffs is heuristic and does not establish the statistical significance of those results, subsequent exploration of the top BN+1 results based on this cutoff may still lead to novel discoveries [56].

4.5.4 Discussion

In this study, we addressed two questions: (1) Does a microarray-based Bayesian network reconstruction match with the known pathway from the literature and existing database? (2) Is a network expansion approach such as BN+1 useful in predicting new, biologically significant genes?

For the first question, our studies indicated that the microarray-based Bayesian network reconstruction did not always agree with the known pathway from the literature and databases. Our studies on the *E. coli* ROS pathway indicated that the network reconstructed by our Bayesian network overlaps at 29% with the known ROS pathway network in EcoCyc and RegulonDB. A 42% agreement was achieved when more evidences from the literature search was included. Inclusion of RegulonDB and literature resources made our comparison more comprehensive. The reason for the large mismatch is probably due to the fact that microarray-based transcriptional data may not reflect the complex biological pathways which involve complex interactions of genes in the protein, RNA, and DNA levels [91]. However, the Bayesian networks built from microarray gene expression data are transcriptional regulatory models that are predicted to reflect the complex ROS pathway.

For the second question, the BN+1 expansion algorithm was found to successfully predict biologically significant genes to the ROS network that were further experimentally verified. Gene *uspE* was one of the top list genes selected by the BN+1 algorithm. Its up-regulation in response to the exposure of hydrogen peroxide suggested that this gene was probably involved in the ROS network, along with the ROS-related gene *gadX* (Figure 4.4). Hierarchical clustering of the *uspE* gene showed a different connectivity pattern in the dendrogram for genes than the Bayesian network, suggesting that the Bayesian network identified a non-traditional (*e.g.* nonlinear) relationship between the genes. Furthermore, the BN+1 algorithm suggested where the new genes could participate in the pathway, and in some cases the model even differentiated between the parents and children genes of a new gene (Figures 4.3-4). Specifically, the BN+1 algorithm found the "V" shape relationships between expressions of genes, e.g., *gadX* and *uspE,* which would not have been identified using traditional clustering approaches. The interaction between gene *gadX* and *uspE* was also confirmed experimentally. Expression of one gene was significantly affected when the other gene was knocked out from the wild type *E. coli* strain (Figure 4.4). Plot of the expression of *gadX* and *uspE* against each other under different tested experimental conditions showed a similar "V" shaped pattern (Figure 4.4), which was in agreement with the finding using the BN+1 algorithm although the expression data from the experimental study were at the translational level.

The term enrichment algorithm successfully identified experimental conditions in which genes might be involved and biologically related with each other. In this study, genes *uspE* and *gadX* were founded to be both up-regulated in the growth of biofilms. The involvement of the two genes in biofilms was confirmed by the fact that single gene knockout mutant strains Δ*gadX* and Δ*uspE* showed difference in the biofilm formation, either biomass or structures, as compared to the *E. coli* wild type strain (shown in [56]). Experimental confirmation of predicted term enrichment results indicates that term enrichment algorithm is a useful method to identify experimental conditions in which gene relationship may take place, or to propose additional areas of investigation. Performance of the term enrichment approach likely depends upon the quality of the experimental descriptions provided by researchers available from the M3D database. The approach may perform better with controlled term or concept vocabularies, or could be further tested with Gene Ontology (GO) terms and other information in future studies.

Bayesian network can be used to expand a pathway network based on microarray gene expression data. The BN+1 method expands a top Bayesian network by adding one gene at a time and running it iteratively based on microarray gene expression data. The BN+1 expansion algorithm showed the ability to predict important factors for a pathway network from thousands of genes in a microarray study. The BN+1 approach is a generalized method to refine and expand biological pathways. Although a ROS pathway in *E. coli* was shown in this study, the BN+1 algorithm can readily be applied to other organisms, pathways, and data types. Furthermore, the text enrichment-based identification of experimental conditions in the context of binned data for BN analysis can provide beneficial information in the interpretation of predicted expansion genes.

4.6 Case Study #3: BN+1 Analysis of the Murine BCR Pathway

4.6.1 Summary

Signalling and regulatory pathways that guide gene expression have only been partially defined for most organisms. Given the increasing number of microarray measurements, it may be possible to reconstruct such pathways and uncover missing connections directly from experimental data. One major question in the area of microarray-based pathway analysis is the prediction of new elements to a particular pathway. Such prediction is possible by independently testing the effects of added genes or variables on the overall scores of the corresponding expanded networks. A general network expansion framework to predict new components of a pathway was suggested in 2001 [76]. Many machine learning approaches for identifying hidden or unknown factors have appeared in the literature recently [10,11,53,54,55,76,77,78,92].

The BCR pathway is an integral component of the adaptive immune response mechanism by which B cells respond to foreign antigens [29]. The BCR pathway involves in the activation of specific protein kinase C (PKC) isoforms that induces ultimate activation of the NF-κB transcription factor. Multiple protein species accumulate at the cell membrane in a signalosome complex and are linked to the B cell receptor. Signal propagation from the BCR via kinase-mediated phosphorylation cascades to downstream effectors such as Nfkb, NFAT (nuclear factor of activated T cells), and AP1 is either enhanced or reduced via signalosome interactions with co-stimulatory or co-inhibitory complexes, respectively. BCR signaling guides many important functions such as anergy, B cell ontogeny, and immune response, and is linked to the several imporant pathways: MAPK, coagulation/complement cascades, and actin cytoskeleton [1,19]. NF-κB plays a crucial role in the antigen-induced B lymphocyte proliferation, cytokine production, and B cell survival [29].

We have recently developed an algorithm termed “BN+1” which implements Bayesian network expansion to predict new factors and interactions that participate in a specific pathway [56,57]. This algorithm has been tested using *E. coli* microarray data [56] and verified with synthetic networks [57]. BN+1 is applied in this chapter towards understanding NF-κB transcriptional regulation and interactions within the BCR signalling pathway.

4.6.2 Method

We used gene expression data from perturbed B-cells obtained from the Alliance for Cellular Signaling (AfCS) [93,94]. This dataset is especially attractive because the same tissues were treated with combinations of ligands that perturb different B cell pathways. The AfCS study gathered 424 microarray chips measuring gene expression in B cells from *M. musculus* splenic extracts that are exposed to 33 different ligands [93,94,95]. Briefly, B cells purified from splenic preparations from 6- to 8-wk-old male C57BL/6 mice were treated in triplicates or quadruplicates with medium alone, or one of 33 different ligands for 0.5, 1, 2, and 4 h (AfCS protocol PP00000016). RNA was extracted following standard AfCS protocol PP00000009. An Agilent cDNA microarray chip that contains 15,494 cDNA probes printed on 15,832 spots was used. It represents 10,615 unique MGI gene matches [93]. Each Agilent array was hybridized with Cy5-labeled cDNA prepared from splenic B cell RNA and Cy3-labeled cDNA prepared from RNA of total splenocytes used as an internal reference (AfCS protocol PP00000019). Hence, each Agilent microarray chip provides one unique observation of relative expression level per selected probe. The arrays were scanned using Agilent Scanner G2505A, and images were processed using the Agilent G2566AA Feature Extraction software version A.6.1.1. The microarray raw data were downloaded from the AfCS repository at ftp://ftp.afcs.org/pub/datacenter/microarray/.

Microarray data were discretized for each variable in the Bayesian networks using quantile normalization with three bins. Though triplicate or quadruplicate microarray experiments were available in most cases per unique treatment and time of drug administration, we assume that each experiment provides an independent source of information. In this analysis, we did not use all BCR pathway genes. We sought to answer here whether expansion of a sub-network from the BCR pathway would preferentially recover other BCR pathway genes. This assumption is advantageous in that the number of variables allows significantly faster simulation searches for the BN and

BN+1 simulations. Particularly, those genes most specifically involved in Nfkb-mediated transcriptional regulation were chosen from the KEGG BCR pathway.

A set of 10,000 top-scoring BNs was generated using the eight variables (the core) and 424 observations. Among the eight variables, two variables are Nfkbie (IκB) probe sets, and two are Ikbkb (IKK) probe sets. In many cases, one gene has multiple probe sets. We chose to separate them as different variables in our BN analysis since often these probe sets have different values with low correlation (Fig. 4.6). This BN analysis was accomplished by running 100 independent simulations and saving the top 100 simulations for each of those runs.

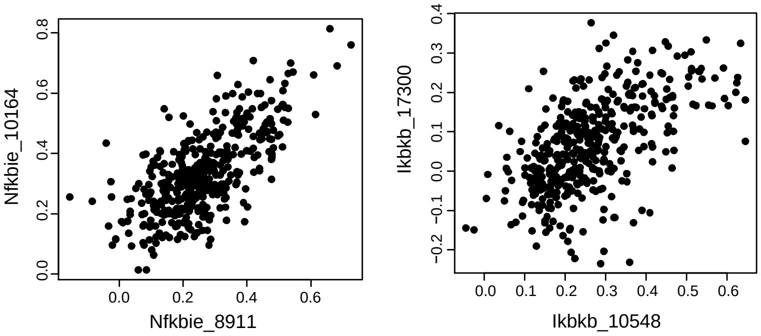


Figure 4.6 Scatter plots for Nfkbie and Ikbkb probes from AfCS study. Agilent probe identifiers are listed next to each respective gene. This figure indicates that the probe sets Nfkbie_10164 and Nfkbie_8911 correlate relatively well with a Pearson correlation coefficient of 0.69 (A). However, the correlation between Ikbkb_17300 and Ikbkb_10548 is low (Pearson correlation coefficient: 0.58) (B).

4.6.3 Results

Figure 4.7 depicts the shared set of interactions appearing in all of the top networks sharing the same best score. Compared with the KEGG BCR pathway, the consensus network found in our BN analysis (Fig. 4.7) has an overlap with 75% of correlation (3 out of 4 were correctly predicted), with only one interaction missing (Fig. 4.8).

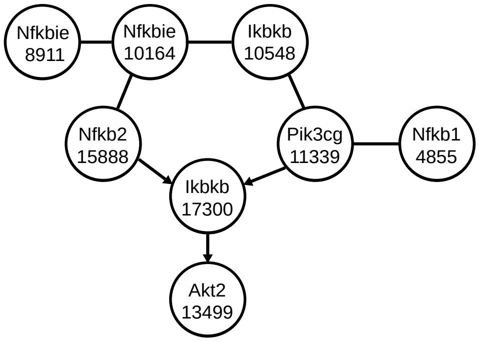


Figure 4.7 Consensus of top scoring Bayesian networks for eight probes representing BCR receptor signaling pathway genes. Gene symbols and corresponding Agilent probe identifiers are represented in nodes in the network. Directed edges represent those influences appearing in the same direction in all top-scoring Bayesian networks, while undirected edges appear at least once in the opposite direction though appearing cumulatively with 100% frequency in all of the top networks.

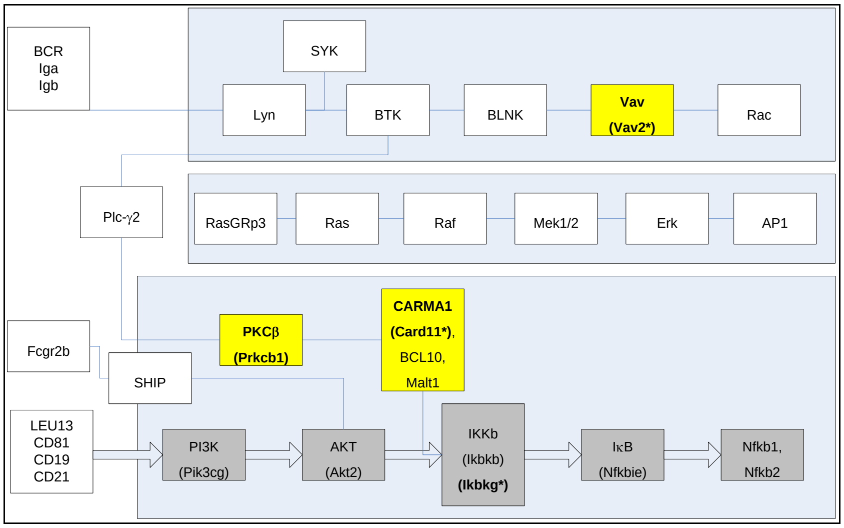


Figure 4.8 Schema of the BN+1 analysis results compared to KEGG BCR pathway. The three blue boxes represent three major sub-networks within the BCR pathway with distinct regulatory and functional roles. The BN core network was defined using members from the third sub-network (dark grey boxes) which reflect major components of Nfkb signalling. Bolded gene names are those genes which were not included in the core network, yet were recovered during BN+1 analysis in the top 100 results. Note that not all members of the listed Nfkb signalling pathway were included in the core network (e.g. Ikbkg), and in some cases were not available on the microarray platform.

4.6.3.1 Defining BN+1 genes

One of the top-scoring networks used to generate the consensus shown in Fig. 4.7 was used as a core network for subsequent BN+1 expansion. BN+1 searching was executed for 14,353 individual probes with 50 million networks searched per probe. If only those genes in close neighbourhood in the KEGG BCR pathway are considered, out of 19 selected genes, nine genes were found to be connected to the core network in our analysis. Furthermore, four of these nine genes are in close proximity (within top 10% of top-scoring BN+1 genes with at least one connection to the core network) with these core genes in the KEGG protein signalling pathway: Card11, Prkcb1, Ikbkg, and Vav2. These results suggest that the neighbourhood of transcriptional regulation around the core network as well as distance between the elements in the protein signalling pathway are related to each other.

Analysis of the top BN+1 variables recovered during simulation revealed several interesting results. First, the top set of BN+1 variables is listed in Table 4.2.

Table 4.2 Top ten predicted BN+1 genes. Identifier information for each ranked gene is provided, including Agilent probe ID (Agi_ID), Entrez gene ID (GENEID), and gene symbol. Probe variables from the core network which directly connect to the BN+1 variables in the top-scoring networks are listed in the "Neighbors" column.

Rank	Agi_ID	GeneID	Symbol	BN1_score	Neighbors
1	11062	77619	Prelid2	-3402.0	Nfkb2
2	9502	20744	Strbp	-3517.0	Nfkbie
3	14138	20823	Ssb	-3545.2	Nfkb2
4	6276	12530	Cdc25a	-3569.2	Nfkb2
5	11361	108829	Jmjd1c	-3586.8	Ikbkb(both), Pik3cg

6	14614	75964	Trappc8	-3587.8	Ikbkb, Pik3cg
7	15876	108786	Cxcl13*	-3593.1	Nfkb2
8	10759	73132	Slc25a16	-3594.8	Ikbkb, Pik3cg
9	5275	67887	Tmem66	-3596.0	Nfkb1, Pik3cg
10	9036	109339	2700018L05Rik	-3599.1	Pik3cg

Many interesting findings were observed from this analysis. Many genes, for example, the Sjorgen syndrome antigen B gene (Ssb) [96], has been proven to be associated with the Nf-kB and BCR pathways. Ssb plays an important role in polysome translation [96], and is an early DNA-damage responder in apoptotic cells and those treated with cytotoxic chemicals [97]. Interestingly, we identified Jmjd1c, a member of the jumonji family proteins, as a top predicted gene in our BN+1 simulation. Jmjd1c is conserved in several mammalian species and has documented roles in metal ion binding, oxidoreductase activity, and transcriptional regulation [98]. The murine Jmjd1c mRNA is expressed in multiple tissues, including hematopoietic and undifferentiated ES stem cells, fertilized egg, pancreatic islet, etc [98]. Jmjd1c has a promoter region orthologous to humans with binding sites for the AP-1 transcription factor, which is considered a member of the BCR signalling pathway and is included in the KEGG representation as AP1 (downstream of the Raf/MEK sub-network in Figure 4.8 though not in our core network. Fig. 4.9 illustrates the strongly-correlated relationships uncovered between the Jmjd1c genes and connected core network members. As another example, the Cxcl13 is a chemokine ligand in B cells with a C-X-C motif. It has already been established that Cxcl13 induction requires activation of canonical and non-canonical NF-κB pathways [99], which confirms the prediction of this gene in our network. These data strongly support the predictions generated by our analysis.

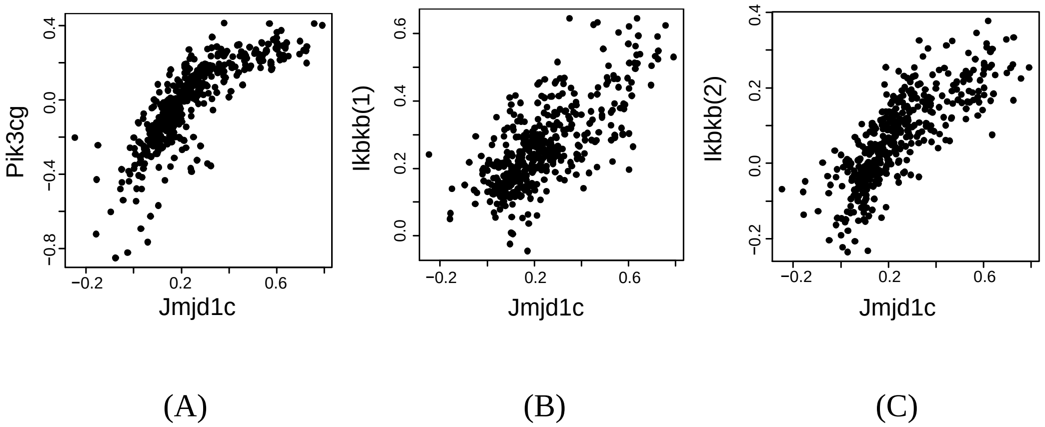


Figure 4.9 Scatter plot of expression values for core genes Pik3cg and Ikbkb (both probes) versus BN+1 gene Jmjd1c. A non-linear association between Pik3cg and Jmjd1c is observed (A). A roughly linear relation is observed between Jmjd1c and Ikbkb(1) (Pearson correlation coefficient: 0.71) (B) and between Jmjd1c and Ikbkb(2) (Pearson correlation coefficient: 0.79) (C).

One property of interest, as shown in the table, is that the core genes which recruit the top BN+1 genes are not always the same. From this analysis and previous studies, we have observed that BN+1 variables which show high correlations to at least one core network variable often appear as top BN+1 results. However, in some cases, the BN+1 variable may connect to multiple variables in the core network, and yet show moderate to low correlations with each of them. It is observed that many BN+1 variables have multiple core network variables as parent nodes in the predicted top network. Multi-parent relationships are less common though statistically more meaningful due to the nature of the implemented conditional probability tables in BDe scoring.

4.6.3.2 Clustering analysis of core genes and BN+1 genes

Different methods, such as clustering and GO gene enrichment, can be used to further analyze BN+1 genes. A clustering method provides a way to group BN+1 genes based on gene expression values. A heapmap clustering analysis was performed using 8 probe sets in the core network and 10 probe sets from the BN+1 analysis (Fig. 4.10). As shown in this heatmap, all NF-κB genes (core genes in our BN simulation) are clustered together, indicating their close association. Our analysis also found that Jmjd1c is closely associated with these NF-κB genes. This further strengthens our BN+1 prediction of the important role of this gene in the NF-κB pathway in B cell signalling.

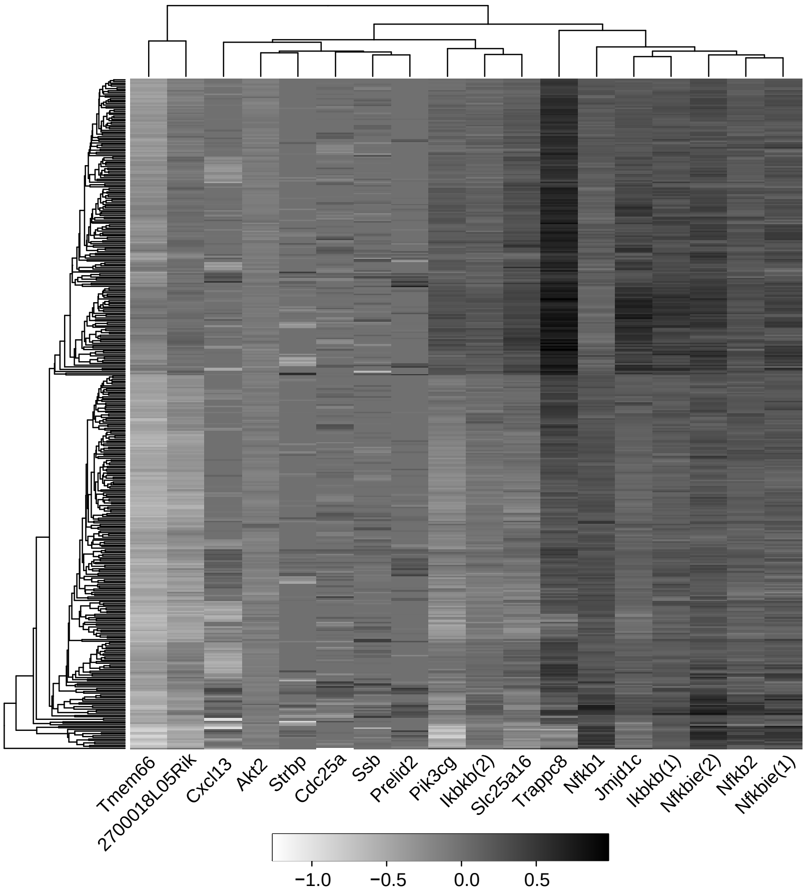


Figure 4.10 Heatmap of expression data for top BN+1 and core variables. Parentheses indicate specific probe identities.

4.6.3.3 GO enrichment of predicted BN+1 genes

Our previous studies indicate that the top few hundred BN+1 genes (i.e. Those genes predicted by the BN+1 algorithm) often interact with the seed gene network and biologically active relevant to the pathway of interest [56,57]. A GO gene enrichment analysis was performed using 250 top BN+1 genes (Table 4.3). Given the nature of the NF-κB selected core network and their roles in nuclear localization and transcriptional initiation, it was not surprising that many of the recovered genes show some nuclear compartmentalization. Interestingly, many apoptotic and death-related genes were enriched (Table 4.3).

Table 4.3 GO enrichment results for top 100 predicted variables in BN+1 analysis. Entrez gene identifiers were input for the top 250 BN+1 results into the DAVID tool for GO analysis. The 250 results mapped to 188 unique *Mus musculus* and seven unknown species genes, revealing that some of the top genes were represented by multiple Agilent probes in the top results. Benjamini-derived p-values of 0.01 were used as cutoffs.

Term	***Count***	***P-Value***	***Benjamini P-value***
Biological Process			
Cellular process (GO:0009987)	106	8.29E-06	0.00981
Lymphocyte apoptosis (GO:0070227)	4	1.44E-04	0.0823
Cell death (GO:0008219)	15	1.73E-04	0.0663
Death (GO:0016265)	15	2.20E-04	0.0634
Post-embryonic organ development (GO:0048569)	4	2.36E-04	0.0546
Apoptosis (GO:0006915)	14	2.65E-04	0.0512
Programmed cell death (GO:0012501)	14	3.12E-04	0.0517
Cellular Compartment			
Intracellular (GO:0005622)	125	2.93E-08	5.68E-06
Intracellular part (GO:0044424)	119	4.32E-07	4.19E-05
Intracellular organelle (GO:0043229)	105	2.76E-06	1.78E-04
Organelle (GO:0043226)	105	2.84E-06	1.38E-04
Intracellular membrane-bounded organelle (GO:0043231)	93	4.08E-05	0.00158
Membrane-bounded organelle (GO:0043227)	93	4.24E-05	0.00137
Nucleus (GO:0005634)	58	0.001749	0.0474

4.6.4 Discussion

In this section, we first demonstrate the BN+1 algorithm's applicability to studying the BCR pathway, a eukaryotic signalling pathway. Our study shows that BN+1 can also be used to predict pathway elements and gene interactions in important eukaryotic pathways. Therefore, the BN+1 algorithm appears to be a generic BN expansion system that can be used to study other prokaryotic and eukaryotic pathways.

The BN+1 algorithm identified several known and previously undiscovered candidates relevant to NF-kB. A variety of top-scoring BN+1 genes contributed to the overall enrichment of apoptotic and death-related processes. This was not surprising, given that the experimental conditions used in the AfCS microarray dataset included drug perturbations which induce such processes. These data suggest that the gene enrichment approach for assessing biological significance of multiple BN+1 candidates is possible in both prokaryotes and eukaryotes.

Furthermore, recovery of the Jmjd1c and Cxcl13 genes gave additional support to the biological validity of top-ranked BN+1 genes. These genes are already implicated in the context of NF-κB and BCR pathway activities via other experimental studies. Several other candidates were implicated in the BCR and/or NF-κB transcriptional regulatory activities which are prime candidates for additional experimental investigation.

One unique finding from this study was the ability to generate novel predictions using multiple and separate probes as variables in our networks. This type of approach, though naïve for some existing microarray platforms, may be amenable in future next-generation dataset analysis. Furthermore, such assumptions may be useful in studying the behaviors of selected exons and transcripts in various biological contexts in either BN or BN+1 analysis.

The analysis did not include all members of the BCR pathway as represented by KEGG. It is likely that expansion of another subset of genes from this pathway (e.g. Syk, Lyn, Blnk, Btk) that were not recovered as top BN+1 genes here will identify a different set of BN+1 genes. Such a hypothesis could be tested by setting the Syk and other genes as the core network and rerunning BN+1. This was not tested, but could be done easily.

4.7 Discussion and Summary of BN+1 studies

The BN+1 algorithm was demonstrated in the preceding examples to be generally applicable to a wide variety of biological systems in prokaryotes and eukaryotes. We successfully identified novel genetic mechanisms relevant to biofilm formation and regulation which were later verified experimentally by our collaborators. This was achieved by combining the expansion algorithm with a naïve natural language processing approach called term enrichment. Many exciting predictions from all of the BN+1 analyses have not yet been evaluated experimentally and await further validation, though will likely have major impacts on our understanding of pathway entities and their interactions with neighboring/interacting biological entities.

Another method of assessing the significance of predicted results in the biological studies was through GO enrichment of the most well-supported and highest-ranked BN+1 genes. This approach was introduced in the prokaryotic ROS pathway and later applied to the murine BCR pathway, with similar ability to recover pathway-relevant molecular functions and biological processes in the top-scoring BN+1 genes.

Aside from extending across evolutionary scales from prokaryotic to eukaryotic pathways, the methods were also robust to the use of different analysis platforms for gene expression data (Affymetrix versus Agilent platforms). Though this was not tested using platforms for different biological scales (e.g. protein expression, phosphorylation states, etc.), we can expect that BN+1 analyses can identify novel interactions with high significance using those other datasets. It will be interesting to compare how the predictions generated using different dataset types (e.g. mRNA expression vs protein phosphorylation abundance data) compare in terms of their rankings.

Many future directions are envisioned. For example, we can extend the BN+1 algorithm to BN+2, BN+3, or BN+n algorithm by iteratively adding more than one variable to the seed gene network. The principle used in the development of the BN+1 algorithm can also be used for dynamic BN analysis. We are currently in the processing of developing a DBN+1 algorithm and using it for dynamic data analysis.

Chapter 5

Combined Bayesian Refinement and Expansion Towards Identification of Novel Molecular Interactors in Progressive Kidney Disease

5.1 Introduction

In this analysis, the developed methods from preceding chapters were applied towards a biomedically-prevalent and relevant disease in humans, progressive kidney disease. Progressive kidney disease is a complication that can occur in some diabetes patients and can include kidney failure and sometimes even death [100].

In order to study the progressive kidney disease, data were analyzed from two different compartments within the kidney nephrons, the glomeruli and tubules. These compartments specifically relate to the activities of the nephron, and have been shown previously to have differential pathological changes at different stages of the disease [101,102,103]. The Jak/Stat signaling pathway which has been implicated in the progressive kidney disease was selected as a starting point for our Bayesian network refinement and expansion algorithms [102,103]. We attempted to see whether the pathway has different regulatory roles or responses in the two compartments using existing microarray data, and whether novel or known regulators are implicated in either or both of the compartments. By identifying the most likely pathway interactors, we hoped to implicate new genes in the different stages of the progressive kidney disease for future validation and eventual therapeutic development.

Prior to the human progressive disease analysis, we tested whether a set of refined networks from the EdgeClipper approach could later be expanded to identify novel hidden factors in a simpler model. Given that the ROS detoxification pathway had already been studied in detail in both the EdgeClipper (Chapter 3) and BN+1 (Chapter 4), the use of this pathway for testing the combinatorial application of EC and BN+1 for refinement and expansion, respectively, was a logical and simple extension. The combined approach was tested using both moderate and strict consensus networks from the EdgeClipper analysis of the ROS pathway (Chapter 3). The results from the expansion of both refined networks are described.

5.1.1 Methods

5.1.2 Selecting core networks from different consensus levels via EdgeClipper

Two levels of consensus were selected from the ROS pathway using the EdgeClipper algorithm. A medium consensus network of intermediate stringency (B-value=10^{-3}) was derived from the top 3,644 simulated networks. The medium network contained 13 genes. When all the top 20,000 networks saved were included in our simulation, the B-value equaled zero. Under this condition, the consensus network was similar to the medium network except that two edges were absent, *gadX-sodC* and *oxyR-crp*. Three separate sub-networks remained. These included *gadE-gadX-gadW*, *fis-sodC-katE,* and *marA-marR*. In this thesis, the highly conserved network that connects the three genes *gadE*, *gadW*, and *gadX* was defined as the small network. These three genes are members of a known "acid fitness island" and are important regulators of *E. coli* resistance to extreme oxidative acid stress [104]. The small network was chosen as the network starting point for our subsequent small network expansion study.

5.1.3 BN+1 settings for medium and small core networks

In each iteration of the BN+1 simulation, the current BN+1 gene was added to the original data file. This was followed by a simulated annealing search 5x10^6 networks for the medium network expansion and 1x10^6 networks for the small network expansion. Although the consensus network was selected as a starting point or seed, during the learning round all edges could be modified such that the addition of genes could change the backbone structure of the resulting model (i.e., unfixed structural prior). Genes were sorted based on their log posterior scores.

5.1.4 BN+1 Neighborhood Analysis.

To display the genes in the core network that were strongly connected in the BN+1 analysis, a heat map-based visualization method we termed consensus neighborhood analysis was introduced to characterize patterns of connectivity between the core BN genes and selected BN+1 genes. Consensus neighborhoods represent conserved connections between core genes and BN+1 genes across a set of replicate BN+1 runs.

For each BN+1 gene, the set of top networks with identical best score predicted for that gene were used to define the consensus network. Edges were shown as directed arrows if the relationship appears in 100% of the selected top networks with specified directionality; otherwise, a relationship was defined as undirected if the cumulative frequency of parent and child relationships between the selected core and BN+1 gene equaled 100%. The relations of the core genes with respect to the BN+1 gene are defined as: 1) core gene is a parent node with an edge directed towards the predicted gene; 2) core gene is a child nodes with an edge directed inwards from predicted gene; and 3) core gene shares an undirected edge with the predicted gene. The top fifty rank-ordered genes from the 13-gene BN+1 analysis were selected for inclusion in the visualization. Hierarchical clustering was generated using a binary distance metric and the Heatplus module in R. Biological terms were manually curated using information from Entrez Gene and literature.

5.1.5 Results

All three consensus networks (including the 27-variable network identified in Chapter 2) with different levels of stringency were individually compared to the known ROS pathway. As stated earlier in this thesis, 29% of the edges in the consensus are supported by corresponding edges in EcoCyc [23] or RegulonDB [81]. However, inclusion of literature information in the comparison revealed that approximately 42% of the edges found in the large consensus network were confirmed. The medium consensus network is more consistent with EcoCyc, RegulonDB, and information contained in the literature with 78% of the edges supported. The two missing edges were also supported when weak evidence was included. A detailed analysis revealed that the interactions involving direct transcription factor binding activities (e.g. *marA-marR, ihfA-ihfB, gadE-gadX-gadW*) [23,105,106,107] were among the most highly-conserved edges in the medium consensus network. The small consensus network (*gadE-gadX-gadW*) is a sub-network of the large and medium consensus networks and is 100% consistent with

EcoCyc, RegulonDB, and information found in the literature. Specifically, it has been reported that *gadX* regulates *gadE* and *gadW*, and that *gadE* and *gadW* regulate *gadX* [23,65,104,107,108].

A novel representation termed consensus neighborhood analysis was developed to test whether specific core genes have a preferential impact on the selection of BN+1 genes. The consensus neighborhoods are derived from consensus BN analysis and applied to networks from replicate BN+1 simulations, and are comprised of those core network genes that strongly influence or are influenced by a specific BN+1 gene. Consensus networks for the fifty top predicted BN+1 genes and their connected core genes from the BN+1 expansion of the medium network were generated (Fig. 5.1). This analysis confirmed the capability of the BN+1 to predict the involvement of new genes and gene interactions related to ROS and other stress responses. Core genes on the left side of Fig. 5.1 (e.g. nearer to *gadE*) have more connections with the predicted BN+1 genes than those on the right side (e.g. nearer to *marR*). This result indicates that certain genes play more important roles than others in prediction of new pathway genes. It also suggests that removal of some genes from a selected core topology during the *B*-value selection may have more important effect on subsequent '+1' recovery, because removal of those core genes with closest correlations to the '+1' genes will limit recovery of the '+1' gene in the "top" hits. This analysis also identified seven "acid fitness island" genes that are clustered together (boxed genes *gadA*, *hdeB*, *hdeD, yhiD*, *gadB*, *hdeA*, and *slp*) and are all connected to *gadE,* suggesting the important role of *gadE* in the selected experimental conditions and overall effects on the selected pathways.

Expansion of the small consensus network resulted in the selection of genes predominantly from a known acid fitness island [104]. The acid fitness island is a coordinately regulated gene cassette (shown in Fig. 5.2). Interestingly, nine of ten acid fitness genes not already included in the small network were recovered within the top 10 BN+1 results (Table 5.1 and Fig. 5.1). The tenth acid fitness gene, *mdtF*, was the 80th top predicted gene from the small network expansion. The acid fitness genes have multiple functions, including decarboxylation of glutamic acid to remove intracellular protons (*gadA*), protection from organic metabolite products produced during fermentation (*yhiF*, *slp, hdeA*), recovery from protein damage induced by the diffusion of organic acids into cells (*hdeA*), direct processing of organic acids (*yhiF*, *slp*), and predicted membrane activity (*yhiD, hdeD*) [104]. The BN+1 search also identified the glutaminase *ybaS*, a gene that has been suggested to participate in acid resistance activity in *E. coli* [109]. *YbaS* is outside of the physical acid fitness island [104]. These results indicate that the BN+1 algorithm is able to accurately predict acid stress regulatory genes within and outside the known acid fitness island.

Table 5.1 Top 10 genes with best log posterior scores predicted from BN+1 expansion based on the large, medium, and small consensus networks. Numbers shown after gene names are negative logs of posterior probabilities for each top network containing the respective predicted gene. Highlighted cells represent known acid fitness genes.

Rank	Large Network (27 gene)	Medium Network (13 gene)	Small Network (3 gene)

1	**dusB** (tRNA-dihydrouridine synthase B); S=-8295.81	**dusB** (tRNA-dihydrouridine synthase B); S=-3821.20	**slp** (outer membrane lipoprotein);
			S=-949.65

2	**fdhE** (formate dehydrogenase formation protein); S=-8298.44	**sra** (30S ribosomal subunit protein S22); S=-3850.29	**hdeA** (stress response protein acid-resistance protein); S=-954.57

3	**uspE** (stress-induced protein);
	S=-8310.63
		yodD (predicted protein); S=-3850.30	**hdeB** (acid-resistance protein)
			S=-958.11

4	**yohF** (predicted oxidoreductase with NAD(P)-binding Rossman-fold domain); S=-8312.24	**fbaB** (fructose-bisphosphate aldolase class I); S=-3860.69	**gadA** (glutamate decarboxylase A, PLP-dependent); S=-968.53

5	**yncG** (predicted enzyme);
	S=-8313.04
		slp (outer membrane lipoprotein);
		S=-3865.13
			gadB (glutamate decarboxylase B, PLP-dependent); S=-972.15

6	**msyB** (predicted protein);
	S= -8318.20
		hdeA (stress response protein acid-resistance protein);
		S=-3870.05
			hdeD (acid-resistance membrane protein); S=-973.65

7	**yedP** (conserved protein);
	S=-8320.30
		msyB (predicted protein);
		S=-3871.68
			yhiD (predicted Mg(2+) transport ATPase inner membrane protein);
			S=-975.68

8	**sra** (30S ribosomal subunit protein S22); S=-8323.97	**hdeB** (acid-resistance protein);
		S=-3873.59
			dctR (predicted DNA-binding transcriptional regulator);
			S=-993.91

9	**ydcK** (predicted enzyme);
	S=-8325.91
		erfK (conserved protein with NAD(P)-binding Rossmann-fold domain);
		S=-3877.97
			ybaS (predicted glutaminase);
			S=-996.20

10	**ynhG** (conserved protein);
	S=-8326.20
		ynhG (conserved protein); S=-3878.40	**mdtE** (multidrug resistance efflux transporter); S=-1017.59

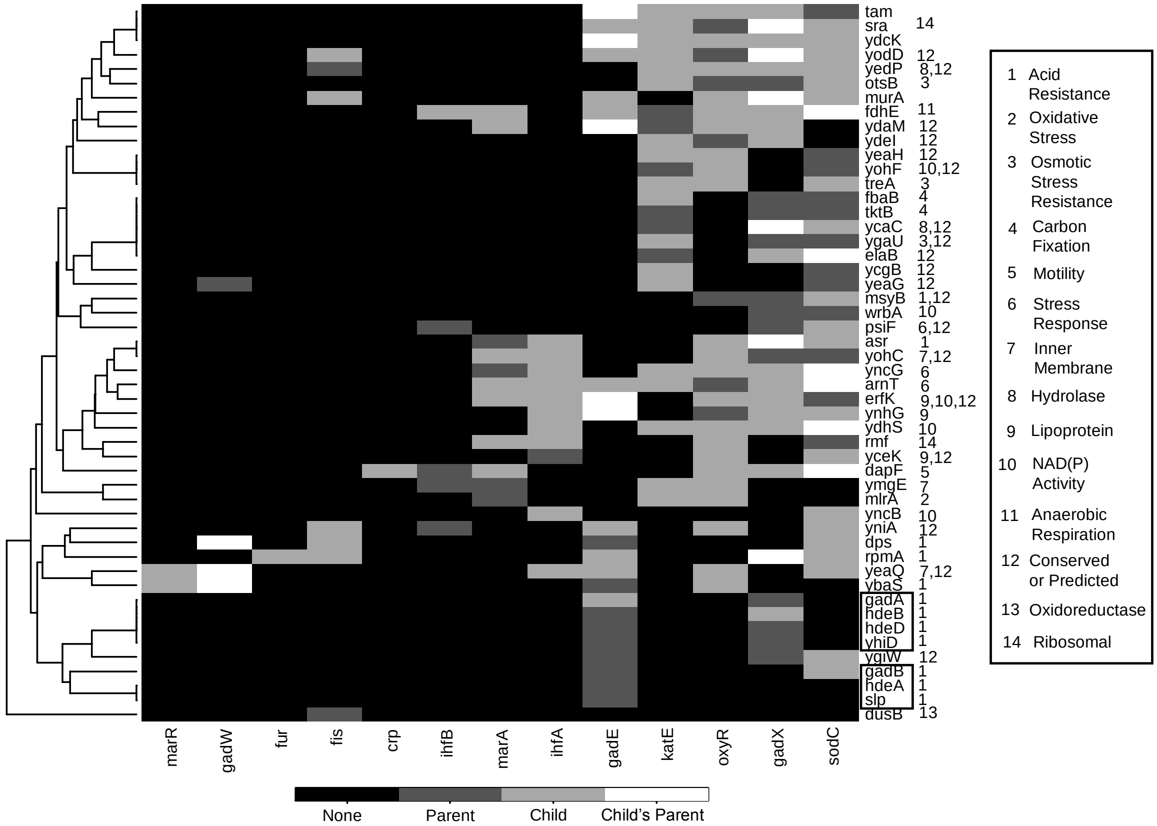


Figure 5.1 Consensus neighborhoods and functions of BN+1 expansion genes. Matrix representation was generated for the top fifty BN+1 genes predicted to interact with medium network by BN+1 analysis. Each cell in the heatmap represents a relationship between a BN core gene (x-axis) and a particular BN+1 gene (y-axis) with selected grayscale shading that represents predicted relationships of core genes respective to the predicted genes. Biological functions and localization (obtained from Entrez Gene and PubMed) curated manually are indicated in margin of vertical axis. The boxed gene names show genes from the acid fitness island.

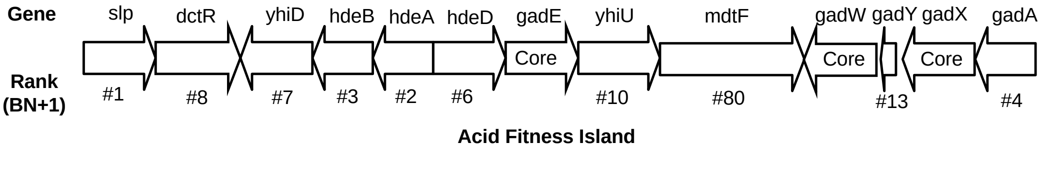


Figure 5.2 Acid Fitness Island Genes Identified By Combining EC and BN+1. Genomic localization of acid fitness island genes presented in graphical form (genes not shown to scale), with order of the genes in BN+1 results listed as numbers (#). Genes appearing in the core network included gadE, gadX, and gadW.

These results suggest that BN+1 can be executed following an initial refinement of some existing network topology using the EdgeClipper algorithm. In all three models, the results recapture many important biological functions and mechanisms related to those of the included set of genes in the respective model. The ability to further refine the functionality down to operon levels (arguably a genomic and not proteomic feature) is itself a novel and interesting finding. However, in terms of general oxidative stress pathway activities, the expansion of only the gad genes gave a more restrictive set of biological functions that do not encompass the majority of molecular function and biological process GO terms relevant to ROS. As listed in Figure 5.1, the biological mechanisms relevant to oxidative stress include a variety of mechanisms, including cold and osmotic shock, anaerobic respiration and NAD(P), and mitochondrial/inner membrane activities. More investigation will be needed to infer whether these proteins have other oxidative stress pathway activities and roles aside from their known annotated functions. However, the validity and power of combining our EC and BN+1 approaches was established with this preliminary study.

5.2 Eukaryotic study: diabetic nephropathy and progressive kidney disease

Diabetic nephropathy (DN) is an increasingly more prevalent and devastating disease worldwide. Diabetic nephropathy develops in approximately 30% of patients with type 1 or type 2 diabetes [32]. This is significant, since the number of human patients worldwide with diabetes is expected to reach 380 million by the year 2025 [32]. Diabetic nephropathy is the most common cause of end-stage renal disease, which itself is an important predictor of cardiovascular risk and mortality. The clinical progression of diabetes nephritis is classified into 5 phases: hyperfiltration and renal hypertrophy, glomerular filtration and increased renal plasma flow, changes to the renal parenchymal basement membrane thickness as well as mesangial expansion, microalbuminuria and early hypertension, proteinuria formation, and end-stage renal disease [32]. Interestingly, there appears to be distinct regulatory processes occurring in the microstructures of the nephrons during progressive kidney disease and DN. Most specifically, some evidence suggests differential regulatory processes in the glomerular and tubulointerstial compartments of the kidney at the genetic level.

It has been proposed that the Jak/Stat signaling pathway may play a role in the events of progressive kidney disease [102,103]. The Jak/Stat signaling pathway directly targets the expression of mammalian genes following response to cytokine and growth hormone receptor signaling [110]. During Jak/Stat signaling, signaling by the effector cytokines or growth hormones will activate Jak, phosphorylate its receptor, recruit and phosphorylate STAT molecules which will dimerize and translate into the nucleus, and activate gene expression [110]. Jak/Stat signaling is known to be involved in various renal diseases, though the mechanisms by which the pathway interact in the various renal disease progressions is complicated and can vary across various species (e.g. human vs murine) [110] and even different tissue types. For example, the Stat3 protein shows differential regulation in multiple cell types (mesangial, podocyte, interstitial fibroblasts, tubular epithelial cells, macrophages, and lymphocytes), though it is unclear what the role of Stat3 might be in renal fibrosis and other processes [110]. It is also unclear as to which Stat gene is the major regulator of Jak2 signaling, which is also proposed to be important in humans [110].

In this analysis, we implement a Bayesian network analysis to determine whether the Jak/Stat signaling pathway plays a role in the differential regulation of two kidney compartments during progressive kidney disease. Kidney biopsy data from human patient for glomerular and tubulointerstitial compartments were used in learning distinct Bayesian networks for the Jak/Stat signaling pathway. We hypothesized that if the mechanisms in the two compartments of kidney are distinct or different, and if the Jak/Stat signaling networks are also distinct for those compartments, then sufficiently different network models and interactions should be observed for the two models. Second, the expansion of the most well-supported interactions for each compartment should identify genes outside the Jak/Stat pathway which relate specifically to the disease processes inherent in each kidney compartment and should be distinct for those compartments. Bayesian networks were generated, refined using our developed EdgeClipper algorithm and compared for the two compartments, and finally each expanded using the developed and tested BN+1 algorithm. These methods were applied to better understand the role(s) of the Jak/Stat signaling pathway in two kidney compartments and hopefully identify new gene regulators of the pathway in the kidney compartments for eventual therapeutic development.

5.3 Methods

5.3.1 Microarray Analysis for Diabetes Study

De-identified microarray data were obtained from the Kretzler laboratory (courtesy of Felix Eichinger and Matthias Kretzler). In summary, the data were obtained from human (including Pima Indian) kidney biopsies in either glomerular or tubule compartments.

Data were obtained for the biopsies using an Affymetrix HGU133A microarray platform (12,025 features). The microarrays were processed using individual normalization followed by ComBat normalization [111] by F. Eichinger prior to receipt of the data by A.P. Hodges. During BN analysis, each gene was discretized using a q3 maximum entropy approach (which was also applied in previous chapters).

Four distinct datasets were created using the available microarray data. The full sets of glomerular and tubular data were used as two distinct data sets. A subset of the full glomerular data set (74 out of 298 chips) was selected to specifically include only those microarray data from patients with either progressive kidney disease and diabetes mellitus (DM) indications or no disease as a third dataset (partial glomerular dataset). Identical rules were applied to the selection of a corresponding partial dataset (71 out of 278 chips) from the full tubule set. Thus, four data sets were generated which were used in the subsequent Bayesian analyses, two of which are more specific to the DM-based progressive kidney disease comparison.

5.3.2 Pathway Selection and Bayesian network construction

A set of 131 genes were identified for the Jak/Stat pathway within the Kyoto Encyclopedia of Genes and Genomes (KEGG) database. This set of genes represents the set of all genes participating in the Jak/Stat pathway as either genes, proteins, or another biological entity within the curated KEGG pathway. Four distinct Bayesian network models were constructed using the four datasets generated above.

In the preliminary simulation studies, the following identical rules were applied. For each of the two smaller datasets (partial glomerular, partial tubule), Bayesian networks were generated using the following rules. Runtimes were increased to a maximum of 50 minutes or 1 x 10^8 maximum searched networks for each of 200 independent simulations for the full tubule and glomerular studies to allow a more thorough search. Standard searcher parameters from our previous studies were employed in these four major analyses, including maximum entropy (q3) quantization, maximum cap of three parent variables per any given variable, and simulated annealing search.

5.3.3 BN Refinement and core network generation using EdgeClipper (EC-R)

The set of top-scoring networks saved in each simulation were then refined using the restrictive EdgeClipper (EC-R) algorithm. Network refinement was implemented to reduce the number of edges in saved networks and allow comparison of the most well-supported interactions in the glomerular and tubule compartments. In short, the restrictive EdgeClipper (EC-R) algorithm was implemented when modeling each of the four network sets. Log posterior probabilities from the Bayesian network results were used to generate a distribution of B-values corresponding to each unique network score. Edges were filtered using B-value cutoffs of roughly 0.10 or stricter (e.g. 0.01 and smaller) for each of the four datasets' Bayesian networks. Different B-values were required for each dataset to achieve a set of variables with count with fewer than 100 variables, and were distinct for each dataset due to distinct B-value distributions for each dataset. A B-value of 1 x 10^{-3} was initially sought to select the genes. Additional selection was implemented to include at least 50 genes. Genes were included based on the B-values beyond which the genes would not be connected to any other genes in the consensus networks. Those interactions amongst variables which appeared within 100%

of the saved networks meeting the B-value criteria were included in the final consensus network as either directed or undirected edges.

5.3.4 Core Network Expansion Using BN+1

Core networks were selected for the two smaller datasets (glomerular and tubule) using a B-value cutoff criterion. The BN+1 expansion algorithm was implemented for each of the core networks and respective datasets to identify novel hidden factors. Each of the genes not included in the pathway were tested for their ability to improve the respective core network when added to the model. A total of 2 x 10^8 networks were searched for each individual gene not already included in the core network (~13,000 genes). After running the simulations, the genes were sorted and ranked according to best log posterior score. The top BN+1 genes identified for the partial dataset experiments were later compared and used to assess the respective neighborhoods around the consensus networks and core genes.

5.3.5 Network Overlay and Comparison

Core networks with the same underlying number of data observations (partial glomerular vs partial tubule, etc.) were overlain and compared to identify which interactions were shared for the two biological compartments. Interactions represented as either directed or undirected edges in the network were compared for the two selected consensus networks. If the interaction appeared in both networks regardless of direction (e.g. directed in the first, undirected in the second), that interactions was counted as present in both networks. Directionality was not included as a criterium for assigning presence/absence of an edge.

Frequency of edge occurrence was considered within the rules of the applied EC-R approach.

5.4 Results

5.4.1 Few gene interactions (*e.g.,* edges) are conserved in either compartment as revealed by EdgeClipper EC-R refinement analysis

Results from EdgeClipper (EC-R) refinement of BNs trained using the four datasets are shown in Figures 5.3-5.6. The EdgeClipper algorithm was applied to refine our results due to several observed issues. First, the Bayesian networks for each of the four datasets contained high numbers of disconnected nodes. This observation suggested that many nodes did not have sufficiently supported interactions with other genes in the list. It is possible that more links would be shown up given more execution time. However, the time performed in this study was considered sufficient to get basic interactions, and those disconnected nodes were likely to have no or very weak connection to any of the genes listed. Second, the number of data points in each analysis was relatively low when compared to the size of the network models (131 variables). In the earlier chapters' ROS analysis, 305 data points were available and used when modeling interactions amongst 27 variables. This issue of sufficient data, combined with the earlier results of 29-42% recovery of known interactions and arguably high false positive rates, suggests that an even higher incidence of false positive rates in the data can be observed. Fourth, it is desirable to identify the most well-supported interactions observed in the two kidney compartments during the progressive kidney disease. This selection allows a more stringent comparison of the two compartments and their best-supported interactions.

Interestingly, the refinement of the networks trained on partial datasets using the EdgeClipper algorithm showed a significant drop-off in number of connections as B-values were successively decreased towards zero. These results suggest that the supported interactions in each BN analysis are only moderately supported given the small amount of data available in each dataset. Hence, it is likely that the addition of more patient data could improve the robustness of these results and provide more confidence in predicted interactions.

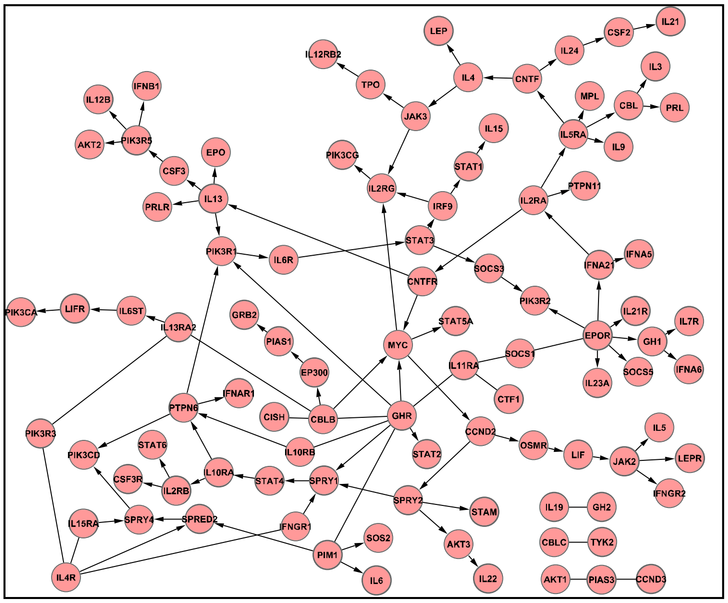


Figure 5.3 Consensus network of Bayesian networks using partial dataset selected from glomerular data. A subset of the full dataset (74 out of 298 chips) was chosen for this analysis using a regular expression (regex) match. Thirty-three nodes were disconnected from the network.

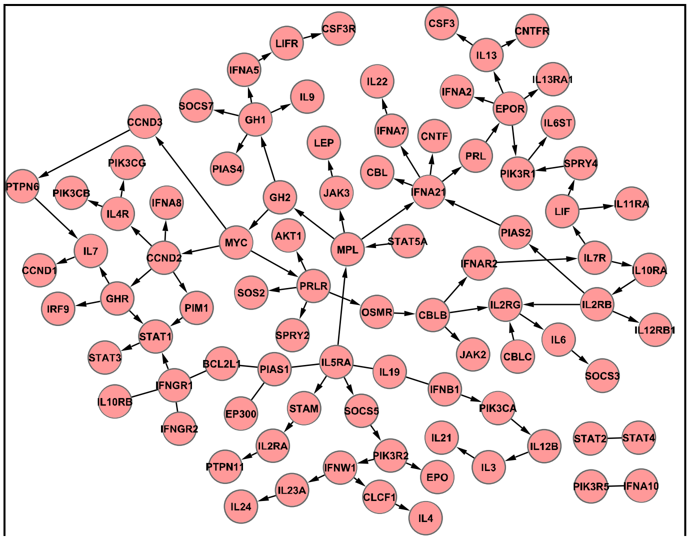


Figure 5.4 Consensus network of Bayesian networks using partial dataset selected from tubulointerstitial data. A subset of the full dataset (71 out of 278 chips) was chosen for this analysis using a regular expression (regex) match. 42 nodes were disconnected from the network.

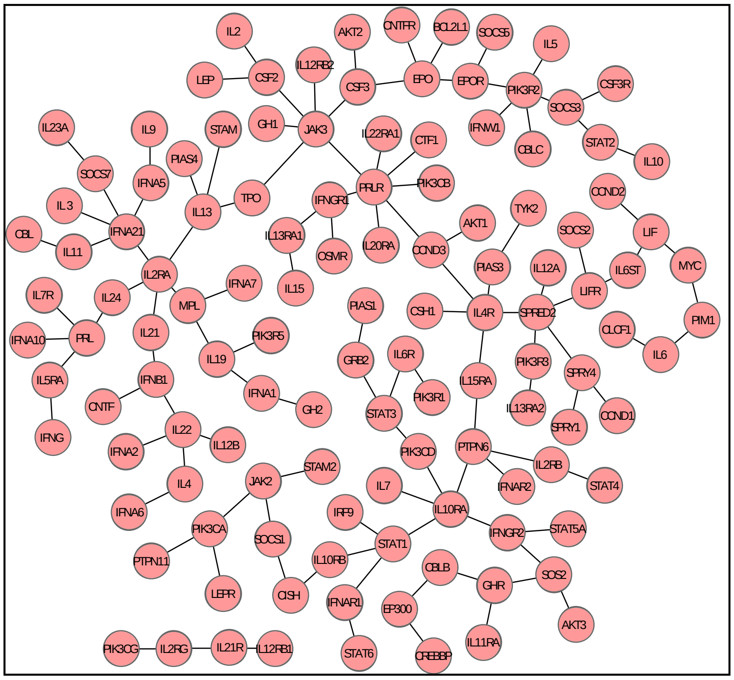


Figure 5.3 Consensus network for Bayesian networks generated using the full glomerular dataset (298 chips**).** 8 nodes were disconnected from the network.

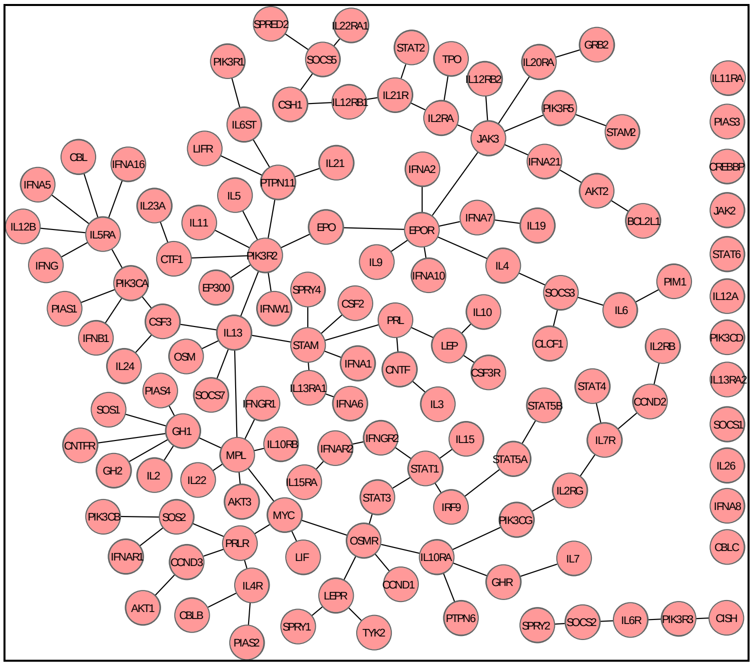


Figure 5.4 Consensus network for Bayesian networks generated using full tubulointerstital dataset (278 chips**).** 12 nodes were disconnected from the network.

Furthermore, the number of disconnected nodes observed in each network analysis was relatively high for networks trained on smaller datasets. A totally disconnected network is often the starting network for scoring comparison. We expect that as more networks are searched, there is an increased expectation of any given node being connected with one or more additional variables in the network due to the number of searched networks increasing. Comparison of the networks in Figures 5.3-5.6 revealed that as more data are added to the simulation, the number of disconnected nodes decreases (disconnected nodes not shown). It is possible that some nodes which were previously connected to other variables can be disconnected following addition of more data (e.g. "Spry2" in Figs. 5.4 and 5.6), though this is less common than the addition of edges to previously-disconnected nodes.

5.4.2 Glomular and tubule compartments show disparately low overlap, suggesting differential pathway activities

The consensus networks generated by EdgeClipper for tubule and glomerular compartments with the same data size were compared in Table 5.2. Overlay of the full tubule and glomerular dataset models revealed minor overlap with only seven shared connections in the follow-up simulation study using the full datasets. Chi-square and Fisher exact tests revealed that the number of interactions shared between the two studies is indistinguishable from random chance, suggesting no significant overlap between the two sets of interactions. These data suggest that the two compartments have different conserved interaction sets with selected biological functions for Jak/Stat pathway genes, suggesting differential regulation of the Jak/Stat pathway elements in the two compartments.

Furthermore, no overlap was observed for the shared interactions of full dataset glomerular and tubule models and those of the partial dataset models. This suggests that the inclusion or exclusion of roughly 150 data points has a major effect on the most well-supported interactions. One possibility is that patients with non-DM kidney disease and/or other conditions may have alternative mechanisms and biological events occurring. However, this statement is weakened by the possibility that all of the simulations have not converged to optimal solutions (e.g. insufficient simulation time). More investigation is needed to elucidate this difference in shared interactions and varying datasets.

Table 5.2 Summary of shared interactions in the partial and full dataset analyses.

	#GA interactions	#TA interactions	#overlapping Interactions	%GA, %TA	Shared interactions
Partial	107	92	7	6.5%, 7.6%	EP300-PIAS1, CNTFR-IL13, CSF3-IL13, IL2RB-IL10RA, IL5RA-MPL, IL2RA-PTPN11, CCND2-MYC
Full	121	117	14	11.6%, 12.0%	IL10RA-PTPN6, STAT1-IRF9, IL12RB1-IL21R, EPO-EPOR, AKT1-CCND3, IFNG-IL5RA, IFNW1-PIK3R2, IL2RG-PIK3CG, IL6-PIM1, IL12RB2-JAK3, LIF-MYC, CCND3-PRLR, IL13-STAM, IL5-PIK3R2

5.4.3 New Jak/Stat pathway elements were discovered through BN+1 expansion

The top BN+1 results for the expansion of core network models with partial glomerular and tubule data are show in Table 5.3. Genes from the BN+1 analysis were ranked according to the best achieved BN score (scores closest to zero). The top genes for each compartment appear to be biologically relevant to the compartment’s known disease processes. For example, several genes with known roles in oxidative stress and redox, mitochondrial activities, or apoptosis were identified for the partial tubule dataset model, including TMSB10, DNAJC16, PRDX4, MAPK10, and ACADL. Interestingly, the top results from the glomerular compartment expansion included genes with known roles in cell growth & differentiation, signal transduction, cytoskeleton remodeling, or membrane transport, such as PLCE1, CCDC91, HPS5, SRGAP2, PHACTR4, ARHGAP19, and IQGAP2.

Genes are ranked according to their maximum BN score generated during the BN+1 search procedure. Other genes which connect to the BN+1 gene in the top networks are listed as "Neighbors" or neighbor genes. These neighbors constitute a portion of the Markov Blanket for the BN+1 gene (note: parents of child nodes are not included as neighbors due to their lack of direct connection to the BN+1 gene).

Table 5.3 BN+1 expansion of glomerular and tubule compartments models identifies distinct novel regulators for the Jak/Stat pathway which are distinct to the respective compartmental disease mechanisms.

BN+1 results for partial tubule dataset				
Rank	**Gene**	**Symbol**	**BN Score**	**Neighbors**
1	9168	TMSB10	-4688.4523	CCND2, PRLR, MYC
2	23341	DNAJC16	-4689.8864	IL4R, GHR, MYC, PIAS2, CCND3
3	10549	PRDX4	-4692.432	MPL, MYC, SOCS7, IL13, JAK3, CNTFR
4	11025	LILRB3	-4692.5516	GH1, IL3, LIFR, JAK3, MPL, IL2RA, IFNW1
5	80339	PNPLA3	-4692.9465	IL13, CNTFR, PRLR, IFNGR1, BCL2L1
6	11177	BAZ1A	-4693.2419	MYC, CBLB, CCND2, IFNGR1, IL13, STAT1
7	5602	MAPK10	-4693.2527	IFNGR1, PRLR, MPL, BCL2L1, IL13, CBLB
8	1629	DBT	-4694.5034	CCND2, SOS2, IL2RB, BCL2L1, IL2RG, IL4R
9	33	ACADL	-4694.9172	GHR, IL4R, IL2RG

10	51765	MST4	-4694.9276	MYC, CCND3
BN+1 results for partial glomerular dataset				
Rank	**Gene**	**Symbol**	**BN Score**	**Neighbors**
1	51196	PLCE1	-4942.4257	GHR, IL11RA, SOS2, CBLB, EPOR
2	64398	MPP5	-4942.9408	GHR, IFNGR2, SOS2, IL11RA
3	55297	CCDC91	-4945.3412	PIK3R1,SOS2, IFNGR2, IL11RA, GHR, IL6
4	54463	FAM134B	-4947.9157	GHR, MYC, EP300, SOS2, CBLB, IL11RA, IL6, IFNGR2
5	9863	MAGI2	-4948.964	GHR, EP300, IL6, IFNGR2, EPOR, CBLB
6	11234	HPS5	-4949.0538	GHR, IL11RA, PIK3CA, EPOR, IL6, SOS2, SOCS7, IFNGR2
7	23380	SRGAP2	-4949.101	GHR, IFNGR2, CBLB, IL11RA
8	65979	PHACTR4	-4949.6565	GHR, SOS2, IFNGR2, CBLB, EP300, IL6, IL11RA
9	84986	ARHGAP19	-4951.9786	GHR, IL11RA, EPOR, CBLB
10	10788	IQGAP2	-4952.731	GHR, IL6, IL11RA, SOS2, IFNGR2, CBLB

The top genes identified in the tubulointerstitial and glomerular compartments were TMSB10 and PLCE1, respectively. Comparison of the relationships between these top BN+1 genes and their interactors in the core networks (listed in Table 5.3) revealed compartment-specific relationships when plotted in a pairwise manner (Figures 4.6 and 4.7). Given the normalization method used in the two separate datasets, the max and min values for both compartments for any particular gene are not directly comparable. However, more obvious relationships are demonstrated in the tubule versus in the glomerular data for TMSB10 and its core network interactors (and, similarly, for PLCE1 in the glomerular over the tubule data).

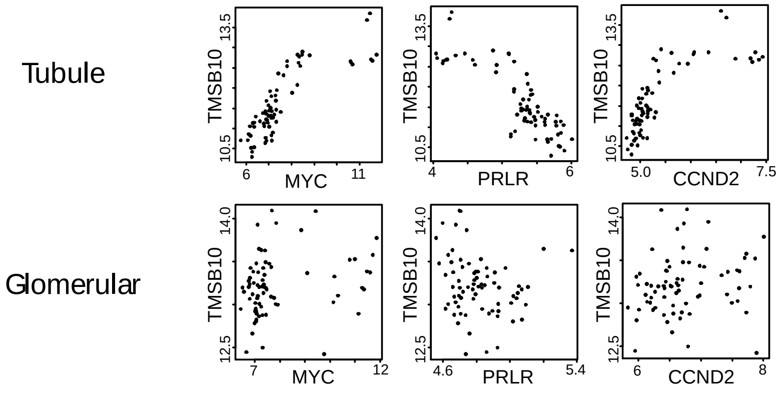


Figure 5.5 Scatterplots of TMSB10 with connected core genes and respective datasets in tubule and glomerular compartments.

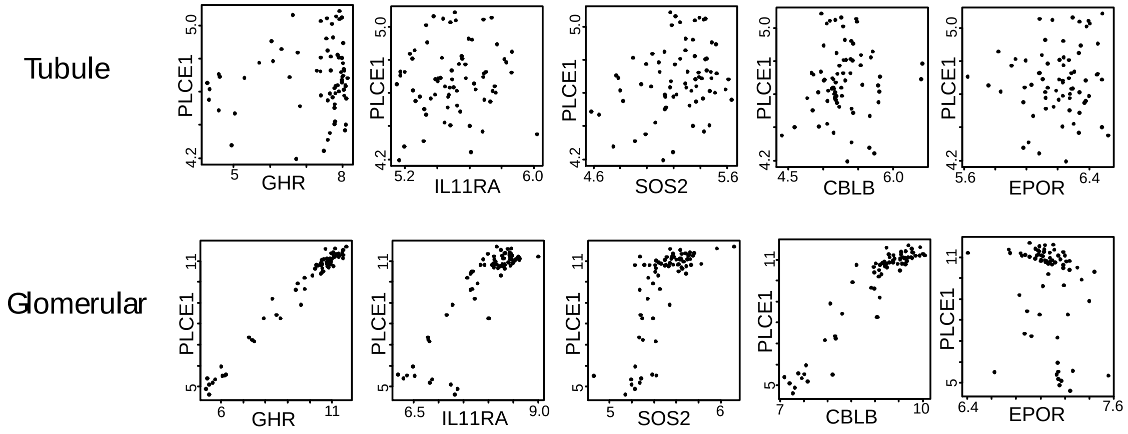


Figure 5.6 Scatterplots of PLCE1 with connected core genes and respective datasets in tubule and glomerular compartments.

Thymosine beta 10 (TMSB10) has no known biological function, though has high sequence similarity between humans, rats, and other mammals [112]. The gene has been isolated from human kidney using cDNA cloning and has been used to show differential expression in the kidney [112]. In murine studies, TMSB10 was shown to be a biomarker in the murine glomerular crescent when perturbed in a chronic graft versus disease modeling [113]. TMSB10 is generally downregulated in human pelvic lymph node metastasis (PLNM) and was proposed to have roles in other cancers [114]. Interestingly, the entire network neighborhood around TMSB10 includes genes with cancer-related associations or functions.

Mutations of phospholipase C epsilon 1 (PLCE1) has already been implicated in diffuse mesangial sclerosis and early onset nephrotic syndrome [115]. PLCE1 is a member of a phospholipase family which catalyzes hydrolysis of phosphotides to generate products which regulate cell growth, differentiation, and gene expression [116]. The gene is expressed and enriched for protein abundance in the mature glomerular podocytes [116]. Hinkes *et al.* showed that recessive mutations in PLCE1 were causative for nephritic syndrome variants [116]. The PLCE1 gene was identified by LOD analysis for nephritic syndrome followed by haplotype analysis, and further implicated by cDNA identification of the 34 exons (distributed over 334.4 kb) and seven homozygous PLCE1 mutations (6 truncating, 1 missense). This finding was especially interesting, given that their demographic groups from Central Europe and Turkey are different than the Pima Indian group in our study dataset and represent an independent source of verification for our findings.

Thus, these findings strongly establish the predictive power of the combined EdgeClipper and BN+1 approaches for characterizing the roles of know pathways in different compartments or tissues, as well as further expanding those pathways to include novel interactors. We again expect that members of our BN+1 lists with no known functions are viable candidates for additional functional assessment and analysis in future studies.

5.4.4 GO Enrichment for BN+1 Results Reveals Relevant and Specific BN+1 Gene Functions for the Selected Compartments

The results were further confirmed using GO enrichment. The top 250 variables in each BN+1 analysis were tested using GO enrichment to see which biological functions were most strongly conserved for those genes (Tables 5.4 and 5.5). Similar to what was observed in Chapter 2 for BN+1 expansion of the ROS pathway, the top set of genes for each expanded compartment model relate biologically to the functions and activities of the core gene network. Redox and mitochondrial-related functions were

Table 5.4 GO enrichment terms for top 250 BN+1 genes from glomerular expansion model meeting Bonferroni-corrected p-value < 0.05.

Term	**Count**	**%**	**P-value**	**Bonferroni**
Biological Process				
Vasculature development (GO:0001944)	15	6.024096	2.03E-05	0.033463
Molecular Function				
Protein binding (GO:0005515)	158	63.45382	6.49E-07	3.03E-04
Glycosaminoglycan binding (GO:0005539)	12	4.819277	7.72E-06	0.003589
Pattern binding (GO:0001871)	12	4.819277	1.91E-05	0.008844
Polysaccharide binding (GO:0030247)	12	4.819277	1.91E-05	0.008844
Carbohydrate binding (GO:0030246)	17	6.827309	8.10E-05	0.037052
Cellular Compartmentalization				

Cytoskeleton (GO:0005856)	39	15.66265	5.16E-05	0.01384

Table 5.5 GO enrichment terms for top 250 BN+1 genes from tubule expansion model meeting Bonferroni-corrected p-value < 0.05.

Term	**Count**	**%**	**P-value**	**Bonferroni**
Biological Process				
Carboxylic acid metabolic process (GO:0019752)	27	10.84337	2.72E-07	4.68E-04
Oxoacid metabolic process (GO:0043436)	27	10.84337	2.72E-07	4.68E-04
Organic acid metabolic process (GO:0006082)	27	10.84337	3.12E-07	5.37E-04
Cellular ketone metabolic process (GO:0042180)	27	10.84337	3.96E-07	6.82E-04
Oxidation reduction (GO:0055114)	26	10.44177	1.13E-05	0.019288
Molecular Function				
Coenzyme binding (GO:0050662)	13	5.220884	9.74E-06	0.005693
Oxidoreductase activity (GO:0016491)	25	10.04016	3.94E-05	0.022826
Cofactor binding (GO:0048037)	14	5.62249	5.24E-05	0.030242
Nucleotide binding (GO:0000166)	54	21.68675	7.30E-05	0.04188
Cellular Compartment				
Mitochondrion (GO:0005739)	38	15.26104	5.30E-07	1.56E-04
Cytoplasmic part (GO:0044444)	105	42.16867	7.00E-07	2.07E-04
mitochondrial part (GO:0044429)	26	10.44177	1.19E-06	3.52E-04

Cytoplasm (GO:0005737)	140	56.2249	1.23E-06	3.62E-04
Organelle part (GO:0044422)	91	36.54618	8.97E-06	0.002643
Intracellular organelle part (GO:0044446)	90	36.14458	1.30E-05	0.003821
Mitochondrial matrix (GO:0005759)	14	5.62249	2.14E-05	0.006299
Mitochondrial lumen (GO:0031980)	14	5.62249	2.14E-05	0.006299

enriched in the tubular BN+1 gene set, whereas cytoskeleton development and vascularization were enriched in the glomerular BN+1 gene set. Thus, these data provide further support to the claim that the Jak/Stat pathway shows differential regulation depending upon which compartment is considered (and hence which disease processes and/or stages are included). GO enrichment terms for top 250 BN+1 genes from tubule expansion model meeting Bonferroni-corrected p-value < 0.05.

5.5 Discussion

Bayesian network analysis coupled with network refinement and expansion algorithms revealed differential roles of Jak/Stat pathway members in two kidney compartments during progressive kidney disease. First, the set of network interactions predicted for each compartment and refined using the EdgeClipper were disparately low and indistinguishable from noise, suggesting that gene expression of Jak/Stat members is distinct for the two compartments and reflects different biological mechanisms. This finding implicates the Jak/Stat pathway in at least one (if not both) of the compartments. Second, expansion of the two refined network models using our Bayesian network expansion identified novel sets of genes with biological roles distinct for the glomerular and tubule compartments corresponding to known disease mechanisms in those compartments. Thus, these results suggest that the Jak/Stat pathway is in fact involved in the different glomerular and tubule methods, and that our approach can identify relevant genes for additional validation and analysis in multi-tissue experimental studies.

One interesting aspect of this analysis was the effect of selecting a known pathway and investigating its regulation using data from two different biological compartments with differential disease regulation. Our results from the BN+1 analyses revealed that those compartments show differential patterns of interactions when considering genes from the same pathway. These results suggest that despite a preferential selection of an existing known pathway for biological analysis, the resulting predictions for the BN and BN+1 analyses will more closely and specifically reflect the underlying data and hence biological conditions. The most convincing results came from the BN+1 results, where it was demonstrated via GO enrichment that the neighborhood of genes (with best BN scores and hence top ranks) predicted around the core network specifically reflect the disease processes in the glomerular and tubule compartments.

Another interesting observation from the BN+1 analysis was the preservation of selected modular structures or sub-networks for the partial glomerular and tubule dataset models in the shared core network to BN+1 gene connections. Each BN+1 simulation included an initial structural prior from one of the top-scoring core networks in the initial BN searches. This core network was often modified during the BN+1 search for any given BN+1 gene's network, though only a handful of interactions were removed. The majority of interactions from the core network were preserved. This conservation of sub-network structures during BN+1 was consistent with what was observed during our other analyses, such as the *E. coli* ROS detoxification pathway analyses in Chapters 2 and 5.

However, an important note is that BN+1 genes sometimes drew from multiple disconnected modules. These results suggest that the expansion of only small biological cores or modules may be an inherent modeling bias which may not necessarily reflect the biological complex or important regulators across large biological pathways. It may be possible to identify genes and other biological entities with multiple roles in complex biological systems and disease. This area of computational analysis could be explored in future studies.

The most exciting finding was the prediction of PLCE1 in the glomerular compartment. This gene has already been implicated in the progressive kidney disease, so our prediction is supported by these previous data. We hope to continue investigation of this gene and its role in kidney disease. Another interesting result was the discovery that several genes from the glomerular model have known neuropathy functions (e.g. FAM134B). Hence, there may be similar regulatory or effector genes which may be involved in another diabetic complication: diabetic neuropathy. This claim would need to be investigated further for additional support and confidence.

Chapter 6

Enabling Enhanced BN Approaches Online in MARIMBA

6.1 Introduction

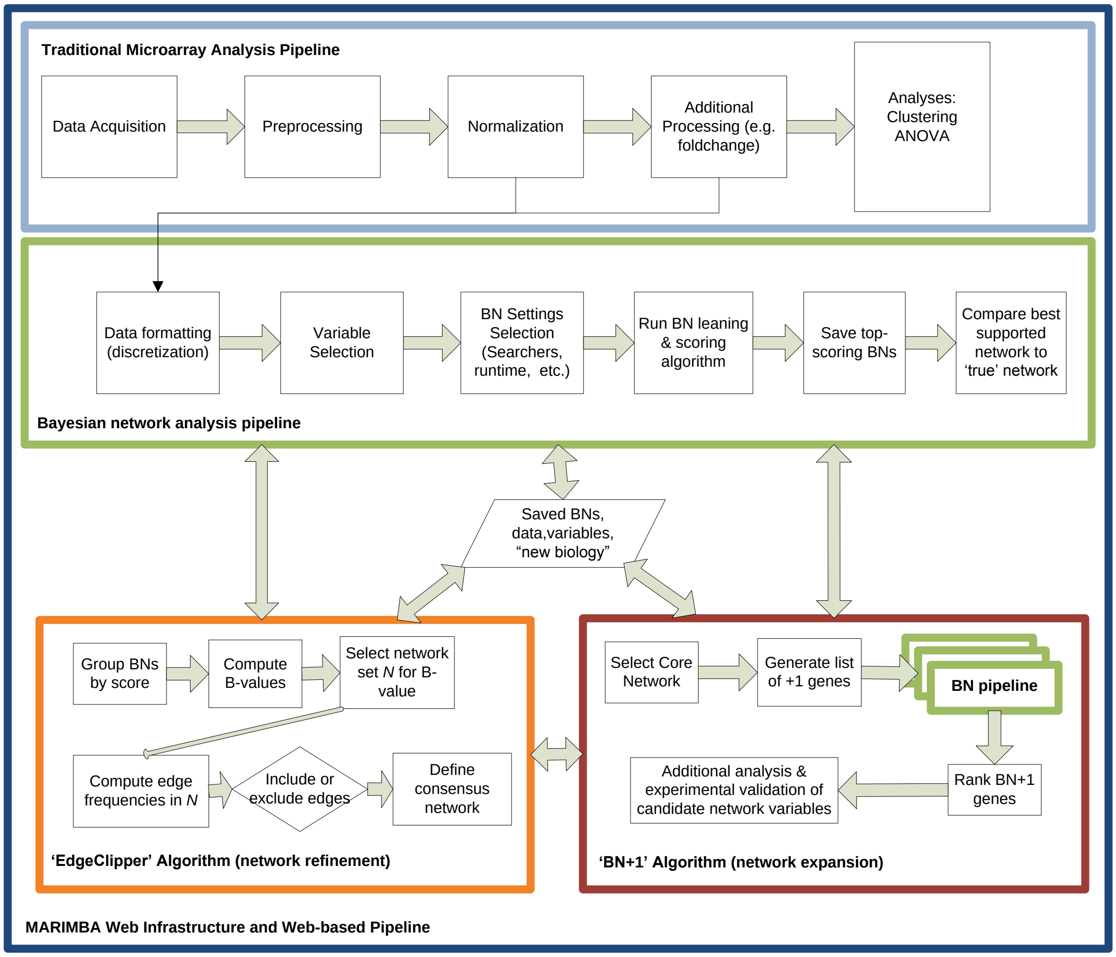
In the preceding chapters, Bayesian network expansion and refinement algorithms were introduced. The purpose of this chapter is to explore the software infrastructure which was developed and used to achieve those previous studies' goals. The web-based tool, called MARIMBA, was designed to permit fast formatting, execution, and analysis of Bayesian networks when using high-throughput biological datasets.

MARIMBA, the Molecular Annotation Resource for Integrating Microarrays with Bayesian Analysis, was originally designed as an annotation resource to map microarray features to corresponding genes and proteins. Over the course of this thesis, MARIMBA was redesigned to answer specific questions in each of the previous chapters for the Bayesian analysis. Thus, MARIMBA has evolved, albeit painstakingly, into a web-based tool for Bayesian network expansion and refinement (Figure 6.1). MARIMBA is accessible at http://marimba.hegroup.org. The welcoming page of MARIMBA is shown in Figure 6.2. The general design and workflow of the system is described, and future developments are suggested.

6.2 MARIMBA Software Pipeline

The main MARIMBA system architecture and pipeline for analysis of project data is described in Figure 6.2 and contains the following steps: *(1) Data Selection, (2) Variable*

Selection, (3) File writing, (4) Preprocessing/Clustering, (5) BN settings selection, (6) BN execution, (7) Visualization and analysis, (8) EdgeClipper analysis, and (9) BN+1 analysis.

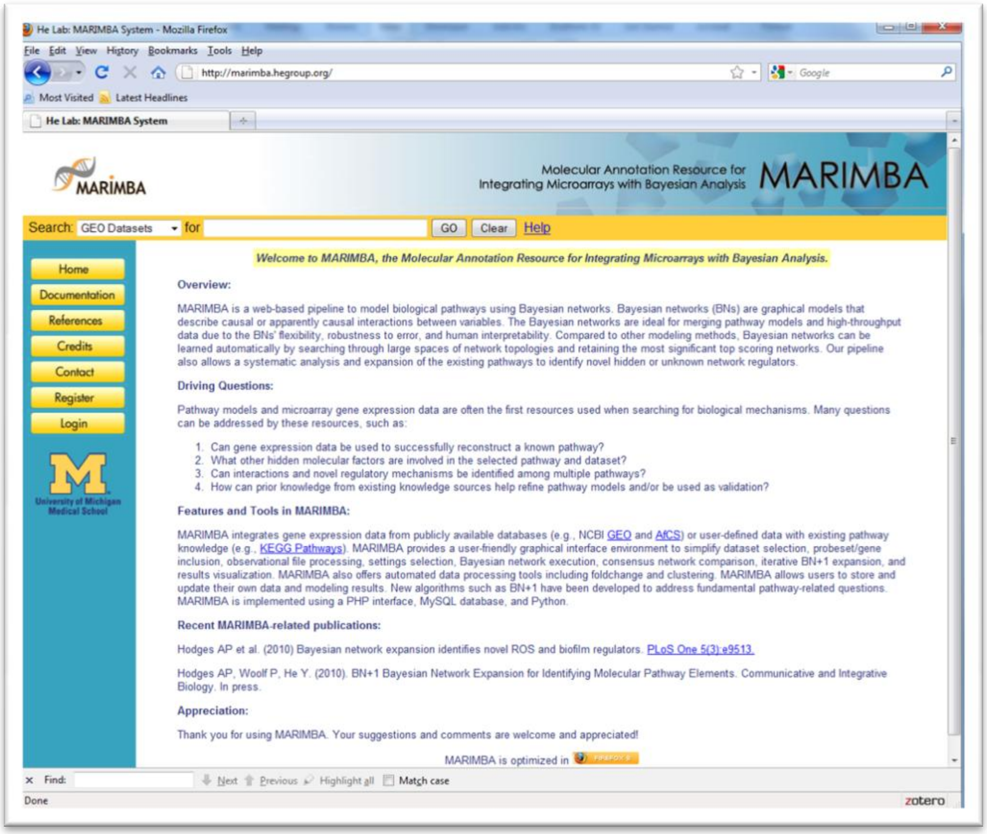


Traditional Microarray Analysis Pipeline
Data Acquisition
Preprocessing
Normalization
Additional Processing (e.g. foldchange)
Analyses: Clustering ANOVA
Data formatting (discretization)
Variable Selection
BN Settings Selection (Searchers, runtime, etc.)
Run BN leaning & scoring algorithm
Save top-scoring BNs
Compare best supported network to 'true' network
Bayesian network analysis pipeline
Saved BNs, data,variables, "new biology"
Group BNs by score
Compute B-values
Select network set N for B-value
Compute edge frequencies in N
Include or exclude edges
Define consensus network
'EdgeClipper' Algorithm (network refinement)
Select Core Network
Generate list of +1 genes
BN pipeline
Rank BN+1 genes
Additional analysis & experimental validation of candidate network variables
'BN+1' Algorithm (network expansion)
MARIMBA Web Infrastructure and Web-based Pipeline

Figure 6.1 Overview of the implemented methods for BN, BN+1 and EdgeClipper in the MARIMBA web pipeline.

6.2.1 Data selection

Biological or other data can be uploaded into MARIMBA for BN, BN+1, and EdgeClipper analyses. Currently the data must be formatted as a tab-delimited text file with "ID" entered in the first cell (first column, first row) of the text file. Transpose



He Lab: MARIMBA System - Mozilla Firefox
File Edit View History Bookmarks Tools Help
http://marimba.hegroup.org/
Google
Most Visited
Latest Headlines
He Lab: MARIMBA System
MARIMBA
Molecular Annotation Resource for Integrating Microarrays with Bayesian Analysis
MARIMBA
Search:
GEO Datasets
for
GO
Clear
Help
Home
Documentation
References
Credits
Contact
Register
Login
University of Michigan Medical School
Find:
Next
Previous
Highlight all
Match case
Done
zotero

Welcome to MARIMBA, the Molecular Annotation Resource for Integrating Microarrays with Bayesian Analysis.

Overview:

MARIMBA is a web-based pipeline to model biological pathways using Bayesian networks. Bayesian networks (BNs) are graphical models that describe causal or apparently causal interactions between variables. The Bayesian networks are ideal for merging pathway models and high-throughput data due to the BNs' flexibility, robustness to error, and human interpretability. Compared to other modeling methods, Bayesian networks can be learned automatically by searching through large spaces of network topologies and retaining the most significant top scoring networks. Our pipeline also allows a systematic analysis and expansion of the existing pathways to identify novel hidden or unknown network regulators.

Driving Questions:

Pathway models and microarray gene expression data are often the first resources used when searching for biological mechanisms. Many questions can be addressed by these resources, such as:

1. Can gene expression data be used to successfully reconstruct a known pathway?
2. What other hidden molecular factors are involved in the selected pathway and dataset?
3. Can interactions and novel regulatory mechanisms be identified among multiple pathways?
4. How can prior knowledge from existing knowledge sources help refine pathway models and/or be used as validation?

Features and Tools in MARIMBA:

MARIMBA integrates gene expression data from publicly available databases (e.g., NCBI GEO and AfCS) or user-defined data with existing pathway knowledge (e.g., KEGG Pathways). MARIMBA provides a user-friendly graphical interface environment to simplify dataset selection, probeset/gene inclusion, observational file processing, settings selection, Bayesian network execution, consensus network comparison, iterative BN+1 expansion, and results visualization. MARIMBA also offers automated data processing tools including foldchange and clustering. MARIMBA allows users to store and update their own data and modeling results. New algorithms such as BN+1 have been developed to address fundamental pathway-related questions. MARIMBA is implemented using a PHP interface, MySQL database, and Python.

Recent MARIMBA-related publications:

Hodges AP et al. (2010) Bayesian network expansion identifies novel ROS and biofilm regulators. PLoS One 5(3):e9513.

Hodges AP, Woolf P, He Y. (2010). BN+1 Bayesian Network Expansion for Identifying Molecular Pathway Elements. Communicative and Integrative Biology. In press.

Appreciation:

Thank you for using MARIMBA. Your suggestions and comments are welcome and appreciated!

MARIMBA is optimized in FIREFOX 3

Figure 6.2 Screenshot of MARIMBA home page.

options are available for selecting the appropriate conditions and variables in the analysis. Existing data files, such as previously-uploaded user data or featured MARIMBA data, can also be selected and used for analysis.

Note that in this step of the analysis, all of the data to be included in the BN, BN+1 and/or EdgeClipper analyses must be included in the same tab-delimited data file. Subsequent steps in the MARIMBA workflow assume that data will be taken from this 'master' data file.

6.2.2 Variable selection

Several options are available for variable selection. First, users can directly select which variables to include by copying and pasting data into a text field or uploading a file with those variable names. A second option is the selection of variable from existing databases such as KEGG. Users are required to enter a valid KEGG pathway ID. MARIMBA returns existing information from the KEGG database and attempts to match those identifiers to the user data fields. The KEGG selection option was deprecated recently to allow more expert data file generation and preprocessing by the users.

6.2.3 Write step for BN and BN+1 files

The basic MARIMBA-formatted files are then generated dynamically for use in static Bayesian modeling. Individual conditions (user-generated identifiers) can be specified using an interactive webpage. Conditions, included genes, and analysis method (BN, BN+1, or SYNTH) were selected at this step.

The write step webpage allows verification of all settings, such as method of gene combination (averaging or top probe selection) and type of file write (BN or BN+1 write). During the writing process, BANJO-format [40] data files are generated. In the cases of multiple or redundant probeset identifiers per specified gene, averaging was employed. Thus, multiple occurrences were treated as replicates, and were averaged at the respective treatment and time. In summary, the final dataset was a BANJO-format data file with rows being unique observations and columns being uniquely-identified genes.

6.2.4 Data processing

Data can be preprocessed with available fold change or clustering tools. A custom python script was created to permit fold change comparison of user-selected treatment versus control samples. Users may select one or more samples for each control and treatment groups. The selected control and treatment chips are averaged separately prior to calculating the fold-change between these two groups. The GUI allows specification of both groups, as well as the threshold for probeset inclusion. Selected probeset results are listed on a subsequent page.

Clustering tools were selected from Pycluster, a Python-version of the C clustering library (http://bonsai.ims.u-tokyo.ac.jp/~mdehoon/software/cluster/software.htm). MARIMBA currently permits k-means and k-median clustering of the working dataset. Graphical results are displayed dynamically for each cluster using Matplotlib (http://matplotlib.sourceforge.net/) and R, including a jpeg image of the individual cluster, checkbox selection of individual clusters, and lists of probesets with links to annotation information. Individual probesets are selected subsequently after cluster selection on a second page.

6.2.5 BN parameter selection

BN simulation settings were selected after completing the data and gene selection processes, respectively. A static Bayesian network simulation was created to analyze the microarray data. Simulated annealing is most commonly selected as the searcher method, due to its improved performance over greedy searches when no prior knowledge of underlying structure is available [64] and is the recommended strategy in MARIMBA. A relatively low cooling factor was implemented to allow less restrictive searching of the sample space and potentially identify as many equivalence classes for the top-scoring network as possible. The simulation included storage of 1,000 networks for comparative purposes. Note that this number of networks is restricted to smaller numbers (assumedly ~100-500) to limit report file size.

6.2.6 Execution of Bayesian network modeling

BN files are submitted via the online interface in MARIMBA. In an earlier version of MARIMBA, each dataset was transferred to a server at the University of Michigan prior to Xgrid simulation. In XGrid, a query is based from a controller to one or more agents. The XGrid is used to pass new BN and BN+1 analyses to free agents on the server. MARIMBA was upgraded to use resources from the Center for Applied Computing at the University of Michigan (CAC), and is under further development for other cloud-based computing strategies. However, in each submission infrastructure, each available agent runs a unique BANJO simulation (e.g. the 1,000 bootstrap simulations in the synthetic network EdgeClipper analyses).

Individual data are passed to the Xgrid or similar submission grid with all conditions and variable labels removed in order to protect the identities of user information. In this regard, the observational file, settings file, and prior knowledge file are tarred and passed to the Xgrid server. Individual variable lists are retained on the He Group servers to protect the integrity and identity of variables and conditions included in an individual analysis. The controller on the opposing grid posts status updates to the primary MARIMBA server via a MySQL table and SSH. After completion of an individual BN or BN+1 analysis, individual report files are returned to the He Group servers. The BN+1 analyses are completed after successful updating of all BN+1 probeset simulations.

6.2.7 EdgeClipper analysis

EdgeClipper is integrated in a convenient fashion within a consensus network tool in the visualization interfaces. Users may select either the top-level consensus, or specific a B-value or C-value cutoff for edge pruning. The current implementation of EdgeClipper returns the most well-supporting interactions meeting the imposed cutoff. The EdgeClipper approaches were recently established and validated, and as such are undergoing additional updates for a more seamless user experience in MARIMBA.

6.2.8 BN+1 analysis

The core BN network is employed as a fixed topology/prior knowledge network in the BN analysis. Probeset selection will be designed as above for the standard BN analysis. In addition, probesets not included in the BN structural file are included in the BN+1 list. BN core files, including the BN settings, dataset, probeset list, and report file are required for BN+1 analysis. A unique BANJO analysis is created for each BN+1 probeset. The BN core files are copied from a previous analysis if not present in the current analysis. Users may select whether the BN core network is a required fixed topology or unfixed starting network.

6.2.9 BN Result display and interpretation

Model averaging and equivalence class searching were then implemented to determine the “core BN” network model. Here, model averaging was defined as inclusion of an edge between two genes if that edge appeared in more than *X* percent of the top-scoring networks with identical score (with X bounded between 0 and 100), and is most oftenly implemented with *X* = 100.

MARIMBA provides several unique features versus the standalone BANJO system. First, MARIMBA is exclusively web-based and allows seamless integration of user project management, analysis construction, BN submission to a distributed computing environment, and analysis and visualization of results. User-friendly GUI environments simplify the dataset selection, probeset/gene inclusion, observational file processing, and settings selection for BANJO. Such features are necessitated for efficient querying by biologists who wish to use such BN tools to analyze their data. The user interface and project/analysis management approach permit large-scale analyses such as BN+1.

Top-scoring networks are displayed as jpeg images on-the-fly, such that the images are converted directly from their original dot files. Furthermore, MARIMBA displays top-scoring networks of BN, BN+1, and combined networks to the user. The BN+1 display environment provides plots for probability of each network in the query, thus enabling comparison of networks for relevance and likelihood.

After selection of modeling parameters, Bayesian networks are searched in BANJO (http://www.cs.duke.edu/~amink/software/banjo) by simulated annealing or greedy algorithm. Top-scoring networks or consensus networks shared by top networks are displayed graphically in a web based GUI using DOJO software (http://www.dojotoolkit.org). To determine if the addition of any single gene would improve the top network score when added to the network, a "BN+1" approach was developed to recalculate the Bayesian networks by iteratively adding a gene from a defined gene list to an existing top network (prior structure) and subsequent BN recalculation for the new gene list. Each new "BN+1" query per gene is recalculated individually on an individual XGrid agent in the Woolf lab Mac cluster.

As described in Chapter 3, the currently-implemented tools in MARIMBA are being rewritten in Python for direct access and self-utilization by scientific researchers. These new approaches will allow individualized tailoring of EdgeClipper and BN+1 for different groups' requirements and computing interests on a variety of computing platforms. The tools are being tested both on standalone laptops and personal machines, and will be tested in the future with other cloud-based computing architectures. These changes will make EdgeClipper and BN+1 even more power and amenable to massively-high throughput and next-generation dataset analysis and global interactome studies.

6.2.10 Hardware configureation

MARIBMA is built on one Dell Poweredge 2580 server which runs the Redhat Linux operating system (Redhat Enterprise Linux ES 4) and Apache HTTP Server. MySQL database and different programming languages including PHP, Perl, and Python are used for development of a variety of MARIMBA components. The MARIMBA data is backed up in another Dell Poweredge 2580 server regularly. A three-tier system architecture is implemented with two Linux servers.

Chapter 7

Future Directions and Conclusions

7.1 Summary and Discussion of Previous Sections

7.1.1 Overview of Chapters

In this thesis, I have described the development, implementation, and interpretation of two novel Bayesian network approaches for biological pathway expansion and refinement, EdgeClipper and BN+1. Chapter 1 described the driving motivation for developing new approaches to refine and expand networks such as Bayesian networks. Chapter 2 provides an overview of the major theory and assumptions associated with our Bayesian network approach. In Chapter 3, the novel EdgeClipper algorithm was designed and tested to refine existing Bayesian networks to identify the most likely interactions supported by the models and the underlying biological data. The BN+1 algorithm was then developed and tested to identify novel hidden variables which likely participate in selected pathways such as ROS detoxification, B cell receptor signaling, and synthetic networks in Chapter 4. After establishing them as valid approaches, EdgeClipper and BN+1 combined and applied to the refinement and expansion of the *E. coli* ROS and human Jak/Stat signaling pathways in Chapter 5. In Chapter 6, I briefly summarize the infrastructure used to generate all of the analyses. And in the current chapter, I summarize many of the findings described previously as well as new areas of exploration.

7.2 Summary of algorithms and major findings

7.2.1 Summary of EdgeClipper algorithm and findings

A unique feature of this analysis was the inclusion of the probability distributions for the top scoring Bayesian networks when assigning ranks to network edges. The approach itself is robust to a highly-parallelized search procedure, which is itself an important consideration, and can be more easily interpreted via an expectation-value metric or B-value. One of the most exciting aspects of the EdgeClipper analysis was its ability to be incorporated along with the BN+1 algorithm to initially refine, and later expand, the pathway network and identify new BN+1 genes for selected pathway genes. I demonstrated that not only is the approach comparable to existing bootstrapping and frequency-based methods (and even inclusive of the frequency method in the EC-F derivation) but also computationally much faster than the traditionally bootstrapping approach. Comparison of the different EC derivations to bootstrapping and existing knowledge from pathway databases can also identify novel interactions which are strongly supported by the underlying data and warrant additional experimental and computational analysis.

The EC-based analysis can help us understand the question of overlap between computational models and knowledge-based pathway networks. From our results, the extent of overlap between the computational networks and biological networks depends on the ability of the algorithm to recapture complex relationships as well as the sufficient representation of those relationships within the underlying dataset. For example, the 29-42% concordance between our networks and the known ROS pathway (Chapter 3) suggest that the Bayesian networks may only be able to identify some known interactions.

One possible reason for this is the types of data used in the modeling, namely transcriptional expression data. If the expression profiles for selected genes in a pathway do not correlate well with the levels and activities of their translated protein products, then the mRNA data may not give adequate support when used to make conditional probability tables and inferences in the Bayesian networks. Someone might argue that protein expression profiles instead may present a better candidate dataset for training the BNs when considering the pathway-level activities. However, these datasets are harder to obtain experimentally. Their argument may be appropriate for selected pathways, assuming that no major feedback occurs between the transcriptional regulatory network and protein signaling pathway. The NF-κB sub-network in BCR signaling is a poor example, since the downstream transcriptional changes induced after NF-κB translocation to the nucleus have major effects on the protein-level signaling pathways (even to the extent of cellular death or apoptosis). It may even be possible to generate a BN trained on protein-level data (e.g. protein-protein interactions) which underperforms the mRNA-trained BN network if the transcriptional network has a greater effect on the pathway activities, or if a variety of biological responses from different hierarchical levels provide moderate contributions to the pathway activities and regulation.

It may still be possible to identify other interactions in the protein-level signaling network using other complementary datasets for independent BN analyses. I hypothesize that some though not all interactions will be recovered using the different datasets, most specifically in the cases where transcriptional and translational machinery (and other processes) are well coordinated and regulated. Where these machinery do not act in concert, we can expect that different Bayesian networks trained on different datasets representing different biological scales will show lower overlap though more distinct interaction hypotheses. However, care must be exercised when comparing models based on different datasets, since the meanings of the statistical influences in a biological context may be interpreted differently.

7.2.2 Summary of BN+1 algorithm and findings

The BN+1 algorithm was introduced and tested using both synthetic network analysis and a relevant genetic regulatory pathway in *E. coli*. Synthetic networks were designed to test the overall performance of the BN+1 procedure. Those simulations successfully benchmarked the BN+1 procedure and illustrated several useful considerations when conducting BN+1 analyses. The BN+1 expansion of the ROS detoxification pathway using publicly-available gene expression data successfully identified known and unknown interactors or regulators for ROS core genes. One of the major findings was the prediction of an influence or interaction between *GadX* and *UspE*, followed by the prediction and later verification of their direct involvement in biofilm formation. Hence, the BN+1 procedure directly identified a new biological mechanism for this novel ROS gene which can be further investigated in future studies. Many other exciting predictions were generated in the BN+1 procedure which can be studied in future projects.

7.2.3 Summary of Combined EC and BN+1 findings

The ability to combine the EdgeClipper refinement algorithm along with the BN+1 approach was investigated in Chapter 4, when both approaches were applied to the analysis of different datasets for progressive kidney disease in glomerular and tubule kidney compartments from *H. sapiens*. Four major datasets were generated, either using all available data for a respective compartment, or only those data per compartment which were known diabetes mellitus or normal data from Pima Indians. Bayesian networks were generated for 131 genes from the known Jak/Stat signaling pathway, and refined using the EdgeClipper algorithm. The EdgeClipper algorithm served a vital function in reducing the number of genes for the subsequent BN+1 algorithm, since the number of genes in each simulation was roughly the same as the number of data observations. The number of observations was too small for the relatively large network size, and hence EdgeClipper was required to refine the networks and identify the most conserved or well-supported interactions to include in the BN+1 core network. This behavior was supported by properties of the networks in our preliminary simulations which had unexpectedly shorter run times.

Most significantly, the BN+1 expansion of the core networks (despite lower runtimes for generating core networks) were able to identify distinct sets of BN+1 genes for each compartment which reflect the different stages of progressive kidney disease in those compartments. The simulations also identified previously-hypothesized gene interactors which are likely involved in the progression of the kidney disease in those compartments. This is an exciting finding, which suggests that the incorporation of refinement and expansion algorithms is both achievable and applicable for studying complex biomedical phenomena in multiple tissues or compartments.

7.2.4 Integrated analysis in MARIMBA

Finally, the integration of the EdgeClipper and BN+1 algorithms into a web-based infrastructure was described in Chapter 5. The Molecular Annotation Resource for Integrating Microarrays with Bayesian Analysis (MARIMBA) was introduced. Formatted microarray or other high-throughput datasets can be uploaded into MARIMBA or selected from the site for BN analysis. Variables (e.g. genes, proteins) in the data can be selected for inclusion in the initial BN run. Some tools for processing and additional variable selection are available, though these pipelines are a work-in-progress. Bayesian networks can be constructed and analyzed, followed by EdgeClipper refinement, BN+1 execution, and results visualization. MARIMBA was developed for and applied to the major topics in this thesis, and has been used for several collaborative projects not discussed in the thesis (with several publications in process). MARIMBA is constantly undergoing updates and will also be submitted for publication soon.

7.2.5 Revisiting the computational versus knowledge-based networks

This thesis was successful in better understanding the overlap between computational and knowledge-based networks, as well as in uncovering new biological entities and interactions which do not yet appear in the existing knowledge repositories (e.g. pathway databases). Hence, the developed approaches further increase the shared overlap between the computational networks (e.g. BNs), knowledge-based networks (e.g. pathway representation in EcoCyc or KEGG), and the ‘real’ underlying biology. Experimental validation following the computational predictions further establishes our confidence in this overlap with reality.

There are some cases where Bayesian networks will identify spurious interactions, though these are likely removable using the developed EdgeClipper approaches. On a related note, all of the approaches rely upon sufficient data for reliable predictions. Insufficient quantities of data and even biased representation of selected experimental conditions may have major effects on the recovery of known and putative novel interactions. Furthermore, the somewhat biased selection of experimental conditions and their sufficient representations may have an effect on which biological pathways and systems are best modeled using the BN, EdgeClipper, and BN+1 approaches.

7.3 Future work and extensions

7.3.1 Investigation of other novel ROS pathway genes

In one of the BN+1 analyses described in Chapter 4, we identified a ranked list of BN+1 genes for a ROS pathway network. A subset of the top-ranked genes was investigated using literature searching and comparison to existing databases to investigate their role in ROS activities. However, in Chapter 3, three major consensus networks identified using the EdgeClipper algorithm were also expanded and used to identify distinct sets of genes with implicated roles in ROS activity. Many of these genes were only superficially investigated, and could be studied in much more detail.

For example, one question regarding the preferential selection of the BN+1 genes is whether those genes share common gene regulators. Many genes in the *E. coli* genome are regulated by global factors called sigma factors [117]. Some sigma factors have specified roles in oxidative and cellular stress, so one might expect these sigma factors to have overrepresented sets of target genes in the top BN+1 results. A simple method to determine this behavior would be to generate the average rank of the top 10 genes for each sigma factor, and rank the sigma factors according to the average rank of their target genes.

7.3.2 Modular behaviors of the BN+1 and EdgeClipper algorithms

In this thesis, the BN+1 and EdgeClipper algorithms were implemented to expand and refine, respectively, Bayesian network models trained using gene expression data. One interesting property of both algorithms was the appearance of modular architecture in networks at different times. Several existing approaches have studied modules which share genetic regulation and/or conserve biological functions. Here, I discuss the preliminary data which suggested that both of our developed algorithms do relate to a modular prediction framework.

In terms of the BN+1 analysis framework, I investigated which genes in the selected ROS detoxification pathway genes were most likely involved in the recruitment and preferential ranking of BN+1 genes. The hypothesis for this analysis was that certain genes in the selected core network would share edges in the BN+1 networks with the added BN+1 expansion gene. Furthermore, those core genes would more often connect to the top-scoring BN+1 genes. A matrix representation of the connections between

BN+1 expansion and core genes was generated to show preferential connection to a subset of core pathway genes. As expected, some core network genes show multiple connections to the BN+1 genes, whereas others show few to no connections. A surprising property was that multiple genes with similar biological roles would share many of the same connections to targeted core genes. These data suggest that the selection of the core network genes may have an important effect on both the ordering as well as conserved biological roles of BN+1 genes during the expansion algorithm.

Some modular architecture was also implied by the fragmentation of networks during EdgeClipper's generation of consensus networks. From the preliminary analysis, it was observed that those interactions which tended to have the best Pearson correlations or most definitive nonlinear patterns were least likely to disappear from the more conserved or well-supported consensus networks with smaller B-values. Hence, it is likely that many members of a biological pathway may not interact at the genetic level, and that only subsets of genes within the pathway do interact with each other. Other influences from different biological scales, such as protein, sRNA, and miRNA may complicate the ability to predict such interactions. This behavior is more likely if the gene expression patterns do not correlate directly with the behavior of those post-transcriptional and post-translational entities.

7.3.3 Applications to miRNA prediction

An exciting area of research is the prediction of novel microRNA (miRNA) targets using BN+1. MiRNAs are often 22-nucleotide RNA species which are cis- or trans-regulators for a gene, binding upstream or downstream of the gene and controlling its expression by targeting the product mRNA and either degrading or down-regulating that target. It is known that many miRNAs share similar genetic targets and often target similar members of the same pathway. This prior knowledge could be used to select appropriate core networks for subsequent expansion. For example, in Section 6.2, the BN+1 was demonstrated to preferentially identify BN+1 targets for certain genes with conserved biological roles and regulation. We expect that the selection of known pathway genes with either similar regulation or overlapping sets of miRNA regulators could be established as a core set of genes for BN generation and subsequent BN+1 expansion. An underlying assumption here is that unknown miRNA regulators which operate in similar biological contexts to the known miRNA regulators for those pathway genes should preferentially score better than other putative miRNA targets with no direct interaction in the pathway. Some miRNAs with few pathway targets may still appear in the top BN+1 results assuming relatively high correlations or conserved nonlinear interactions. Similar patterns were observed during the ROS pathway expansion for selected genes, lending some support to this claim. A major challenge is obtaining a representative dataset with paired miRNA and mRNA expression data and sufficient observations in order to conduct this exploratory analysis.

7.3.4 Applications to next-gen sequencing technologies

Another exciting area of research is the analysis and incorporation of next-generation sequencing technologies with Bayesian networks for biomarker discovery, disease analysis, and translational applications for personalized medicine. The exciting aspect of this research is the ability to measure an individual's expression profiles for hundreds of thousand of putative expressed transcripts. These new technologies can allow investigation of the effects of individual mutations and individualized genome on disease progression and response to drugs, food, and the environment. There are many studies ongoing at the NIH and other institutes which are specifically implementing next-gen sequencing for biomedical research (e.g. drug responses, the microbiome, etc.).

A major challenge in this field is the size and quantity of data generated by the technologies. In our recent analyses, we have explored those biological networks with a semi-reductionist approach. We have often designed networks to include less than 150 variables (genes), and to reflect specific biological pathways with known documented interactions. Upcoming analyses will require 10-100 times as many variables to be included in the network analysis, which presents a major computational issue. Some of the problems with generating large-scale network models are the amount of run-time for simulations and the amount of data needed to give an accurate prediction. Towards this regard, it is expected that similar refinement and pathway-based selection approaches will be needed to make BN simulations computationally feasible. Furthermore, distributed and highly-parallelizable simulation architectures with a cloud-based computing infrastructure will be required. The MARIMBA infrastructure in this thesis is a preliminary model and has provided many useful insights into the requirements for such analyses in future web systems.

7.3.5 Bayesian networks and natural language processing

The availability of prior knowledge is often advantageous as a source of validating knowledge, or as starting structural priors. There is an abundant source of interesting correlative data, hypotheses, and documented interactions buried within published literature in the form of text, tables and figures. Current natural language processing (NLP) technologies can generate on-the-fly relationships amongst genes, proteins, and other variables using defined semantic rules when parsing textual documents.

Many challenges, unfortunately, remain. For example, not all documents are currently available to researchers for full textual searching, and are often restricted to the publicly-available resources at PubMed Central. Intellectual property rules have also limited some document searching to abstract searches, which may significantly limit the extent of biological knowledge to be extracted automatically from these sources. Furthermore, the availability of searchable and automatically interpretable figures and tables using new representations and semantics is significantly lower than that of textual information. New algorithms for automatic figure interpretation and network or model generation would be highly desirable for computational and experimental researchers alike.

7.4 The future and beyond

It will be interesting to see how the BN framework as well as other approaches (MI, ODE, neural network, fitness functions, etc.) will adapt to the onslaught of next-generation (next-gen) sequencing technologies and parallelized experimental protocols across multiple biological scales. Bioinformatics is a constantly evolving field. Despite its infancy, bioinformatics is providing major changes to our conceptualization of health, disease, and individuality. Unfortunately, the traditional microarray analyses are being phased out in larger studies as the new sequencing and assay technologies are adopted by major institutions. Thus, several of the issues which were discussed and targeted in this thesis will be replaced by many other issues and considerations. However, our approaches can be modified in future studies to incorporate the new protocols, pipelines, and assumptions. Reductionist strategies such as the selection of known pathway genes may yet be implemented for these future technologies and serve as a starting point to benchmark newer pipelines and analyses. It is an exciting time in biomedical research, and many new findings are expected throughout the next several decades for prokaryotic throughout the higher eukaryotes.

References

1. Kanehisa M, Goto S (2000) KEGG: kyoto encyclopedia of genes and genomes. Nucleic Acids Res 28: 27-30.
2. Schaffer AA, Aravind L, Madden TL, Shavirin S, Spouge JL, et al. (2001) Improving the accuracy of PSI-BLAST protein database searches with composition-based statistics and other refinements. Nucleic Acids Res 29: 2994-3005.
3. Keene JD (2001) Ribonucleoprotein infrastructure regulating the flow of genetic information between the genome and the proteome. Proc Natl Acad Sci U S A 98: 7018-7024.
4. Ma'ayan A, Jenkins SL, Neves S, Hasseldine A, Grace E, et al. (2005) Formation of regulatory patterns during signal propagation in a mammalian cellular network. Science 309: 1078.
5. Seok J, Xiao W, Moldawer LL, Davis RW, Covert MW (2009) A dynamic network of transcription in LPS-treated human subjects. BMC Syst Biol 3: 78.
6. Scherf U, Ross DT, Waltham M, Smith LH, Lee JK, et al. (2000) A gene expression database for the molecular pharmacology of cancer. Nat Genet 24: 236-244.
7. Margolin AA, Califano A (2007) Theory and limitations of genetic network inference from microarray data. Ann N Y Acad Sci 1115: 51-72.
8. Faith JJ, Driscoll ME, Fusaro VA, Cosgrove EJ, Hayete B, et al. (2008) Many Microbe Microarrays Database: uniformly normalized Affymetrix compendia with structured experimental metadata. Nucleic Acids Res 36: D866-870.
9. Slonim DK (2002) From patterns to pathways: gene expression data analysis comes of age. Nat Genet 32 Suppl: 502-508.

10. Hashimoto RF, Kim S, Shmulevich I, Zhang W, Bittner ML, et al. (2004) Growing genetic regulatory networks from seed genes. Bioinformatics 20: 1241-1247.

11. Pena JM, Bjorkegren J, Tegner J (2005) Growing Bayesian network models of gene networks from seed genes. Bioinformatics 21 Suppl 2: ii224-229.

12. Basso K, Margolin AA, Stolovitzky G, Klein U, Dalla-Favera R, et al. (2005) Reverse engineering of regulatory networks in human B cells. Nat Genet 37: 382-390.

13. Fleischmann RD, Adams MD, White O, Clayton RA, Kirkness EF, et al. (1995) Whole-genome random sequencing and assembly of Haemophilus influenzae Rd. Science 269: 496-512.

14. Collins FS, Green ED, Guttmacher AE, Guyer MS (2003) A vision for the future of genomics research. Nature 422: 835-847.

15. Lander ES, Linton LM, Birren B, Nusbaum C, Zody MC, et al. (2001) Initial sequencing and analysis of the human genome. Nature 409: 860-921.

16. Venter JC, Adams MD, Myers EW, Li PW, Mural RJ, et al. (2001) The sequence of the human genome. Science 291: 1304-1351.

17. Thomas DJ, Rosenbloom KR, Clawson H, Hinrichs AS, Trumbower H, et al. (2007) The ENCODE Project at UC Santa Cruz. Nucleic Acids Res 35: D663-667.

18. Birney E, Stamatoyannopoulos JA, Dutta A, Guigo R, Gingeras TR, et al. (2007) Identification and analysis of functional elements in 1% of the human genome by the ENCODE pilot project. Nature 447: 799-816.

19. Kanehisa M, Goto S, Furumichi M, Tanabe M, Hirakawa M (2010) KEGG for representation and analysis of molecular networks involving diseases and drugs. Nucleic Acids Res 38: D355-360.

20. Kanehisa M, Goto S, Hattori M, Aoki-Kinoshita KF, Itoh M, et al. (2006) From genomics to chemical genomics: new developments in KEGG. Nucleic Acids Res 34: D354-357.

21. Nishimura D (2001) BioCarta. Biotech Softw Int Rep 2: 117-120.

22. Salgado H, Gama-Castro S, Martinez-Antonio A, Diaz-Peredo E, Sanchez-Solano F, et al. (2004) RegulonDB (version 4.0): transcriptional regulation, operon organization and growth conditions in Escherichia coli K-12. Nucleic Acids Res 32: D303-306.

23. Keseler IM, Collado-Vides J, Gama-Castro S, Ingraham J, Paley S, et al. (2005) EcoCyc: a comprehensive database resource for Escherichia coli. Nucleic Acids Res 33: D334-337.

24. Jayapandian M, Chapman A, Tarcea VG, Yu C, Elkiss A, et al. (2007) Michigan Molecular Interactions (MiMI): putting the jigsaw puzzle together. Nucleic Acids Res 35: D566-571.

25. Nunoshiba T, deRojas-Walker T, Wishnok JS, Tannenbaum SR, Demple B (1993) Activation by nitric oxide of an oxidative-stress response that defends Escherichia coli against activated macrophages. Proc Natl Acad Sci U S A 90: 9993-9997.

26. Laval J (1996) Role of DNA repair enzymes in the cellular resistance to oxidative stress. Pathol Biol (Paris) 44: 14-24.

27. Volkert MR, Elliott NA, Housman DE (2000) Functional genomics reveals a family of eukaryotic oxidation protection genes. Proc Natl Acad Sci U S A 97: 14530-14535.

28. Zheng M, Wang X, Templeton LJ, Smulski DR, LaRossa RA, et al. (2001) DNA microarray-mediated transcriptional profiling of the Escherichia coli response to hydrogen peroxide. J Bacteriol 183: 4562-4570.

29. Lucas PC, McAllister-Lucas LM, Nunez G (2004) NF-kappaB signaling in lymphocytes: a new cast of characters. J Cell Sci 117: 31-39.

30. DeFronzo RA, Abdul-Ghani M (2011) Type 2 diabetes can be prevented with early pharmacological intervention. Diabetes Care 34 Suppl 2: S202-209.

31. Reimann M, Bonifacio E, Solimena M, Schwarz PE, Ludwig B, et al. (2009) An update on preventive and regenerative therapies in diabetes mellitus. Pharmacol Ther 121: 317-331.

32. Choudhury D, Tuncel M, Levi M (2010) Diabetic nephropathy -- a multifaceted target of new therapies. Discovery Medicine 10: 406-415.

33. Pearl J (1985) Bayesian networks: A model of self-activated memory for evidential reasoning. Irvine, CA: Computer Science Department, University of California.

34. Pearl J (1988) Probabilistic reasoning in intelligent systems: networks of plausible inference. San Mateo, CA: Morgan Kaufmann. 552 p.

35. Cooper GF, Herskovits E (1992) A Bayesian method for the induction of probabilistic networks from data. Mach Learn 9: 309-347.

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

37. Heckerman D, Geiger, D. (1995) Learning Bayesian networks: the combination of knowledge and statistical data. Mach Learn 20: 197-243.

38. Heckerman D (2008) A tutorial on learning with Bayesian networks. Innov Bayesian Netw: 33-82.

39. Chen Y, Kamat V, Dougherty ER, Bittner ML, Meltzer PS, et al. (2002) Ratio statistics of gene expression levels and applications to microarray data analysis. Bioinformatics 18: 1207-1215.

40. Smith VA, Yu J, Smulders TV, Hartemink AJ, Jarvis ED (2006) Computational inference of neural information flow networks. PLoS Comput Biol 2: e161.

41. Chickering DM (1996) Learning Bayesian networks is NP-complete. Learning from data: Artif Intell Stat 112: 121-130.

42. Schwarz G (1978) Estimating the dimension of a model. Ann Statist 6: 461-464.

43. Ward EJ (2008) A review and comparison of four commonly used Bayesian and maximum likelihood model selection tools. Ecological Modelling 211: 1-10.

44. Liddle AR (2007) Information criteria for astrophysical model selection. Mon Not R Astr Soc 377: L74-L78.

45. Bansal M, Belcastro V, Ambesi-Impiombato A, di Bernardo D (2007) How to infer gene networks from expression profiles. Mol Syst Biol 3: 78.

46. Djebbari A, Quackenbush J (2008) Seeded Bayesian Networks: constructing genetic networks from microarray data. BMC Syst Biol 2: 57.

47. Friedman N, Goldszmidt M, Wyner A. Data analysis with Bayesian networks: A bootstrap approach; 1999; San Francisco, CA. Morgan Kaufmann. pp. 206–215.

48. Friedman N, Koller D (2003) Being Bayesian about network structure. A Bayesian approach to structure discovery in Bayesian networks. Mach Learn 50: 95-125.

49. Bose R, Molina H, Patterson AS, Bitok JK, Periaswamy B, et al. (2006) Phosphoproteomic analysis of Her2/neu signaling and inhibition. Proc Natl Acad Sci U S A 103: 9773-9778.

50. Koch M, Broom BM, Subramanian D (2009) Learning robust cell signalling models from high throughput proteomic data. Int J Bioinform Res Appl 5: 241-253.

51. Armananzas R, Inza I, Larranaga P (2008) Detecting reliable gene interactions by a hierarchy of Bayesian network classifiers. Comput Methods Programs Biomed 91: 110-121.

52. Sakai S, Kobayashi K, Nakamura J, Toyabe S, Akazawa K (2007) Accuracy in the diagnostic prediction of acute appendicitis based on the Bayesian network model. Methods Inf Med 46: 723-726.

53. Gat-Viks I, Shamir R (2007) Refinement and expansion of signaling pathways: the osmotic response network in yeast. Genome Res 17: 358-367.

54. Needham CJ, Manfield IW, Bulpitt AJ, Gilmartin PM, Westhead DR (2009) From gene expression to gene regulatory networks in Arabidopsis thaliana. BMC Syst Biol 3: 85.

55. Parikh A, Huang E, Dinh C, Zupan B, Kuspa A, et al. (2010) New components of the Dictyostelium PKA pathway revealed by Bayesian analysis of expression data. BMC Bioinformatics 11: 163.

56. Hodges AP, Dai D, Xiang Z, Woolf P, Xi C, et al. (2010) Bayesian network expansion identifies new ROS and biofilm regulators. PLoS One 5: e9513.

57. Hodges A, Woolf P, He Y (2010) BN+ 1 Bayesian network expansion for identifying molecular pathway elements. Commun Integr Biol 3: 59-64.

58. Luo W, Hankenson KD, Woolf PJ (2008) Learning transcriptional regulatory networks from high throughput gene expression data using continuous three-way mutual information. BMC Bioinformatics 9: 467.

59. Team TRDC (2009) R: A Language and Environment for Statistical Computing. Vienna, Austria: R Foundation for Statistical Computing.

60. Efron B, Tibshirani R (1993) An introduction to the bootstrap. New York, NY: Chapman & Hall/CRC. 436 p.

61. Zar JH (1972) Significance testing of the Spearman rank correlation coefficient. J Am Stat Assoc 67: 578-580.

62. Chickering DM (2002) Learning equivalence classes of Bayesian-network structures. J Mach Learn 2: 445-498.

63. Xiang Z, Minter RM, Bi X, Woolf PJ, He Y (2007) miniTUBA: medical inference by network integration of temporal data using Bayesian analysis. Bioinformatics 23: 2423-2432.

64. Hartemink AJ, Gifford DK (2001) Principled computational methods for the validation and discovery of genetic regulatory networks. Massachusetts Institute of Technology, Ph D dissertation.

65. Salgado H, Santos A, Garza-Ramos U, van Helden J, Diaz E, et al. (1999) RegulonDB (version 2.0): a database on transcriptional regulation in Escherichia coli. Nucleic Acids Res 27: 59-60.

66. Subramanian A, Tamayo P, Mootha VK, Mukherjee S, Ebert BL, et al. (2005) Gene set enrichment analysis: a knowledge-based approach for interpreting genome-wide expression profiles. Proc Natl Acad Sci U S A 102: 15545-15550.

67. Sherman BT, Huang da W, Tan Q, Guo Y, Bour S, et al. (2007) DAVID Knowledgebase: a gene-centered database integrating heterogeneous gene annotation resources to facilitate high-throughput gene functional analysis. BMC Bioinformatics 8: 426.

68. Chen Y, Blackwell TW, Chen J, Gao J, Lee AW, et al. (2007) Integration of genome and chromatin structure with gene expression profiles to predict c-MYC recognition site binding and function. PLoS Comput Biol 3: e63.

69. Ideker T, Ozier O, Schwikowski B, Siegel AF (2002) Discovering regulatory and signalling circuits in molecular interaction networks. Bioinformatics 18 Suppl 1: S233-240.

70. Margolin AA, Nemenman I, Basso K, Wiggins C, Stolovitzky G, et al. (2006) ARACNE: an algorithm for the reconstruction of gene regulatory networks in a mammalian cellular context. BMC Bioinformatics 7 Suppl 1: S7.

71. Novak BA, Jain AN (2006) Pathway recognition and augmentation by computational analysis of microarray expression data. Bioinformatics 22: 233-241.

72. Rice JJ, Tu Y, Stolovitzky G (2005) Reconstructing biological networks using conditional correlation analysis. Bioinformatics 21: 765-773.

73. Soranzo N, Bianconi G, Altafini C (2007) Comparing association network algorithms for reverse engineering of large-scale gene regulatory networks: synthetic versus real data. Bioinformatics 23: 1640-1647.

74. Sachs K, Perez O, Pe'er D, Lauffenburger DA, Nolan GP (2005) Causal Protein-Signaling Networks Derived from Multiparameter Single-Cell Data. Science 308: 523-529.

75. Woolf PJ, Prudhomme W, Daheron L, Daley GQ, Lauffenburger DA (2005) Bayesian analysis of signaling networks governing embryonic stem cell fate decisions. Bioinformatics 21: 741-753.

76. Tanay A, Shamir R (2001) Computational expansion of genetic networks. Bioinformatics 17 Suppl 1: S270-278.

77. Herrgard MJ, Covert MW, Palsson BO (2003) Reconciling gene expression data with known genome-scale regulatory network structures. Genome Res 13: 2423-2434.

78. Ihmels J, Friedlander G, Bergmann S, Sarig O, Ziv Y, et al. (2002) Revealing modular organization in the yeast transcriptional network. Nat Genet 31: 370-377.

79. Zou KH, Tuncali K, Silverman SG (2003) Correlation and simple linear regression. Radiology 227: 617-622.

80. Upton GJG (1992) Fisher's exact test. J Roy Statistical Society 155: 395.

81. Huerta AM, Salgado H, Thieffry D, Collado-Vides J (1998) RegulonDB: a database on transcriptional regulation in Escherichia coli. Nucleic Acids Res 26: 55-59.

82. Morett E, Bork P (1998) Evolution of new protein function: recombinational enhancer Fis originated by horizontal gene transfer from the transcriptional regulator NtrC. FEBS Lett 433: 108-112.

83. Weinstein-Fischer D, Elgrably-Weiss M, Altuvia S (2000) Escherichia coli response to hydrogen peroxide: a role for DNA supercoiling, topoisomerase I and Fis. Mol Microbiol 35: 1413-1420.

84. Battistoni A, Pacello F, Folcarelli S, Ajello M, Donnarumma G, et al. (2000) Increased expression of periplasmic Cu,Zn superoxide dismutase enhances survival of Escherichia coli invasive strains within nonphagocytic cells. Infect Immun 68: 30-37.

85. Luke I, Butland G, Moore K, Buchanan G, Lyall V, et al. (2008) Biosynthesis of the respiratory formate dehydrogenases from Escherichia coli: characterization of the FdhE protein. Arch Microbiol 190: 685-696.

86. Nachin L, Nannmark U, Nystrom T (2005) Differential roles of the universal stress proteins of Escherichia coli in oxidative stress resistance, adhesion, and motility. J Bacteriol 187: 6265-6272.

87. Beloin C, Dorman CJ (2003) An extended role for the nucleoid structuring protein H-NS in the virulence gene regulatory cascade of Shigella flexneri. Mol Microbiol 47: 825-838.

88. Patrauchan MA, Sarkisova SA, Franklin MJ (2007) Strain-specific proteome responses of Pseudomonas aeruginosa to biofilm-associated growth and to calcium. Microbiology 153: 3838-3851.

89. Domka J, Lee J, Bansal T, Wood TK (2007) Temporal gene-expression in Escherichia coli K-12 biofilms. Environ Microbiol 9: 332-346.

90. Lee J, Jayaraman A, Wood TK (2007) Indole is an inter-species biofilm signal mediated by SdiA. BMC Microbiol 7: 42.

91. Bidaut G, Suhre K, Claverie JM, Ochs MF (2006) Determination of strongly overlapping signaling activity from microarray data. BMC Bioinformatics 7: 99.

92. Yu T, Li KC (2005) Inference of transcriptional regulatory network by two-stage constrained space factor analysis. Bioinformatics 21: 4033-4038.

93. Lee JA, Sinkovits RS, Mock D, Rab EL, Cai J, et al. (2006) Components of the antigen processing and presentation pathway revealed by gene expression microarray analysis following B cell antigen receptor (BCR) stimulation. BMC Bioinformatics 7: 237.

94. Zhu X, Hart R, Chang MS, Kim JW, Lee SY, et al. (2004) Analysis of the major patterns of B cell gene expression changes in response to short-term stimulation with 33 single ligands. J Immunol 173: 7141-7149.

95. Papin JA, Palsson BO (2004) The JAK-STAT signaling network in the human B-cell: an extreme signaling pathway analysis. Biophys J 87: 37-46.

96. Brenet F, Socci ND, Sonenberg N, Holland EC (2009) Akt phosphorylation of La regulates specific mRNA translation in glial progenitors. Oncogene 28: 128-139.

97. Al-Ejeh F, Darby JM, Pensa K, Diener KR, Hayball JD, et al. (2007) In vivo targeting of dead tumor cells in a murine tumor model using a monoclonal antibody specific for the La autoantigen. Clin Cancer Res 13: 5519s-5527s.

98. Katoh M (2007) Comparative integromics on JMJD1C gene encoding histone demethylase: conserved POU5F1 binding site elucidating mechanism of JMJD1C expression in undifferentiated ES cells and diffuse-type gastric cancer. Int J Oncol 31: 219-223.

99. Suto H, Katakai T, Sugai M, Kinashi T, Shimizu A (2009) CXCL13 production by an established lymph node stromal cell line via lymphotoxin-beta receptor

engagement involves the cooperation of multiple signaling pathways. Int Immunol 21: 467-476.

100. Go AS, Chertow GM, Fan D, McCulloch CE, Hsu CY (2004) Chronic kidney disease and the risks of death, cardiovascular events, and hospitalization. N Engl J Med 351: 1296-1305.

101. Camici M (2007) The Nephrotic Syndrome is an immunoinflammatory disorder. Med Hypotheses 68: 900-905.

102. Berthier CC, Zhang H, Schin M, Henger A, Nelson RG, et al. (2009) Enhanced expression of Janus kinase-signal transducer and activator of transcription pathway members in human diabetic nephropathy. Diabetes 58: 469-477.

103. Brosius FC, 3rd (2008) New insights into the mechanisms of fibrosis and sclerosis in diabetic nephropathy. Rev Endocr Metab Disord 9: 245-254.

104. Mates AK, Sayed AK, Foster JW (2007) Products of the Escherichia coli acid fitness island attenuate metabolite stress at extremely low pH and mediate a cell density-dependent acid resistance. J Bacteriol 189: 2759-2768.

105. Mangan MW, Lucchini S, Danino V, Croinin TO, Hinton JC, et al. (2006) The integration host factor (IHF) integrates stationary-phase and virulence gene expression in *Salmonella enterica* serovar Typhimurium. Mol Microbiol 59: 1831-1847.

106. Martin RG, Rosner JL (2004) Transcriptional and translational regulation of the marRAB multiple antibiotic resistance operon in Escherichia coli. Mol Microbiol 53: 183-191.

107. Sayed AK, Odom C, Foster JW (2007) The Escherichia coli AraC-family regulators GadX and GadW activate gadE, the central activator of glutamate-dependent acid resistance. Microbiology 153: 2584-2592.

108. Tramonti A, De Canio M, De Biase D (2008) GadX/GadW-dependent regulation of the Escherichia coli acid fitness island: transcriptional control at the gadY-gadW divergent promoters and identification of four novel 42 bp GadX/GadW-specific binding sites. Mol Microbiol 70: 965-982.

109. Brown G, Singer A, Proudfoot M, Skarina T, Kim Y, et al. (2008) Functional and structural characterization of four glutaminases from Escherichia coli and Bacillus subtilis. Biochemistry 47: 5724-5735.

110. Chuang PY, He JC (2010) JAK/STAT signaling in renal diseases. Kidney Int 78: 231-234.

111. Johnson WE, Li C, Rabinovic A (2007) Adjusting batch effects in microarray expression data using empirical Bayes methods. Biostatistics 8: 118-127.

112. McCreary V, Kartha S, Bell GI, Toback FG (1988) Sequence of a human kidney cDNA clone encoding thymosin beta 10. Biochem Biophys Res Commun 152: 862-866.

113. Ka SM, Rifai A, Chen JH, Cheng CW, Shui HA, et al. (2006) Glomerular crescent-related biomarkers in a murine model of chronic graft versus host disease. Nephrol Dial Transplant 21: 288-298.

114. Huang L, Zheng M, Zhou QM, Zhang MY, Jia WH, et al. (2011) Identification of a gene-expression signature for predicting lymph node metastasis in patients with early stage cervical carcinoma. Cancer 117: 3363-3373.

115. Boyer O, Benoit G, Gribouval O, Nevo F, Pawtowski A, et al. (2010) Mutational analysis of the PLCE1 gene in steroid resistant nephrotic syndrome. J Med Genet 47: 445-452.

116. Hinkes B, Wiggins RC, Gbadegesin R, Vlangos CN, Seelow D, et al. (2006) Positional cloning uncovers mutations in PLCE1 responsible for a nephrotic syndrome variant that may be reversible. Nat Genet 38: 1397-1405.

117. Potvin E, Sanschagrin F, Levesque RC (2008) Sigma factors in Pseudomonas aeruginosa. FEMS Microbiol Rev 32: 38-55.