GENOMIC APPLICATIONS OF STATISTICAL SIGNAL PROCESSING

A Dissertation

by

WENTAO ZHAO

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

August 2008

Major Subject: Electrical Engineering

GENOMIC APPLICATIONS OF STATISTICAL SIGNAL PROCESSING

A Dissertation

by

WENTAO ZHAO

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Approved by:

Co-Chairs of Committee,	Erchin Serpedin
	Edward R. Dougherty
Committee Members,	Andrew Chan
	Deepa Kundur
	Sing-Hoi Sze
Head of Department,	Costas Georghiades

August 2008

Major Subject: Electrical Engineering

ABSTRACT

Genomic Applications of Statistical Signal Processing. (August 2008)

Wentao Zhao,

B.S., Tsinghua University;

M.S., Tsinghua University;

M.E., Texas A&M University

Co–Chairs of Advisory Committee: Dr. Erchin Serpedin
Dr. Edward R. Dougherty

Biological phenomena in the cells can be explained in terms of the interactions among biological macro-molecules, e.g., DNAs, RNAs and proteins. These interactions can be modeled by genetic regulatory networks (GRNs). This dissertation proposes to reverse engineering the GRNs based on heterogeneous biological data sets, including time-series and time-independent gene expressions, Chromatin ImmunoPrecipatation (ChIP) data, gene sequence and motifs and other possible sources of knowledge. The objective of this research is to propose novel computational methods to catch pace with the fast evolving biological databases.

Signal processing techniques are exploited to develop computationally efficient, accurate and robust algorithms, which deal individually or collectively with various data sets. Methods of power spectral density estimation are discussed to identify genes participating in various biological processes. Information theoretic methods are applied for non-parametric inference. Bayesian methods are adopted to incorporate several sources with prior knowledge. This work aims to construct an inference system which takes into account different sources of information such that the absence of some components will not interfere with the rest of the system.

It has been verified that the proposed algorithms achieve better inference accuracy and higher computational efficiency compared with other state-of-the-art schemes, e.g. REVEAL, ARACNE, Bayesian Networks and Relevance Networks, at presence of artificial time series and steady state microarray measurements. The proposed algorithms are especially appealing when the the sample size is small. Besides, they are able to integrate multiple heterogeneous data sources, e.g. ChIP and sequence data, so that a unified GRN can be inferred. The analysis of biological literature and *in silico* experiments on real data sets for fruit fly, yeast and human have corroborated part of the inferred GRN. The research has also produced a set of potential control targets for designing gene therapy strategies.

To My Family

ACKNOWLEDGMENTS

I would like to express my gratitude to my research advisors, Dr. Erchin Serpedin and Dr. Edward R. Dougherty, for their continuous support to complete my dissertation. I would also like to thank my committee members, Dr. Andrew Chan, Dr. Deepa Kundar and Dr. Sing-Hoi Sze for their constructive advice.

The work was accomplished with the help from the warm-hearted colleagues, Dr. Yufei Xiao, Dr. Ranadip Pal, Dr. Ashish Choudhary, Dr. Xiaoning Qian, Dr. Chao Sima, Dr. Jianping Hua, Dr. Qian Xu, Dr. Ivan Ivanov, Mr. Babak Faryabi, Ms. Golnaz Vahedi and many other friends.

This work was supported by the National Cancer Institute (CA-90301) and the National Science Foundation (ECS-0355227 and CCF-0514644).

TABLE OF CONTENTS

CHAPTER Page

CHAPTER Page

CHAPTER Page

LIST OF TABLES

TABLE Page

LIST OF FIGURES

FIGURE Page

FIGURE Page

CHAPTER I

INTRODUCTION

The mystery of various living organisms has been revealed gradually by generations of biologists. The cell was discovered by Robert Hooke through a microscope and was recorded in his book *Micrographia* in 1665. Matthias Jakob Schleiden and Theodor Schwann in 1839 established the cell theory and described the cell as the structural and functional unit of life forms. In the 1860s, Gregor Mendel disclosed concepts of modern genetics when he hybridized pea plants and studied the inheritance of traits. Later in 1936, Warren Weaver coined the name of molecular biology. Since then, the life phenomena have been explored at the most fundamental levels with the participation of physicists and chemists. James Watson and Francis Crick in 1953 discovered the double helix structure of Deoxyribonucleic acid (DNA). Quickly in 1957 Crick presented the *central dogma*, which exposed the information transfer process from the hereditary material, i.e. genes on the DNA strand, to the structural and mechanical compounds, namely protein.

The birth of genomic molecular biology brought forth the explosion of interdisciplinary biotechnology. The accelerating evolvement of experimental methods was accompanied by high throughput data, which provided further insights into the operation of biological processes. Mathematical and engineering methods came to play quantitative roles in the analysis of the output data and prediction of outcomes. As a major component of the current information technology revolution, statistical signal processing techniques are playing a major role in the analysis of genomic data. In this chapter, the biological background of the research work conducted in this disserta-

The journal model is *IEEE Transactions on Automatic Control.*

tion is briefly reviewed so that the key contributions of the signal processing methods are identified. Also, the research methodology is formulated and the computational framework is introduced.

A. Biological Background of Genetic Regulatory Networks

The hereditary information of living organisms is encoded in the double helix of Deoxyribonucleic acid (DNA), which is characterized by two entwined strands composed of sequences of four nucleo-bases, namely, adenine (A), guanine (G), cytosine (C) and thymine (T). The double helix is maintained by hydrogen bonds between bases attached to the two strands in such a way that adenine on one chain is always paired with thymine on the other chain, while guanine is always paired with cytosine. Not all DNA segments bear information. Those encoding functional products are genes. The DNA is folded to form chromosomes, which are found in nucleus in eukaryotes or cytoplasm in prokaryotes. The entire genetic information on the chromosomes is referred to as genome.

The functions of living cells are achieved via proteins, which are three-dimensional polymers composed of twenty different amino acids. They catalyze biochemical reactions as enzymes, maintain cell shape as cytoskeleton and also play mechanical and signaling roles. The order of amino acids on the protein chain is determined by the corresponding gene's nucleotide sequence. This transfer of sequential information is termed as the *central dogma* of molecular biology: the DNA can be transcribed into messenger Ribonucleic acid (mRNA), which serves as the template to translate into protein.

The mechanism governing the above gene expression procedure underlies all cellular processes. Early studies have reported that gene expressions are predominantly regulated at transcription level by regulatory proteins, which receive exterior signals, relay information and serve as intracellular factors. Signals, e.g., the increased concentration of glucose, propagate along the signal transduction pathways with the involvement of enzymes. Signal-communicated transcription factors, activators and repressers influence the target gene's expression. In prokaryote, a transcription factor can bind to the promoter region of DNA, and prevents the RNA polymerase from attaching to DNA, thus forbidding transcription and acting as a repressor. On the contrary, a transcription factor can also recruit RNA polymerase and helps to change the closed DNA double helix into an open complex, and therefore it might function as an activator. In eukaryotes, a transcription factor can unwind nucleosome to make the gene accessible for transcription. Some other transcription factors can recruit histone-modifying enzymes to help the transcription machinery bind to the promoter.

Actually the regulation mechanism remains mostly unknown and extremely complicated. Later findings verified that the gene expression can also be controlled by RNA molecules, which can inhibit the expression of homologous genes. Regulation can also take place at post-transcriptional stages, e.g., through splicing and translation. Since we can view participating enzymes and RNAs as products of their associated genes, a network can be constructed for a genetic process to account for the interactions between regulatory factors and their target genes. Such a map constitutes a genetic regulatory network (GRN) and shield details of the regulation machinery. GRNs systematically explain how genes and their products cooperatively participate in molecular-biological processes and straightforwardly illustrate their logical interactions.

The effects of GRNs can be observed both in phenotype and genotype. The rapidly evolving gene technologies are providing us with various experimental methods, which are capable of measuring gene expressions at transcription and translation stages. The large amount of data produced thereafter has attracted extensive research on the reverse engineering problem, i.e., the inference of GRN. Learning GRN not only enables the possibility of understanding the function of organisms at the molecular level but it also helps to infer potential control targets for designing intelligent therapies and drugs.

B. Heterogeneous Experimental Data

In the middle of 1990s the birth of DNA microarrays equipped the industry with the capability to simultaneously measure the concentration of genome-wide mRNA expressions, which are quantifications of gene expressions and reflect gene transcription rates. There are two types of DNA microarray data: time series and time independent (or steady state). The time series data are obtained by temporally sampling the measurement process, while time independent data sets are obtained by recording the gene expressions from independent sources, e.g., different individuals, tissues, experiments, etc. Available data share three characteristics. Firstly, most data sets are of small sample size, usually not more than 50 data points. Large sample sizes are not financially affordable due to high cost of gene chips. For time course experiments, the cell cultures lose their synchronization and render data meaningless after a period of time. Secondly, many time points are missing and time course data are usually unevenly sampled. Thirdly, most data sets are customarily corrupted by experimental noise and the produced uncertainty should be addressed in a stochastic framework. Formidable costs, ethical concerns and implementation issues obstruct the collection of large time series data sets. Currently, about 70% of the data sets are time independent [1]. The microarray experiments can also be designed and conducted in controlled conditions. One popular technique is RNA inference, which can shut-off a specific gene using its corresponding double strand RNA (dsRNA).

The advent of in vivo Chromatin Immuno-Precipitation (ChIP) assays has enabled to test whether a protein acting as a transcription factor binds to a specific DNA segment. Hence, ChIP assays serve as a promising mechanism to examine the regulatory relationships. In ChIP experiments, the protein is immobilized on the chromatin, and then the chromatin is broken into DNA fragments. The DNA-protein complexes are immunoprecipitated by using antibodies corresponding to the tested protein. Afterwards the DNA bound by the protein in question can be isolated and identified by using a cDNA microarray chip. The whole process is also called a ChIP-chip experiment, and inherits several disadvantages. The protein to be tested has to possess a specific antibody, which might not be synthesized, discovered or known. In addition, the transcriptional regulation is a complex process that is expressed in several different aspects. The binding of the transcription factor to the promoter region of the target gene is the most pristine mode. Especially for eukaryotic organisms, some regulatory bindings take place in a region far away from the regulated gene. This fact makes the binding information questionable for determining the regulation relationships. Furthermore, the experimental results are represented by p-values and the determination of the binding relationship is achieved through threshold comparison. However, the selection of the p-value threshold introduces a dilemma. A high threshold not only identifies the most probable binding relationships but also might miss many true relationships with lower p-values, while a low threshold infers more relationships, among which more might be false alarms. A good trade-off is not easy to make. Besides, the cost has to be taken into consideration. Generally ChIP-chip experiments are very expensive and testing thousands of proteins is not affordable.

Multiple genome sequencing projects have been accomplished or are currently under way for such organisms as E. coli, yeast, fruit fly, bee, mouse, cattle and human. The genome data are stored in databases in terms of a sequence of letters, which are selected from the alphabet A, T, C, G corresponding to the four nucleotides. The sequence data might yield information about the binding motifs, i.e., the sequence pattern on the target genes regulatory region. This data can be exploited to further refine our knowledge about the regulation at molecular level.

Biological experiments also produce various other sources of information which may be of interest. The protein experiments using mass spectrometry or protein microarrays provide insights about the protein-protein interactions, which somehow help to explain co-regulations. The well-established knowledge of some biological processes in certain organisms might not only serve as a prior knowledge but might also be used as a benchmark in evaluating the performance of the proposed schemes. A cross-species comparison is also highly desirable since similar regulation mechanisms are expected to be conserved along the family tree of evolution. If a gene is conserved in both humans and mice, then the knowledge of the genes pathway in the mouse will be an excellent reference for the study of human genetic diseases.

A variety of data and knowledge sources are generally available through public databases. For example, the yeast database at Stanford University (`http://www.yeastgenome.org/`) provides up-to-date microarray and sequence data sets. At Texas A&M University, genome data for honey bee and bovine can be accessed through `http://racerx00.tamu.edu/`. Other sources of information are coming from our collaborators: Translational Genomic Research Institute (TGEN) at Phoenix and M.D. Anderson Cancer Center at Houston.

C. Mathematical Models of Genetic Regulatory Networks

Thus far, multiple models have been proposed for capturing the gene interactions. Boolean networks [2] model regulatory relations in terms of combinatorial logic circuits, while probabilistic Boolean networks (PBNs), e.g. [3] and [4], are composed of a finite number of constituent Boolean networks, each of which corresponding to a contextual condition determined by the variables outside the model. The immediate extension of PBNs to any finite quantization is represented by the class of Bayesian networks, e.g. [5] and [6], which model the non-temporal probabilistic dependency relations among genes. The dynamic Bayesian networks (DBNs), e.g. [7] and [8], extend the class of Bayesian networks to the time domain by modeling the temporal stochastic relationships among genes. In this regard, Relevance networks [9] are undirected graphs that account for significant statistical relationships among genes.

Specifically, Bayesian networks present a long history for modeling the causal relationships [10]. It constrains the structural model to be an acyclic graph. Unfortunately such a constraint does not reflect always the true characteristics of gene regulatory networks since feedbacks or loops are common motifs in genetic regulation. Several fundamental relationships have been established recently between the class of PBNs and the class of DBNs [11]. However, with the exception of some one-to-many mappings between the two classes, a complete understanding of the relationships between the two classes is not yet available.

Herein, we will be working towards establishing a unified GRN model which assumes continuous values for each variable (gene), presence of cycles and oriented edges. In addition, the specific structure of GRN will be refined based on the type of available data.

D. Current Inference Approaches

The existing inference schemes can be coarsely categorized based on their different features with respect to the type of modeling framework and data source. Until recently, microarray gene expressions served as the main data source. However, recent developments suggested the acute need for data fusion methods that account for heterogeneous data sources. Next, a short overview of the most representative GRN inference algorithms will be presented.

Kim et al. [12] proposed the concept of coefficient of determination (CoD) to identify the predictor set of the target gene based on the gene expression profiles. The method was validated by simulations on a set of genes undergoing genotoxic stress. Zhou etal. [13] exploited the reversible jump Markov Chain Monte Carlo algorithm to determine the model order and parameters. Pal et al. [14] proposed two schemes for constructing Boolean networks based on the concept of attractor states. The inferred Boolean networks were then employed to construct probabilistic Boolean networks (PBN).

Butte and Kohane designed a relevance network (RN) by exploiting the mutual information to represent the interaction significance between two genes [9]. In [9], two genes were considered to be relevant if their mutual information assumed a larger value than a pre-specified threshold and an undirected edge was laid between them. The proposed scheme was run on the Yeast data set and the inferred networks were examined by comparing them with experimental results reported in the biological literature. It was shown that genes located in the same relevance network shared similar biological functions or participated in the same biological process.

Chow-Liu algorithm [15] approached the inference problem by finding the maximum spanning tree in which the edge weights stood for the mutual information between the expression profiles of the two genes. However, Chow-Liu algorithm loses validity if the underlying model is a cyclic graph. In addition, when the graph is densely connected, this scheme might miss too many edges.

Margolin et al. [16] proposed the Algorithm for the Reconstruction of Accurate Cellular Networks (ARACNE) based on the information provided by independent microarray samples. ARACNE inferred the direct connectivity among genes using the mutual information and data processing inequality (DPI). ARACNE assumes first a fully connected graph and a pre-defined mutual information threshold. Whenever the mutual information between two genes X and Y, i.e., $I(X;Y)$, is less than the pre-specified threshold, it disconnects the two genes. Next, if in the preliminary graph there exists another gene Z so that $I(X;Y) < \min(I(X;Z), I(Y;Z))$, then ARACNE will disconnect X and Y. ARACNE relies on the critical assumption that the gene interactions could be described by Markov chains. ARACNE was run on the synthetic networks generated by Mendes in 2003. The performance was evaluated favorably in terms of precision and specificity. ARACNE was also simulated in the presence of the human B-cell data. The inferred B-cell network was compared with those previously identified through biochemical methods. The published targets of hub gene c-MYC were found to be mostly c-MYCs direct neighbors in the reconstructed network.

Liang et al. proposed the REVerse Engineering Algorithm (REVEAL) to reconstruct Boolean networks from time series microarray data [17]. REVEAL compared the mutual information, defined between the possible predictor set and the target gene, with the entropy of the target gene. When these two quantities matched, the predictor set was determined. To evaluate the performance of REVEAL, a set of synthetic Boolean networks were created and the state transitions were generated without noise. The false alarm error did not occur due to the absence of noise, and only miss errors were illustrated with respect to the sample size.

Friedman et al. employed Bayesian Networks to model genetic networks [5]. Bayesian Networks are directed acyclic graphs hence they lose efficacy in the presence of feedbacks which are common motifs in the biological world. Heuristic search was exploited in finding the best-fit network. The in silico experiment was conducted on the Spellman's Yeast data set [18]. The inherent temporal information of the data set was ignored. The recovered network was compared with the network inferred from a randomized data set in terms of the distribution of confidence estimates for Markov chain parameters and order features. It was shown that the proposed scheme recovered different patterns for experimental Yeast data, measured by Spellman et al. [18], and randomized data, respectively. This discrepancy was attributed to the genetic regulations and the found pattern was treated as true positive. Chen et al. [19] improved and simplified the learning of Bayesian networks by exploiting mutual information and identifying the ordering of nodes. The proposed scheme was simulated on Bayesian networks, i.e., the ASIA network [20] and the ALARM network [21]. The false negatives, false positives and false orientation errors were tabulated. The algorithm was also run on the yeast data [18] and inferred genetic networks were discussed. Pe'er et al. [22] improved the inference of Bayesian networks by enforcing biologically motivated constraints and reducing the search space. The proposed scheme, referred to as MinReg, was tested on synthetic data from a known network. The two types of error, false alarms and misses, were used to corroborate the algorithm performance. The scheme was further run on yeast and mouse data sets.

Murphy and Mia extended the Bayesian network modeling framework to dynamic Bayesian networks (DBNs) so that the genetic model structure allowed directed cycles and exploitation of temporal data [23]. Zou and Conzen [24] assumed the transcriptional time lag to be a variable and the regulator genes are allowed to change their expressions prior to their targets. Also, the regulators were constrained to the discovered transcription factors. Therefore, the search space was largely reduced and the computational efficiency was greatly improved. The approach was simulated on the yeast data reported by Chou et al. [25]. The inferred network was compared with the results reported in the cell cycle regulation literature, and specificity and mis-orientation errors were tabulated. Instead of quantizing the gene expressions, Kim et al. [26] proposed a continuous DBN model. The proposed algorithm was simulated on Spellman et al.'s yeast data set. The recovered network was also compared with the cell cycle pathway reported in the KEGG database [27] and the metabolic pathway reported by DeRisi et al. [28].

The inference of genetic networks is currently moving toward the integration of heterogeneous data sources. Bar-Joseph et al. [29] proposed the genetic regulatory modules (GRAM) algorithm to combine the gene expression data with transcription factor binding location data. GRAM clustered genes into modules with similar expressions. Alternatively, Bernard [30] treated the binding location data as the prior knowledge for the inference of dynamic Bayesian networks. The proposed algorithm was simulated on stochastic Boolean networks. The Hamming distance between the inferred and synthetic networks were plotted. Based on the Saccharomyces Genome Database `http://www.yeastgenome.org`, a "gold standard" network was constructed to represent the true scenario of the cell cycle. The scheme was then simulated on Spellman et al.'s time series data [18] and Lee et al.'s binding location data [31]. Applications based on the integration of other data sources include protein-protein interaction data [32] and sequence data [33]. Data fusion has also been proposed for other inference purposes, e.g., discovery of regulatory motifs through the combination of gene expression and DNA sequence knowledge [34],[35], and protein function prediction [36].

The objective of this dissertation is to design a computational framework that excels in inference accuracy, computing complexity, and configuration flexibility.

E. Proposed Methodology

1. Graphical Models

Graphical models have been exploited to represent the structure of genetic networks. Generally, the network structure can be represented by a graph $G(V;E)$, where V denotes the set of vertices (genes) and E stands for the set of edges (regulation relationships). Since proteins and RNAs regulating the target gene are products of their associated genes, alternatively we view these Protein-DNA and RNA-DNA interactions as gene-gene interactions.

If gene X regulates gene Y, graphically such a relation is represented in terms of an oriented edge $X \rightarrow Y$, where X is a parent or predecessor of Y and Y is considered a child or successor of X. All genes that present incidence edges with gene X represent the set of parental genes of X, and are compactly denoted in terms of the notation Π_X. For instance, if gene X is regulated cooperatively by genes Y and Z, then $\Pi_X = \{Y, Z\}$. Similarly, the notation Ξ_X is used to represent the set of successor genes which are regulated by gene X. If gene X regulates simultaneously only the genes Y and Z, then $\Xi_X = \{Y, Z\}$.

If two genes X and Y interact with each other but the regulation orientation can not be determined, an undirected edge is laid between the two genes as $X - Y$. In many models a direct connectivity between two genes X and Y in the graph stands for a vague biological relationship, which might represent a broad class of relationships such as both genes X and Y are regulating or regulated by a common gene, X directly regulates Y, or X indirectly regulates Y by means of several intermediate genes. Although our inference is also based on statistical relationships, we are aiming to capture the direct regulation relationships as accurately as possible so that the real genetic regulation machinery is discovered and the above mentioned distinct relationships are differentiated.

Associated with a specific gene X is the regulation function $f_X(\Pi_X)$, which denotes the expression value for gene X determined by the values of the genes in the set of predecessors Π_X. For simplicity, the shorthand notation f_X will be used since Π_X is uniquely determined in the biological world. The function f_X might be a simple logic function as proposed by Kauffman [2]. It could also be chosen from a set of candidate functions as considered in the probabilistic Boolean network (PBN) framework [3]. Alternatively, f_X can be specified in the form of a contingency table if X assumes discrete values, e.g. [5], [37] and [38] or in the form of a probability distribution function if X is a continuous variable. Linear and non-linear differential equations are also accepted for modeling the kinetics of molecular level reactions, which in general assume much intense computations, e.g. [39]–[42]. We assume that all the parameters are recorded in the parameter set Θ, as opposed to the graph structure notation G.

A sequence of consecutive oriented edges constitutes a directed path. If there is no directed path which starts and ends at the same vertex, in other words the graph contains no loops, the graph is called a directed acyclic graph (DAG). DAGs lie at the basis of Bayesian networks, which are commonly employed to model causal relationships [10]. Bayesian networks were not chosen in our study due to several reasons. Firstly, there exist many Markov equivalent Bayesian networks which fit the observational data equally well, share the same connectivity structure but differ in the connectivity orientations. Secondly, Bayesian networks do not allow loops, which are common in many real biological processes. We will allow the presence of cycles and also accommodate undirected graphs. The inference of the direct connectivity has to differentiate between X-Z-Y and X-Y. In the former case gene X interacts with gene Y through an intermediate gene Z, while in the latter case gene X directly interacts with gene Y.

2. Computational Framework

The goal of the proposed computation framework is to preserve with high accuracy only the direct connectivity among the participating genes, maintain a low complexity in network inference, and when a false-alarm connectivity is produced between two genes, the two falsely connected genes are located closely enough in the actual network. The computation procedure does not need to be changed much for different combination of knowledge and data owing to the structured computing flowchart.

Fig. 1 illustrates the general computational procedure by using a combination of ChIP-chip and microarray steady state data. The two rows of operations correspond to two types of data. The left segment is conducted by biologists, who present the data in terms of spreadsheets. For each type of data, prior knowledge is integrated to preprocess the data and the proposed inference schemes are then applied. Finally, the genetic regulatory network is inferred. The integration of data is achieved through Bayesian methods along with information theoretic approaches. Parameterized and non-parametric approaches will both be tested and compared in terms of performance and efficiency. When a new data source is available, we hope that only one extra row of data and associated operations will be needed so that the whole framework remains unchanged.

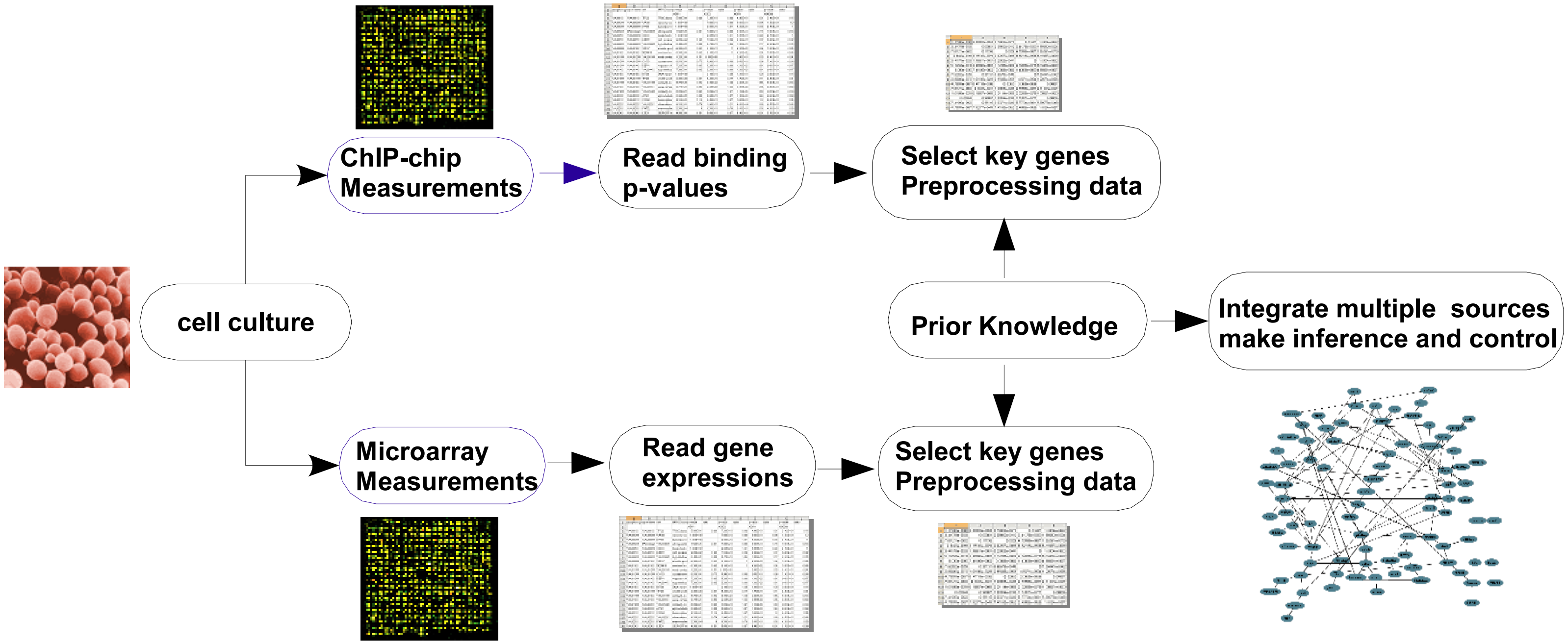

ChIP-chip Measurements
Read binding p-values
Select key genes Preprocessing data
cell culture
Prior Knowledge
Integrate multiple sources make inference and control
Microarray Measurements
Read gene expressions
Select key genes Preprocessing data

Fig. 1. Computation flowchart for combining two data sources

3. Nonparametric and Bayesian Methods

Nonparametric methods do no impose specific assumptions on the form and range of the stochastic variables. Therefore, they are attractive when not much knowledge is available about the underlying biological processes. Information theoretical quantities, such as entropy and mutual information [43], are employed to measure the significance of gene interactions, and the Minimum Description Length principle [44] is exploited to rule out the intermediary interactions. Information theoretic quantities will be constructed based on multivariate entropy estimates. In turn the entropy estimation depends on the mass or density estimators. Recent progress in the area of estimating information theoretic quantities has led to a number of alternatives for estimating the entropy, e.g. [45]–[47]. Note that usually it is the rank of the mutual information that accounts for the connectivity. Therefore, the desired estimator has to exhibit small variance and acceptable bias.

The Bayesian methodology is also proposed to jointly analyze the available data sets and to establish a confidence measure for gene interactions. The Bayesian schemes proposed in this dissertation possess four key features which make them different from the existing algorithms. First, most of the current schemes recover a unique genetic network represented by a graph which best fits the observed data in a certain metric, while the proposed approaches determine the posterior probabilities for all gene-pair interactions and avoid to make a dichotomous decision that classifies each gene interaction as being either connected or disconnected. The proposed approaches can be easily transformed into dichotomous schemes by only preserving the highly probable gene interactions. Second, the proposed approaches will assume continuous-valued variables and treat discrete values as special cases. This prevents the information loss incurred by data quantization and represents an advantage compared with the discrete-valued networks. Third, the proposed connectivity score is oriented and has a very clear meaning, in the sense of posterior probabilities, while the existing scores are vague and lack orientation information. Fourth, in the proposed approaches the system kinetics is assumed to be nonlinear, while linear models are commonly utilized for the purpose of simplification. Besides, the proposed schemes establish a general framework whose components can be customized to fit the nature of the underlying biological system.

4. Performance Evaluation and Method Validation

There are two types of inference errors. The type 1 errors are false positives (FP) and are also called false alarms. If the inference algorithm determines an interaction for two vertices X and Y in the inferred graph, denoted as $X \rightarrow Y \in \hat{E}$, but there is no such edge in the synthetic graph, i.e., $X \rightarrow Y \notin E$, then an FP is produced. The number of FPs, represented by N_{FP}, can be counted as follows:

$$N_{FP} = \sum_{\forall X,Y} \left((X \rightarrow Y \in \hat{E}) \bigcap (X \rightarrow Y \notin E) \right),$$

where $\bigcap$ stands for the logic *and* operator. The type 2 errors are false negatives (FN) and also named misses. If the inference does not discover the connectivity $X \rightarrow Y$ which resides in the synthetic network, an FN is generated. The number of FNs, depicted by N_{FN}, is given by:

$$N_{FN} = \sum_{\forall X,Y} \left((X \rightarrow Y \in E) \bigcap (X \rightarrow Y \notin \hat{E}) \right).$$

Correct inference can also be divided into two categories. If $X \rightarrow Y \in \hat{E}$ and $X \rightarrow Y \in E$, the correctness is defined as a true positive (TP). Its summation, annotated by N_{TP}, is:

$$N_{TP} = \sum_{\forall X,Y} \left((X \to Y \in \hat{E}) \bigcap (X \to Y \in E) \right).$$

On the other hand, if $X \to Y \notin \hat{E}$ and $X \to Y \notin E$, such correctness is called a true negative (TN). The number of TNs, represented by N_{TN}, is defined as follows:

$$N_{TN} = \sum_{\forall X,Y} \left((X \to Y \notin \hat{E}) \bigcap (X \to Y \notin E) \right).$$

Different performance metrics are proposed in the literature. The three most popular metrics are considered here. The first metric, referred to as the Hamming distance, is the summation of all the inference errors and is given by

$$Hamming\ distance = N_{FP} + N_{FN}.$$

The Hamming distance is widely accepted as a good measure of the distance between two graphs.

The second metric is called the sensitivity, and is defined as:

$$Sensitivity = \frac{N_{TP}}{N_{TP} + N_{FN}}.$$

The sensitivity describes the inference algorithm's ability to identify the regulation relationships among genes. The third metric is called the specificity, and it assumes the form:

$$Specificity = \frac{N_{TN}}{N_{TN} + N_{FP}}.$$

The specificity represents the inference algorithm's capability to avoid falsely connecting two unrelated genes.

The error rates are usually estimated through simulation on artificial networks.

In the first step a set of network structures are randomly created and their parameters are set to conform to the assumptions governing the system kinetics. Then the synthesized networks are sampled in transient states or in steady states. Some experiments, e.g., RNAi, can be emulated by forcing some specific nodes to assume fixed values. Later the proposed schemes are applied on artificial data sets and the inferred networks are compared with the original networks so that both the inference errors and corrections can be identified and counted. Various platforms have been set up to provide benchmarks in evaluating the inference performance, e.g. [48] and [49]. We will use these third party softwares to prove the consistent superior performance of the proposed schemes.

The established networks, e.g., [31], [50] and [51], which are verified through biological experiments, can serve as benchmarks. The public databases, e.g., `http://www.pubmed.org` and TRANSFAC, represent excellent references. By exploiting the real-world data sets, the proposed methodology is desired to not only confirm the biologists results and discoveries, but also provide a systematic view of the gene interactions and potential control targets.

F. Organization of the Dissertation

The following chapters are organized as described below.

Chapter II discusses the identification of periodically expressed genes as an example to constrain the research target within a specific cellular process. The power spectral density estimation methods are compared in the case of non-uniformly sampled data. The performance is evaluated via a combination of experimental knowledge. A list of genes for Drosophila melanogaster are proposed to be cyclicly expressed.

Chapter III recognizes the challenge of genetic network inference in the presence of time independent microarray measurements. Information theoretic quantities are exploited and their estimation methods are discussed. Two algorithms are proposed and they are compared with other state-of-the-art schemes based on third-party artificial genetic networks.The proposed algorithms are also applied on realistic biological measurements, such as the cutaneous melanoma data set, and biological meaningful results are inferred.

Chapter IV addresses the problem of inferring genetic regulatory networks from time series gene-expression profiles. Based on the Minimum Description Length (MDL) principle, it proposes a network inference algorithm to recover not only the direct gene connectivity but also the regulating orientations. Simulation results show that the algorithm achieves good performance in the case of synthetic networks and excels in efficiency, accuracy, robustness and scalability. Given a time series data set for Drosophila melanogaster, the paper proposes a genetic regulatory network involved in Drosophila's muscle development.

Chapter V proposes a novel approach for reconstruction of genetic regulatory networks in light of heterogeneous data sets, particularly measurements from DNA microarrays and chromatin immunoprecipitation (ChIP) assays. Built within the framework of Bayesian statistics and computational Monte Carlo techniques, the proposed approach presents the posterior probabilities between interacting genes. A genetic regulatory network for Saccharomyces cerevisiae is inferred based on published real data sets and biological meaningful results are discussed.

Chapter VI extends the current work with three other applications: applying reversible jump Markov Chain Monte Carlo to incorporate sequence information, identifying cell cycle genes based on prior experimental knowledge, and clustering gene expressions in frequency domain.

Chapter VII summarizes the dissertation and proposes potential future research targets.

G. Main Contributions

The main contributions are summarized as follows:

- Established the Bayesian framework to combine heterogeneous data sources for the inference of genetic regulatory networks (GRN).
- Designed the GRN inference schemes based on information theoretic quantities for time independent microarray measurements.
- Developed the GRN inference schemes based on minimum description length principle (MDL) for time course microarray measurements.
- Evaluated applicability and efficiency of the power spectral density methods for non-uniform biological observations.
- Presented the scheme to identify genes involved in specific biological processes, particularly cell cycle.
- Proposed control targets and summarized network features for the inferred GRNs based on real data sets.

CHAPTER II

IDENTIFICATION OF PERIODICALLY EXPRESSED GENES *

A. Problem Overview

Multiple genome projects have been accomplished. These include the human genome, which consists of approximately 25,000 genes, the fruit fly genome, which is composed of around 14,000 protein-coding genes, and yeast genome, which contains about 6,000 genes. For an organism, all its genes in the genome cooperate systematically to function as a living body. At the molecular level, the research has to be confined to some relatively independent cellular processes, such as metabolism, cell cycle and response to stimulus. The regulation mechanisms behind these processes involve tens to hundreds of key genes, which are greatly reduced to subsets of all the genes located in the genome. The underlying genetic networks are therefore possible to be computationally recovered from the gene expression observations based on the current computing resources and methods. Fortunately the fast advancing signal processing literature has provided various methods to identify genes participating in specific biological processes, such as cell cycle and circadian rhythm, which control the accurate timing of biological cycles.

Particularly, the eukaryotic cell cycle is an echelon of molecular-level events that lead to cell division into two daughter cells. The wrongly regulated cell cycle leads to tumor formation. Besides, the cells expose their DNA during division, hence allowing

*Part of the data reported in this chapter is reprinted from "Detecting Periodic Genes from Irregularly Sampled Gene Expressions: A Comparison Study," by W. Zhao, K. Agyepong, E. Serpedin and E. R. Dougherty, 2008, *EURASIP Journal on Bioinformatics and System Biology*, Open Access 2008 by Hindawi Publishing Corporation.

themselves controllable via genetic therapy. Therefore, the cell cycle has been a hot topic for cancer research. At the transcription level, the events of the cell division can be quantitatively observed by measuring the concentration of messenger RNA (mRNA). To achieve this goal, in the microarray experiments high-throughput gene chips are exploited to measure genome-wide gene expressions sequentially at discrete time points.

Extensive genome-wide time course microarray experiments have been conducted on organisms such as *Saccharomyces cerevisiae* (budding yeast) [18], human Hela [52], and *Drosophila melanogaster* (fruit fly) [53]. Budding yeast in [18] has served as the predominant data source for various statistical methods in search of periodically expressed genes, mainly due to its pioneering publication and relatively larger sample size compared with its peers. By assuming the signal in the cell cycle to be a simple sinusoid, Spellman et al. [18] and Whitfield et al. [52] performed a Fourier transformation on the data sampled with different synchronization methods, while Giurcaneanu [54] explored the stochastic complexity of the detection mechanism of periodically expressed genes by means of generalized Gaussian distributions. Ahdesmaki et al. [55] implemented a robust periodicity testing procedure also based on the non-Gaussian noise assumption. Alternatively, Luan and Li [56] employed guide genes and constructed cubic B-spline based periodic functions for modeling, while Lu et al. [57] employed up to third harmonics to fit the data and proposed a periodic normal mixture model. Power spectral density estimation schemes have also been employed. Wichert et al. [58] applied the traditional periodogram on various data sets. Jakobsson et al. [59] compared Capon and robust Capon methods in terms of their ability to identify a predetermined frequency using evenly sampled data sets, under the assumption of a known period. Lichtenberg et al. [60] compared [18], [56] and [57] while proposing a new score by combining the periodicity and regulation magnitude. The majority of these works dealt with evenly sampled data. When missing data points were present, either the vacancies were filled by interpolation in time domain, or the genes were discarded if there were more than 30% data samples missing.

Biological experiments generally output unequally spaced measurements. The major reasons are experimental constraints and event-driven observation. The rate of measurement is directly proportional to the occurrence of events. Therefore, an analysis based on unevenly sampled data is practically desired, although technically it is more challenging. While providing modern spectral estimation methods for stationary processes with complete and evenly sampled data [61], the signal processing literature has witnessed an increased interest in analyzing unevenly sampled data sets, especially in astronomy, in the last decades. The harmonics exploited in discrete Fourier transform (DFT) are no longer orthogonal for uneven sampling. However, Lomb [62] and Scargle [63] demonstrated that a phase shift suffices to make the sine and cosine terms orthogonal. The Lomb-Scargle scheme has been exploited in analyzing the budding yeast data set by Glynn et al. [64]. Schwarzenberg-Czerny [65] employed one way analysis of variance (AoV) and formulated an AoV periodogram as a method to detect sharp periodicities. However, it relies on an infeasible biological assumption, i.e., the observation duration covers many cycles. Along this line of research, Ahdesmaki [66] proposed to use robust regression techniques, while Stoica [67] updated the traditional Capon method to cope with the irregularly sampled data. Wang et al. [68] reported a novel technique, referred to as the missing-data amplitude and phase estimation (MAPES) approach, which estimates the missing data and spectra iteratively through the Expectation Maximization (EM) algorithm. In general, Capon and MAPES methods possess a better spectral resolution than Lomb-Scargle periodogram. In this chapter, we analyze the performance of three of the most representative spectral estimation methods: Lomb-Scargle periodogram, Capon method, and the MAPES technique in the presence of missing samples and irregularly spaced samples. The following questions are to be answered in this study: do technically more sophisticated schemes, such as MAPES, achieve a better performance on real biological data sets than simpler schemes, and is the sacrifice in efficiency by these advanced methods justifiable?

B. Methods for Periodicity Identification

In this section the Lomb-Scargle periodogram, Capon method and MAPES approach are introduced and compared in terms of their features and implementation complexity.

1. Lomb-Scargle Periodogram

The deployment of Fourier transform and traditional periodogram relies on evenly sampled data, which are projected on orthogonal sine and cosine harmonics. The uneven sampling ruins this orthogonality. Hence, the Parseval's theorem fails, and there exists a power discrepancy between the time and frequency domains. When analyzing astronomical data, which in general are collected at uncontrollable observation times, Lomb [62] found that a phase-shift of the sine and cosine functions would restore the orthogonality among harmonics. Scargle [63] complemented the Lomb's periodogram by exploiting its distribution. Since then the established Lomb-Scargle periodogram has been exploited in numerous fields and applications, including bioinformatics and genomics (see e.g., Glynn [64]).

Given N time-series observations $(t_l, y_l), l = 0, \dots, N-1$, where t stands for the time tag and y denotes the sampled expression of a specific gene, the normalized

Lomb-Scargle periodogram for that gene expression at angular frequency ω is

$$\Phi_{LS}(\omega)=\frac{1}{2\hat{\sigma}^2}\left(\frac{\left(\sum_{l=0}^{N-1}[y_l-\bar{y}]cos[\omega(t_l-\tau)]\right)^2}{\sum_{l=0}^{N-1}cos^2[\omega(t_l-\tau)]}+\frac{\left(\sum_{l=0}^{N-1}[y_l-\bar{y}]sin[\omega(t_l-\tau)]\right)^2}{\sum_{l=0}^{N-1}sin^2[\omega(t_l-\tau)]}\right), \tag{2.1}$$

where $\bar{y}$ and $\hat{\sigma}^2$ stand for the mean and variance of the sampled data, respectively, and τ is defined as:

$$\tau=\frac{1}{2\omega}atan\left(\frac{\sum_{l=0}^{N-1}sin(2\omega t_l)}{\sum_{l=0}^{N-1}cos(2\omega t_l)}\right). \tag{2.2}$$

For evenly sampled data, the sampling interval Δ can be expressed as

$$\Delta=t_{l+1}-t_l=\frac{t_{N-1}-t_0}{N-1}, \qquad l=0,\ldots,N-2. \tag{2.3}$$

The highest frequency, namely the Nyquist frequency, is $1/(2\Delta)$. Beyond this limit, the computed spectra repeat. For unevenly sampled data, a straightforward way to introduce the Nyquist frequency is by keeping the definition as in the evenly sampled case, i.e., using the averaged sampling interval defined in the second equality of Equation (2.3), as is employed in Glynn's work [64]. Actually, Eyer in [69] proved that the highest frequency is much larger than $1/(2\Delta)$. Let δ be the greatest common divisor (gcd) for all intervals t_k-t_l $(k\neq l)$, then the highest frequency that should be searched is given by

$$f_{max}=\frac{\omega_{max}}{2\pi}=\frac{1}{2\delta}. \tag{2.4}$$

The number of probing frequencies is denoted by

$$\tilde{N}=\frac{t_{N-1}-t_0}{\delta}+1, \tag{2.5}$$

and the frequency grid can be defined in terms of the following equation

$$\omega_l \delta = \frac{2\pi}{\tilde{N}} l, \quad l = 0, \ldots, \tilde{N} - 1. \tag{2.6}$$

Notice further that the spectra on the front and rear halves of the frequency grid are symmetric since the microarray experiments output real values.

Lomb-Scargle periodogram represents an efficient solution in estimating the spectra of unevenly sampled data. Our simulation results verify its superior performance for biological data with small sample size and various unevenly sampled patterns.

2. Capon Method

Capon method represents a modern power spectral estimation technique that yields better spectral resolution compared with traditional periodogram [61]. The original Capon method tries to design a filter-bank by taking properties of its data into account. Assuming N observations are equally spaced with a sampling interval Δ, at a frequency ω, the Capon filter is designed so that the power of the filter's output is minimized while the frequency ω is passed without distortion. Solving this optimization problem provides the spectrum estimate at frequency ω as

$$\Phi_C(\omega) = \frac{1}{\mathbf{a}^H(\omega\Delta)\mathbf{R}^{-1}\mathbf{a}(\omega\Delta)}, \tag{2.7}$$

where the $\mathbf{R}$ stands for the data covariance matrix with a dimension N_0, which is also the bandwidth of the Capon filter. The ancillary vector is defined as follows

$$\mathbf{a}(\omega) = \left(1 \; e^{j\omega} \cdots e^{j\omega(N_0-1)}\right)^T. \tag{2.8}$$

Note that we have not included in this spectrum estimate a scaling factor. However, the absence of this scaling factor does not affect periodicity analysis for the genes. Therefore, we neglect this scaling factor. The bandwidth parameter N_0 can not exceed

$\lfloor (N-1)/2 \rfloor$ to guarantee an inverse $\mathbf{R}^{-1}$. The larger the N_0, the better the resolution of the obtained spectra.

Recently, the Capon method has been updated to cope with the presence of irregular samples [67]. The same frequency grid denoted in Equation (2.6) is employed. The sampling interval Δ has to be changed to δ, the greatest common divisor between any two sampling times. In order to take advantage of the best resolution, N_0 is set to be equal to $\lfloor (\tilde{N}-1)/2 \rfloor$, where $\tilde{N}$ is defined in Equation (2.5). In our simulation, an estimate of the autocorrelation matrix $\hat{\mathbf{R}}$ can be obtained from the Lomb-Scargle periodogram. It can be represented by

$$\hat{\mathbf{R}} = \frac{1}{\tilde{N}\delta} \sum_{l=0}^{\tilde{N}-1} \mathbf{a}(\omega_l \delta) \mathbf{a}^H(\omega_l \delta) \Phi_{LS}(\omega_l). \tag{2.9}$$

The Capon method is slightly more computationally complex than Lomb-Scargle periodogram, and it usually achieves a better performance in terms of resolution provided there are sufficient samples. However, for highly corrupted biological data with small sample size, this is not true.

3. MAPES Method

Regular sampling can be treated as a case of missing data as long as the sampling time tags share a greatest common divisor. This constraint is satisfied in most biological experiments and published data sets. The missing-data amplitude and phase estimation (MAPES) method, proposed in [68], is a non-parametric spectral estimation approach. It is robust to error modeling and it deals with arbitrary data-missing patterns as opposed to gapped or periodically gapped data, and achieves a better spectral resolution in the sense of resolving closely spaced spectral lines. However, the exploitation of the expectation maximization (EM) algorithm sacrifices its computational efficiency.

The data, $y_l, l = 0, \ldots, \tilde{N}$, are assumed to be sampled uniformly, however, only N data points are available and there are $\tilde{N} - N$ missing data points. Noticeably $\tilde{N}$ still conforms to the definition in Equation (2.5). The gene expression signal with frequency ω can be modeled as

$$y_l = \alpha(\omega)e^{j\omega l} + \varepsilon_l(\omega), \quad l = 0, \ldots, \tilde{N} - 1, \quad \omega \in [0, 2\pi], \tag{2.10}$$

where $\alpha(\omega)$ represents the complex amplitude of the sinusoidal component and $\varepsilon_l(w)$ denotes the residual term. The probing frequencies still follow Equation (2.6). Employing the EM algorithm, MAPES tries to iteratively assess the missing data, and meanwhile to update the estimation of spectra and error.

The data vector $\mathbf{y} = (y_0, \cdots, y_{\tilde{N}-1})^T$ can be partitioned into L overlapping subvectors, each with dimension $M \times 1$, and $L = \tilde{N} - M + 1$. These subvectors constitute the enhanced data vector $\tilde{\mathbf{y}}$ $(LM \times 1)$, which assumes the following expression

$$\tilde{\mathbf{y}} = \begin{pmatrix} \tilde{\mathbf{y}}_0 \\ \vdots \\ \tilde{\mathbf{y}}_{L-1} \end{pmatrix} = \mathbf{U}\boldsymbol{\gamma} + \mathbf{V}\boldsymbol{\mu}, \tag{2.11}$$

where $\boldsymbol{\gamma}$ $(N \times 1)$ and $\boldsymbol{\mu}$ $((\tilde{N} - N) \times 1)$ represent the available and missing data, respectively, and $\mathbf{U}$ $(LM \times N)$ and $\mathbf{V}$ $(LM \times (\tilde{N} - N))$ denote their selection matrices, respectively. Alternatively, given $\mathbf{U}, \mathbf{V}$ and $\tilde{\mathbf{y}}$, the data vectors $\boldsymbol{\gamma}$, $\boldsymbol{\mu}$ can be computed in the least-squares (LS) sense as follows

$$\boldsymbol{\gamma} = (\mathbf{U}^T\mathbf{U})^{-1}\mathbf{U}^T\tilde{\mathbf{y}} = \tilde{\mathbf{U}}^\dagger\tilde{\mathbf{y}}, \quad \text{where} \quad \tilde{\mathbf{U}}^\dagger = (\mathbf{U}^T\mathbf{U})^{-1}\mathbf{U}^T, \tag{2.12}$$

$$\boldsymbol{\mu} = (\mathbf{V}^T\mathbf{V})^{-1}\mathbf{V}^T\tilde{\mathbf{y}} = \tilde{\mathbf{V}}^\dagger\tilde{\mathbf{y}}, \quad \text{where} \quad \tilde{\mathbf{V}}^\dagger = (\mathbf{V}^T\mathbf{V})^{-1}\mathbf{V}^T. \tag{2.13}$$

The residual vector and its covariance matrix are next defined

$$\mathbf{e}_l(\omega) = \left(\varepsilon_l(\omega)\ \varepsilon_{l+1}(\omega) \cdots \varepsilon_{l+M-1}(\omega)\right)^T, \tag{2.14}$$

$$\mathbf{Q}(\omega) = E\left(\mathbf{e}_l(\omega)\mathbf{e}_l^H(\omega)\right), \tag{2.15}$$

where $E(\cdot)$ denotes the expectation operator, and in practice is replaced by a sample mean estimator. The following two notations are also required by the definition of MAPES power spectral estimator:

$$\boldsymbol{\rho}(\omega) = \begin{pmatrix} e^{j\omega 0}\mathbf{a}(\omega) \\ \vdots \\ e^{j\omega(L-1)}\mathbf{a}(\omega) \end{pmatrix}, \tag{2.16}$$

$$\mathbf{D}(\omega) = \begin{pmatrix} \mathbf{Q}(\omega) & & 0 \\ & \ddots & \\ 0 & & \mathbf{Q}(\omega) \end{pmatrix}. \tag{2.17}$$

In the ith EM iteration, the probability density function (PDF) of the missing data vector $\boldsymbol{\mu}$ conditioned on the available data $\boldsymbol{\gamma}$ and other context parameters is complex Gaussian with mean and variance denoted by $(\mathbf{b}, \mathbf{K})$ as follows

$$\mathbf{b}_i(\omega) = \tilde{\mathbf{U}}^T\boldsymbol{\rho}(\omega)\alpha_i(\omega) + \tilde{\mathbf{U}}^T\mathbf{D}_i(\omega)\tilde{\mathbf{V}}\Big(\tilde{\mathbf{V}}^T\mathbf{D}_i(\omega)\tilde{\mathbf{V}}\Big)^{-1}\Big(\boldsymbol{\gamma} - \tilde{\mathbf{V}}^T\boldsymbol{\rho}(w)\alpha_i(w)\Big), \tag{2.18}$$

$$\mathbf{K}_i(\omega) = \tilde{\mathbf{U}}^T\mathbf{D}_i(\omega)\tilde{\mathbf{U}} - \tilde{\mathbf{U}}^T\mathbf{D}_i(\omega)\tilde{\mathbf{V}}\left(\tilde{\mathbf{V}}^T\mathbf{D}_i(\omega)\tilde{\mathbf{V}}\right)^{-1}\tilde{\mathbf{V}}^T\mathbf{D}_i(\omega)\tilde{\mathbf{U}}. \tag{2.19}$$

Then the estimates for spectral magnitude $\alpha(\omega)$ and residual matrix $\mathbf{Q}$ are updated in terms of equations

$$\alpha_{i+1}(\omega) = \frac{\mathbf{a}^H(\omega)\mathbf{S}^{-1}(\omega)\mathbf{Z}(\omega)}{\mathbf{a}^H(\omega)\mathbf{S}^{-1}(\omega)\mathbf{a}(\omega)}, \tag{2.20}$$

$$\mathbf{Q}_{i+1}(\omega) = \mathbf{S}(\omega) + \left(\alpha_{i+1}(\omega)\mathbf{a}(\omega) - \mathbf{Z}(\omega)\right)\left(\alpha_{i+1}(\omega)\mathbf{a}(\omega) - \mathbf{Z}(\omega)\right)^H, \tag{2.21}$$

where the auxiliary matrices are defined as follows

$$\begin{pmatrix} \mathbf{z}_0 \\ \vdots \\ \mathbf{z}_{L-1} \end{pmatrix} = \mathbf{U}\boldsymbol{\gamma} + \mathbf{V}\mathbf{b}(\omega), \tag{2.22}$$

$$\mathbf{Z}(\omega) = \frac{1}{L}\sum_{l=0}^{L-1} \mathbf{z}_l e^{-j\omega l}, \tag{2.23}$$

$$\mathbf{S}(\omega) = \frac{1}{L}\sum_{l=0}^{L-1} \Gamma_l + \frac{1}{L}\sum_{l=0}^{L-1} \mathbf{z}_l \mathbf{z}_l^H - \mathbf{Z}(\omega)\mathbf{Z}^H(\omega). \tag{2.24}$$

In (2.24), $\Gamma_{\mathbf{0}}, \cdots, \Gamma_{\mathbf{L-1}}$ are $M \times M$ sub-block matrices located on the main diagonal of matrix $\mathbf{UKU^T}$.

Finally, the MAPES power spectral density estimator can be expressed as

$$\Phi_{MAPES}(\omega) = \frac{|\alpha(\omega)|^2}{\tilde{N}}. \tag{2.25}$$

Actually, in our *in silico* experiments, assuming $\tilde{N} \leq 50$, MAPES yields an estimate of power spectral about two orders of magnitude more computational time (roughly about one hundred times slower) than Lomb-Scargle and Capon methods. Also, the simulation results do not indicate any performance improvement for MAPES in terms of the ability to discover published cell cycle genes. A more detailed comparison between these schemes will be presented in the simulation section.

4. Periodicity Test

Based on the obtained power spectral density, each gene is to be classified as either cyclic or non-cyclic. The null hypothesis is usually formed to assume that the measurements are generated by a Gaussian noise stochastic process. For a general periodogram or power spectral density estimator $\Phi(\omega)$, Fisher's test can be exploited to examine the significance of the detected peak. The Fisher's test statistic is defined as

$$T = \frac{\max_{1 \le k \le N_0} \Phi(\omega_k)}{N_0^{-1} \sum_{1 \le k \le N_0} \Phi(\omega_k)} , \tag{2.26}$$

where $N_0 = \lfloor (\tilde{N} - 1)/2 \rfloor$ since the spectra on the defined frequency grid are symmetric. The p-value for detecting the largest peak is given by [70]

$$P(T > t) = 1 - e^{-N_0 e^{-t}} . \tag{2.27}$$

A rejection of the null hypothesis based on a p-value threshold implies the power spectral density contains a frequency with magnitude substantially greater than the average value. This indicates that the time series data contain a periodic signal and the corresponding gene is cyclic in expression. Notice also that a more accurate estimation method for the p-values can be found in Fisher [71] or Brockwell [72]. The rank of genes ordered by their p-values is of additional importance and it helps to hedge the risk of dichotomous decisions.

For the Lomb-Scargle periodogram, $\Phi_{LS}(\omega)$ is exponentially distributed under the null hypothesis [63], a result which is also exploited in [64]. However, this exponential distribution is not applicable for a general power spectral density. Therefore, Fisher's test is employed to perform the comparison among different spectral schemes. Our simulation results also show that for Lomb-Scargle periodogram, the gene ranks generated by Fisher's test do not differ much from that produced by the exponential distribution. Finally, we remark that other periodicity detection tests exist, as indicated by the robust Fisher test [55], the likelihood ratio test and the χ^2 test [70].

5. Multiple Testing Correction

In order to prevent the false positives from overwhelming the true positives, the multiple testing correction, as proposed in [73] and [74], is performed to control the false discovery rate (FDR). For each of the n measured genes, the periodicity is tested and a p-value is generated. All p-values are sorted in ascending order with the smallest ith p-value denoted by $p_{(i)}$. Assume an estimate of the number of non-cyclic genes among all n genes is $\widehat{n_0}$, and the testing procedure preserves the k genes with the smallest p-values, then an estimate of FDR can be expressed as

$$\widehat{FDR}_k = \frac{p_{(k)}\widehat{n_0}}{k}, \tag{2.28}$$

where the numerator is an estimate of the number of false positives. Since generally periodic genes only occupy a small portion of all genes, $\widehat{n_0}$ is set to n directly in our simulation. Such an action brings a slightly larger estimate. There exist other statistical methods to estimate $\widehat{n_0}$, e.g., [74].

The $\widehat{FDR}$ is not a monotonic function of k, the number of preserved genes. This property makes it tough to choose a p-value threshold. To combat this, the q-value proposed in [73], is defined as follows:

$$q_k = \min_{k \le j \le n} \widehat{FDR}_j. \tag{2.29}$$

The q-value is a monotonically increasing function with respect to k. The FDR can be controlled via specifying the q-value threshold as τ, through which the number of genes to preserve can then be derived as

$$k = \max_{1 \leq j \leq n} q_j \leq \tau. \tag{2.30}$$

C. Simulation Results

Our *in silico* experiments are first performed on the *Saccharomyces cerevisiae* (budding yeast) data set. The Lomb-Scargle, Capon and MAPES are compared. Then we proceed to analyze the *Drosophila melanogaster* (fruit fly) data set.

1. Simulation on *Saccharomyces Cerevisiae*

The performance of the three schemes is evaluated based on the *Saccharomyces cerevisiae* (budding yeast) data set reported by Spellman et al. [18]. In the biological experiments the mRNA concentrations of more than 6,000 Open Reading Frames (ORF) were measured for the yeast strains synchronized by using four different methods, namely, α factor, cdc15, cdc28 and elutriation. The data set contained 73 sampling points, while several observations were missing for some genes.

The current literature provides prior knowledge about the yeast cell cycle genes. Spellman et. al. [18] enumerated 104 cell cycle genes that were verified in previous biological experiments, while Lichtenberg et al. [75] summarized 105 genes that were not involved in the cell cycle. By exploiting these two control sources, we can evaluate the true and false positives generated by the three spectral estimation methods.

The comparison procedure is as follows: based on the given data set, the three schemes are run in such a manner to preserve a pre-specified number of genes. These genes are marked as cell-cycle genes and are compared with two control gene sets, from which the number of positives is counted. If a preserved gene also exists in the gene set which has been verified to be cell cycle regulated, this hit is counted as a true positive. On the other hand, if the preserved gene appears in the gene set which has been corroborated not to be involved in the cell cycle, this hit is counted as a false positive. Notice that since we expect the non-cell-cycle genes to be the majority of all measured genes, but the verified non-cell-cycle genes are only a small portion of all the genes, the false positives from verified non-cell-cycle genes only provide a reference but not a significant knowledge of the false positives. Because the three algorithms perform similarly for all four data sets, only simulation outcomes for cdc15 are presented here to exemplify the general results. The cdc15 data set contained 24 time points sampled from $t_0 = 10$ minute to $t_{N-1} = 290$ minute. The greatest common divisor (gcd) for all time intervals is $\delta = 10$ minutes. Therefore, $N = 24$ and $\tilde{N} = 29$. The bandwidth N_0 of Capon method is 14 while the subvector length M of MAPES is equal to N_0. All three schemes, i.e., Lomb-Scargle, Capon and MAPES, are applied on the data set.

The *in silico* results based on the cdc15 data set are illustrated in Fig. 2. When the number of preserved genes increases, all three schemes increase their ability to identify more cell cycle genes with more false discoveries as a trade-off. Lomb-Scargle achieves the best performance in terms of identifying the highest number of true positives and producing the lowest number of false positives, while MAPES exhibits the worst performance with respect to these two metrics.

To test the algorithm performance on highly corrupted data, two *in silico* experiments are performed. First, one third of all measurements are randomly set to be missing. The results are organized in Fig. 3. Second, a gene's sampled data are added with Gaussian noise of mean 0 and variance equal to half the variance of the gene's measurements. The outcomes of the artificially generated noisy data are presented in Fig. 4. Compared with Fig. 2, all of them identify less verified genes due to the artificially added noise or missed data. The false positives are controlled at a low

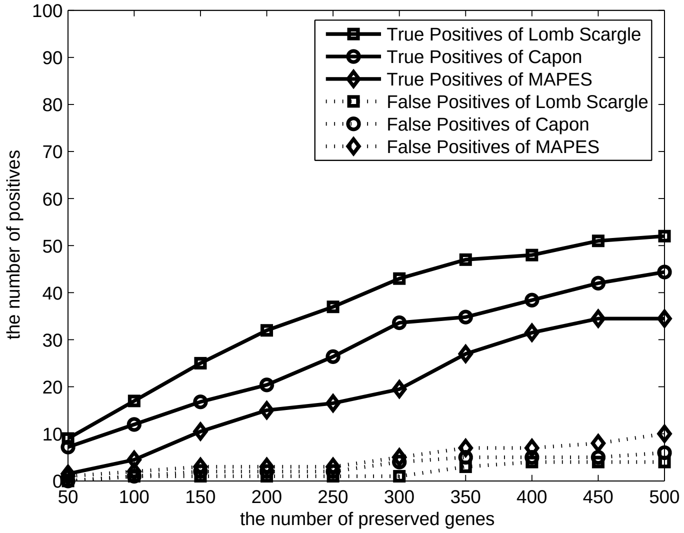


Fig. 2. Performance comparison based on the cdc15 data set.

level. The three algorithms behave in a similar pattern with respect to the increasing number of preserved genes.

Above all, Lomb-Scargle scheme always identifies the largest number of cell cycle genes that have been verified in previous biological experiments. Due to its simplicity, we recommend the use of this simplest method.

2. Simulation on *Drosophila melanogaster*

The *Drosophila melanogaster* (fruit fly) is selected as our research target because it is a well-studied, relatively simple organism with a short generation time and only 4 pairs of chromosomes. In addition, 75% of human diseases have their counterparts in fruit fly, and 50% of fruit fly proteins have their mammalian analogs [76]. These make the fruit fly an excellent model for the research of human diseases. In the literature for the fruit fly most of the research work was conducted through experimental biological methods, and the computational analysis tools have not been fully explored for the

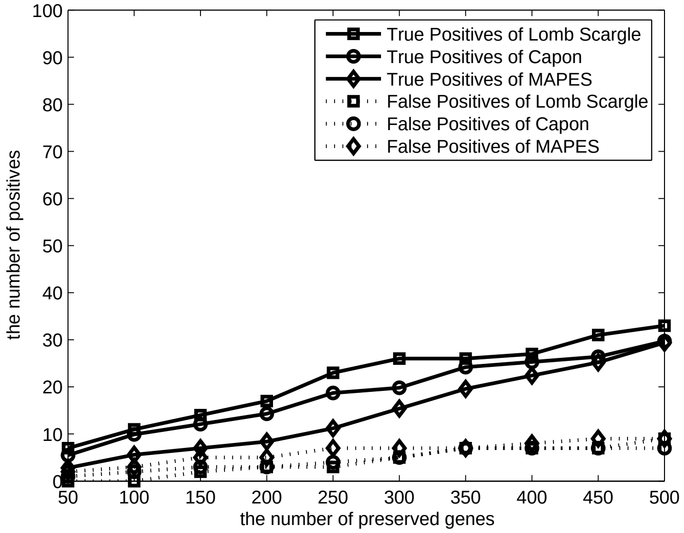


Fig. 3. Performance comparison assuming one third of measurements are randomly set to be missing.

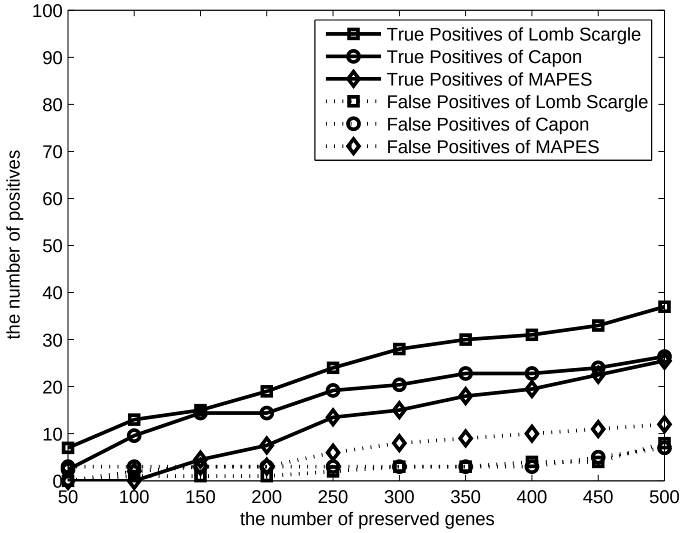


Fig. 4. Performance comparison when noise is intentionally added.

detection of periodically expressed genes. Our *in silico* experiments are performed on the fruit fly data set published by Arbeitman et al. [53]. With the usage of cDNA microarrays, the RNA expression levels of 4028 genes were measured. These stand for about one-third of all found fruit fly genes.

In Arbeitman's experiments 75 sequential sampling points were observed, starting right after fertilization and through embryonic, larval, pupal and early days of adulthood. The time series data during the embryonic stage are analyzed. The embryonic stage gives us insight into the developmental process, i.e., how the fruit fly grows from a zygote to a complex organism with cell specialization. The embryonic data takes the instant of egg lay as the time origin. 30 time points were sampled from $t_0 = 0.5$ hour to $t_{N-1} = 23.5$ hours. The greatest common divisor (gcd) for all time intervals is $\delta = 0.5$ hour. Therefore, $N = 30$ and $\tilde{N} = 47$. The best candidate, Lomb-Scargle algorithm is applied on the data set.

The top 144 genes with the smallest p-values are selected and conferred to be periodic with the highest confidence. These genes are listed in Appendix A. To remove the effects of the DC component, the first two frequency probes are filtered out. The q-value is controlled to be less than 0.2. The majority of genes are associated with a periodicity of about 20 hours, we hypothesize that a portion of them are related to the circadian rhythm. The cell cycle genes are not fully detectable because in the embryonic stage the cells proliferates very fast (minutes). However, the implemented sampling rate was not fast enough to capture the phenomenon in the cell cycle.

3. Discussion on Synchronization Effects

In order to measure a valid sample, the cell culture has to be synchronized, in other words, all cells within the culture should be homogeneous in all aspects, e.g., cell size, DNA, RNA, protein and other cellular contents, and should also mimic the unperturbed cell cycle. Cooper in [77] argued that the ideal synchronization is a mission impossible due to the different dimensions, like cell size and DNA content, that can not be controlled at the same time. Therefore, current popular synchronization methods, like serum starvation and thymidine block, are only one-dimensional synchronization techniques and fail to achieve a truly global synchronization. Cooper also argued it was fully possible that the discovered periodicity was completely caused by chance or by the specific employed synchronization method. The available fruit fly data set was sampled with the synchronization yielded by the Cryonics method. Cryonics is the low temperature preservation method of tissues in which all cell activities are believed to be halted. The cells frozen with liquid nitrogen are compared with control cells, that were fomaldehyde fixed, to ensure that the cells were at the expected developmental stages during sampling. This synchronization method differentiates itself from the one-dimensional methods employed in [18, 52], which have been shown in [77] to present cell cultures that are not actually representative of the cell cycle. Though the damage caused by the freezing was not known, the fly's development assumed true synchronization with the control cells at every developmental check point. This provided enough evidence to consider Arbeitman's data set out of the scope of the issues raised in [77]. Therefore, one can claim with confidence that any discovered periodicity will not have risen from chance fluctuations alone.

CHAPTER III

STEADY STATE MICROARRAY DATA ANALYSIS [*]

A. Problem Overview

Currently, about 70% of the available data sets continue to be time independent due to various reasons such as financial, ethical and practical implementation issues encountered in implementing time course experiments [1]. This impediment represents a strong motivation for developing network inference techniques that exploit the time independent data sets. Since the time independent data sets do not present explicit temporal information, in general it is difficult to infer accurately the regulation relationships. However, it is still possible to infer the direct connectivity between genes due to the inherent properties of the biological system under investigation.

Multiple inference algorithms have been proposed for capturing the gene interactions based on steady state gene expressions. These include [78] for Boolean network models, [13] for probabilistic Boolean networks, [5] and [79] for Bayesian networks and relevance networks [9]. There are also a bunch of scheme in the social science literature to mine the relationships between variables [38] and [80]. The existing machine learning techniques have to be tailored and improved before they can be applied to solve bioinformatics challenges which are significantly different from the traditional learning problems encountered in sociology, industry and other areas.

Score based schemes, e.g., [5], represent a class of computationally intense methods

[*] Part of the data reported in this chapter is reprinted with permission from "Inferring Connectivity of Genetic Regulatory Networks Using Information Theoretic Criteria," by W. Zhao, E. Serpedin and E. R. Dougherty, 2008, *IEEE/ACM Transaction on Computational Biology & Bioinformatics*, vol. 5, no. 2, pp. 262–274, Copyright 2008 by IEEE.

for inference of gene regulatory networks. When heuristic searching approaches are employed for network structure optimization, the efficiency of the inference is greatly impaired and only small scale networks can be inferred. ARACNE [16] represents one of the most recently proposed algorithms in this regard and that infers the direct connectivity among genes using the mutual information as a metric. As reported, ARACNE achieves a better accuracy and high efficiency for large scale networks. However, ARACNE relies on the critical assumptions that gene interactions can be described by Markov chains and the data processing inequality holds [43]. In addition, determination of the significance threshold for mutual information plays an important role and its incorrect specification might induce significant errors, in which case ARACNE falsely connects two distantly separated genes.

By exploiting the conditional mutual information, novel algorithms are designed in this chapter to accommodate more general scenarios. The goal of the proposed algorithms is to preserve with high accuracy only the direct connectivity among the participating genes, maintain a low complexity in network inference, and when a false-alarm connectivity is produced between two genes, the two falsely connected genes are located closely enough in the actual network. Two algorithms are developed along these lines. The first algorithm is for precise inference of direct connectivity. Based on it, an alternative simplified algorithm is proposed, where the connectivity confidence among genes is represented by the so called direct connectivity metric (DCM). DCM is a continuous-valued function that exploits the mutual information and conditional mutual information of gene expressions and provides a more comprehensive description of the connectivity degree between genes, as opposed to the dichotomy of being connected or disconnected. The performance of the proposed inference algorithms is evaluated in the case of several artificial networks. The inference algorithm is then applied on the realistic data sets produced by measurements on cutaneous melanoma. A network containing 470 genes and the $WNT5A$ pathway are recovered using the proposed algorithms. The obtained results are compared with the existing state-of-the-art results, and research target genes are proposed.

B. Algorithm Formulation

The genetic regulation takes place at all stages including transcription, splicing and translation. However, since our inference is based on the microarray data and regulation takes place predominantly at the transcription initiation stage, we constrain these genetic interactions at the transcription stage. In other words, by exploiting the information provided by mRNA transcript data, the proposed network inference algorithms model only the gene-to-gene interactions. Such a modeling framework assumes a large scale modeling of the gene interactions, and not a detailed molecular scale modeling of the interactions among various macromolecules [81], [82]. Since the mutual information represents a consistent measure of the correlation between two random variables even in the presence of nonlinear dependencies, the proposed information theoretic algorithms present a wide applicability area and the inferred conclusions truly reflect the dependencies present in measurement data. The un-oriented graphical model, as depicted in Chapter I, is exploited to represent the structure of genetic networks. Although the inference is also based on statistical relationships, we are aiming to capture the direct regulation relationships as accurately as possible. Next, the concepts of mutual information, conditional mutual information and direct connectivity metric are introduced, and the network inference algorithms are formulated.

1. Information Theoretic Quantities

The information theoretic quantity entropy is a measure of the uncertainty present in the values assumed by a random variable [43]. For a discrete random variable X, which might be either a vector or a scalar, the entropy $H(X)$ is defined by

$$H(X) = -\sum_{x \in \mathcal{X}} [p(x) \cdot \log p(x)], \tag{3.1}$$

where $p(x)$ denotes the probability mass function, and $\mathcal{X}$ stands for the alphabet of X. The entropy of a discrete variable is always non-negative. For a continuous-valued random variable X, the differential entropy $h(X)$ is defined as

$$h(X) = -\int_{x \in \mathcal{S}_{\mathcal{X}}} [f(x) \cdot \log f(x)] dx, \tag{3.2}$$

where $f(x)$ denotes the probability density function, and $\mathcal{S}_{\mathcal{X}}$ represents the support of X. The differential entropy is also denoted as $h(f)$ and can take negative values. Therefore, some discrete network inference algorithms, e.g., REVEAL [17], can not be deployed for continuous-valued gene expression data unless the data are quantized and the associated information loss is tolerated.

The mutual information is in general used as a powerful criterion for measuring the dependence between two random variables (RVs) X and Y. For two discrete-valued RVs, the mutual information is expressed as

$$\begin{aligned} I(X;Y) &= \sum_{\mathcal{X},\mathcal{Y}} [p(x,y) \cdot \log \frac{p(x,y)}{p(x) \cdot p(y)}] \\ &= H(X) + H(Y) - H(X,Y), \end{aligned} \tag{3.3}$$

while for two continuous-valued RVs it takes the expression:

$$\begin{aligned} I(X;Y) &= \int_{\mathcal{S}_{\mathcal{X}}}\int_{\mathcal{S}_{\mathcal{Y}}}[f(x,y)\cdot\log\frac{f(x,y)}{f(x)\cdot f(y)}]dxdy \\ &= h(X)+h(Y)-h(X,Y). \end{aligned} \tag{3.4}$$

Both discrete and continuous versions of $I(X;Y)$ are non-negative and assume the value zero if and only if X and Y are independent. Continuous-valued RVs should be employed to describe the original DNA microarray data, while discrete-valued RVs are used to model quantized expression data.

If gene X interacts with gene Y, in the steady state it is hypothesized that the expression values of X and Y show a strong dependence. This is partially evidenced by the study of chemical kinetics. When the chemical reaction achieves the equilibrium, the concentrations of all participating complexes can be modeled by an equation and they depend on each other. Therefore, if $I(X;Y)$ assumes a very small value, it can be reasonably inferred that X and Y are disconnected in the genetic regulatory network. However, the opposite statement does not hold. Given a large $I(X;Y)$, X and Y can be either directly connected or connected through an intermediate gene. Considering a scenario where three genes X, Z and Y are positioned in a chain $X \to Z \to Y$. In this case all three pairs (X,Y),(X,Z) and (Y,Z) present mutual information greater than zero. If only the mutual information is used to evaluate the connectivity, it is highly possible that the inference might provide a false alarm edge $X - Y$.

ARACNE [16] employs the data processing inequality (DPI) to remove the indirect connectivity. Taking into account both orientations, the DPI states the following result: if X, Z and Y form a Markov chain, i.e., $X \to Z \to Y$ or $X \leftarrow Z \leftarrow Y$ then

$$I(X;Y) \leq \min[I(X;Z), I(Y;Z)] \ . \tag{3.5}$$

If DPI is satisfied, ARACNE infers that X and Y are disconnected.

The DPI works for the chain scenario but loses validity in other general cases. For example, assume a diverging scenario where two genes X and Y share the same regulator Z, i.e., $X \leftarrow Z \rightarrow Y$. All three pairs ($X$,$Y$),($X$,$Z$) and ($Y$,$Z$) present positive mutual information but there is no definite inequality between them. This diverging case is common in scale-free networks where some hub genes regulate several downstream genes. To deal with such cases we exploit the concept of conditional mutual information. Its discrete-valued version is defined as

$$I(X;Y|Z) = \sum_{\mathcal{X},\mathcal{Y},\mathcal{Z}} [p(x,y,z) \cdot \log \frac{p(x,y|z)}{p(x|z) \cdot p(y|z)}], \tag{3.6}$$

where $p(x,y|z)$, $p(x|z)$ and $p(y|z)$ are conditional probability mass functions. The continuous-valued version of conditional mutual information is defined in the form of

$$I(X;Y|Z) = \int_{\mathcal{S_X}} \int_{\mathcal{S_Y}} \int_{\mathcal{S_Z}} [f(x,y,z) \cdot \log \frac{f(x,y|z)}{f(x|z) \cdot f(y|z)}] dxdydz, \tag{3.7}$$

where $f(x,y|z)$, $f(x|z)$ and $f(y|z)$ stand for conditional probability density functions.

The conditional mutual information can be expressed alternatively by the summation of different entropies [43]:

$$\begin{aligned} I(X;Y|Z) &= H(X|Z) - H(X|Y,Z) \qquad (3.8) \\ &= H(X,Z) - H(Z) - (H(X,Y,Z) - H(Y,Z)) \\ &= H(X,Z) + H(Y,Z) - H(Z) - H(X,Y,Z). \end{aligned}$$

For continuous-valued case, the notation $H(\cdot)$ is often represented in terms of $h(\cdot)$. In both the diverging and chain scenarios, given the intermediate or hub gene Z, genes X and Y become independent, and therefore, the conditional mutual information $I(X;Y|Z)$ tends to be zero.

2. Entropy Estimation

Since both mutual information and conditional mutual information can be represented as a summation of entropies, the (conditional) mutual information estimators will be constructed based on multivariate entropy estimates. In turn the entropy estimation depends on the mass or density estimators.

The gene expressions are quantized into q-level discrete values, which are predetermined by the data nature or quantization process. For example, for $q = 3$, the values $\{-1, 0, 1\}$ represent the gene expression levels: repressed, normal and induced, respectively. In general, it is assumed that the q-level quantization admits the alphabet $\mathbf{A}_q = \{0, 1, \cdots, q-1\}$. Then, the probability mass function from m samples $\{\mathbf{s_1}, \cdots, \mathbf{s_m}\}$ is estimated as:

$$\hat{p}(\mathbf{x} = \mathbf{v}) = \frac{1}{m} \sum_{k=1}^{m} \mathbf{1}_{\{\mathbf{v}\}}(\mathbf{s}_k) \ , \tag{3.9}$$

where $\mathbf{1}_{\{\cdot\}}(\cdot)$ stands for the indicator function, defined as

$$\mathbf{1}_{\mathbf{A}}(\mathbf{s}) = \begin{cases} 1 & \text{if} \quad \mathbf{s} \in \mathbf{A}, \\ 0 & \text{if} \quad \mathbf{s} \notin \mathbf{A}. \end{cases} \tag{3.10}$$

By plugging (3.9) into (3.1), and substituting the entropy estimates into (3.3) and (3.8), the estimates of mutual information and conditional mutual information are obtained for the discrete case.

Estimation of (conditional) mutual information of continuous-valued RVs is also divided into two steps. Kernel density estimation methods are first applied for obtaining the empirical density function as follows. If m samples $\{\mathbf{s}_1, \cdots, \mathbf{s}_m\}$ are collected, then a general approach is given by

$$\hat{f}(\mathbf{x}) = \frac{1}{m}\sum_{i=1}^{m}\frac{\mathcal{K}\big(\mathbf{H}^{-1}(\mathbf{x}-\mathbf{s}_i)\big)}{\det(\mathbf{H})} \ , \tag{3.11}$$

where $\mathcal{K}$ stands for the multivariate kernel function and $\det(\cdot)$ stands for determinant function. $\mathbf{H}$ represents the bandwidth matrix, and is a key parameter in density estimation. For simplicity, a diagonal bandwidth matrix $\mathbf{H}$ and multiplicative kernel $\mathcal{K}$ are used. Assuming $\mathbf{x} = (x_1, \cdots, x_d)^T$, we have

$$\mathbf{H} = \mathrm{diag}(h_1, \cdots, h_d), \quad \mathcal{K}(\mathbf{u}) = \mathcal{K}(u_1, \cdots, u_d) = \prod_{j=1}^{d}\mathcal{K}(u_j). \tag{3.12}$$

By plugging (3.12), equation (3.11) takes the form:

$$\hat{f}(\mathbf{x}) = \hat{f}(x_1, \cdots, x_d) = \frac{1}{m}\sum_{i=1}^{m}\Big(\prod_{j=1}^{d}\frac{1}{h_j}\mathcal{K}\big(\frac{x_j - s_{i,j}}{h_j}\big)\Big) \ . \tag{3.13}$$

The kernel $\mathcal{K}$ can be selected as Gaussian, Epanechnikov, cosine functions etc. The bandwidth vector $(h_1, \cdots, h_d)^T$ can be specified according to the rule-of-thumb criteria in [83].

By substituting the density estimates into the differential entropy and by computing the integral, estimates of differential entropy are obtained and a natural estimator of entropy is given by

$$\hat{h}(f_{\mathbf{X}}) = -\frac{1}{m}\sum_{i=1}^{m}\log\big(\hat{f}(\mathbf{x}_i)\big). \tag{3.14}$$

We remark that the recent progress in the area of estimating information theoretic quantities has lead to a number of alternatives for estimating the entropy: [45], [84]. In the proposed algorithms, it is the rank of the mutual information that accounts for the connectivity. Therefore, the desired estimator has to exhibit small variance and acceptable bias.

3. Inference Algorithm

Our inference algorithm utilizes both mutual information and conditional mutual information. In the first step, the continuous-valued expressions of each gene $X_{(\cdot)}$ are rank-transformed. For example, let $x_1, x_2, \cdots, x_m$ stand for m observations of gene X's expression. If x_i $(i \in [1, m])$ is the k-th smallest from the m values, then x_i is reassigned the value k/m. Only ranks of data are preserved. Therefore, outliers with incredible large values are removed and the negative preprocessing effects are reduced. The same technique is also used in [16]. Then all pairwise mutual information terms $I(X_i; X_j)$ are calculated and stored into the mutual information matrix $\mathbf{M}$. Let $\mathbf{M}_{i,j}$ stand for the entry (i, j) of matrix $\mathbf{M}$. If $\mathbf{M}_{i,j}$ is less than a threshold t_M, X_i is assumed disconnected from X_j. Otherwise, we have to proceed to evaluate all the conditional mutual information terms given any other gene X_k. If X_k is a gene belonging to a totally different biological process, the conditional mutual information $I(X_i; X_j|X_k)$ approximates the mutual information $I(X_i; X_j)$ and both assume large values. On the contrary, if X_k is an intermediate or hub gene between X_i and X_j, $I(X_i; X_j|X_k)$ assumes a small value. Hence, given any other gene if the least conditional mutual information is greater than a threshold t_S, it can be inferred that X_i connects X_j. The inference algorithm is formulated as the Algorithm 1 and it returns the connectivity matrix $\mathbf{C}$, in which a null entry means disconnection.

Contrary to optimization-based schemes, which randomly generate candidate networks and select the best one with the highest score, the proposed algorithm does not involve any heuristic search procedure and is non-parametric. These properties are especially appealing for inference of large scale networks with unknown kinetics. A major difficulty of the algorithm is to specify appropriate values for the two thresholds t_M and t_S. Similar difficulties exist in other schemes such as relevance networks

```
Input gene expression data set;
Initialize $n, \mathbf{M} \in \Re^{n\times n}, \mathbf{L} \in \Re^{1\times n}, \mathbf{C} \in \{0,1\}^{n\times n}, t_M, t_S$;
Preprocess the input data set, perform rank transformation
for $i = 1$ to $n - 1$ do
    for $j = i + 1$ to $n$ do
        $\mathbf{M}_{i,j} \Leftarrow I(X_i; X_j)$;
        if $\mathbf{M}_{i,j} < t_M$ then
            $\mathbf{C}_{i,j} = 0$,$\mathbf{C}_{j,i} = 0$;
        else
            $\mathbf{C}_{i,j} = 1$,$\mathbf{C}_{j,i} = 1$;
            for $k = 1$ to $n$ and $k \neq i, j$ do
                $\mathbf{L}_k \Leftarrow I(X_i; X_j | X_k)$;
                if $\mathbf{L}_k < t_S$ then
                    $\mathbf{C}_{i,j} = 0$,$\mathbf{C}_{j,i} = 0$;
                    Break;
                end
            end
        end
    end
end
Return $\mathbf{C}$.
```

Algorithm 1: Connectivity Inference Algorithm

[9] and ARACNE [16]. One possible approach is to learn these thresholds from past knowledge or simulations. For example, we can run simulations on data produced by biologically verified genetic networks and determine the thresholds which optimize the performance of the algorithm. Because of the various sample sizes, data processing techniques and volatile biological phenomena, the predetermined thresholds may still not be reliable. Another disadvantage of the algorithm is that it recovers all the relationships within the dichotomy of being either connected or disconnected. However, in practice, it is more desirable to evaluate the significance of the recovered connectivity, i.e., given any two genes X and Y, with how much confidence can the connectivity $X - Y$ be recovered. The concept is similar to the hypothesis test: not simply accept or reject the null hypothesis, but provide a p-value as a measure of how

much evidence we have against the null hypothesis. This kind of approach helps to hedge the bet.

4. Direct Connectivity Metric and Simplified Algorithm

The inference of direct connectivity between two genes X and Y is based on two information theoretic criteria, the mutual information $I(X;Y)$, and the least conditional mutual information given any other gene Z, i.e., $\min_{Z \in \mathbf{V}_{-XY}} I(X;Y|Z)$, where $\mathbf{V}_{-XY}$ stands for the whole gene set excluding the genes X and Y. Therefore, the direct connectivity metric (DCM) can be defined as a function $g(\cdot,\cdot)$ of these two parameters

$$\eta(X;Y) = g\big(I(X;Y), \min_{Z \in \mathbf{V}_{-XY}} I(X;Y|Z)\big), \tag{3.15}$$

where $\eta(X;Y)$ represents the DCM between X and Y. Larger DCM values are associated with a higher confidence level on the hypothesis that the inferred relationship assumes a direct connectivity.

Specifically, the Algorithm 1 infers a pair of genes to be either connected or disconnected. Hence, the DCM is binary valued, i.e., $\eta(X;Y) \in \{0,1\}$, and the DCM function $g(\cdot,\cdot)$ is defined as

$$g(a,b) = \mathbf{1}_{(t_M,\infty)}(a) \cdot \mathbf{1}_{(t_S,\infty)}(b). \tag{3.16}$$

The intuition behind the design of the DCM function is the observation that when both the mutual information and the least conditional mutual information assume large values, the two genes are more likely to be directly connected. Therefore, we propose $g(\cdot,\cdot)$ to be the product of its arguments:

$$g(a,b) = a \cdot b. \tag{3.17}$$

```
Input gene expression data set, specify $e$ the expected number of connections;
Initialize $n, \mathbf{M} \in \Re^{n \times n}$, $\mathbf{S} \in \Re^{n \times n}$, $\mathbf{L} \in \Re^{1 \times n}$, $\mathbf{C} \in \{0,1\}^{n \times n}$;
Preprocess the input data set, perform rank transformation
for $i = 1$ to $n$ do
    for $j = i + 1$ to $n$ do
        $\mathbf{M}_{i,j} \Leftarrow I(X_i; X_j)$;
        for $k = 1$ to $n$ and $k \neq i, j$ do
            $\mathbf{L}_k \Leftarrow I(X_i; X_j | X_k)$;
        end
        $\mathbf{S}_{i,j} \Leftarrow \min_k \mathbf{L}_k$;
        $\eta_{j,i} = \eta_{i,j} = g(M_{i,j}, S_{i,j})$;
    end
end
$\eta_a = reshape(\eta, 1, n \times n)$, change the matrix $\eta$ into an array;
$\eta_b = sort(\eta_a)$ in descending order;
$\forall i, j \in \{1 \cdots n\}$ if $\eta_{i,j} > \eta_b(e)$ then
    $\mathbf{C}_{i,j} = \mathbf{C}_{j,i} = 1$;
end
else
    $\mathbf{C}_{i,j} = \mathbf{C}_{j,i} = 0$;
end
return $\mathbf{C}$.
```

Algorithm 2: Simplified Algorithm. Function names conform to Matlab.

The genetic regulatory networks are usually sparse. The average degree of each vertex, i.e., the average number of edges connected with each vertex shows statistical stability. Relevant statistics can be found in various works on genetic networks such as [50] and [51]. Therefore, given a large scale genetic regulatory network with a specific number of genes, the amount of network edges can be predetermined within a small range. In addition, biologists desire to examine first the connectivity that present high confidence and then proceed with less confident connectivity. Hence, we can only consider an expected number of edges corresponding to the highest DCM's. The second algorithm, formulated as the Algorithm 2, associates each gene pair (X,Y) with a DCM and returns a specified number of edges.

C. Simulation Results

The proposed simulation results include two main constitutive parts. The performance of the proposed algorithms is first tested on a set of data created by artificial networks, which are connected graphs. The algorithms are then applied on a realistic cutaneous malignant melanoma data set to propose meaningful intervention targets.

1. Simulation on Synthetic Data Sets

a. Refined Performance Definition

By representing the synthetic and inferred graphs as $\mathbf{G}(\mathbf{V},\mathbf{E})$ and $\hat{\mathbf{G}}(\hat{\mathbf{V}},\hat{\mathbf{E}})$, respectively, the performance of an algorithm is evaluated based on the differences between $\mathbf{G}$ and $\hat{\mathbf{G}}$, as is defined in Chapter I. Type I errors, i.e., false alarms, can be further grained into sub-categories in terms of actual vertex distance. If the inference algorithm creates in $\hat{\mathbf{G}}$ a false-alarm connection $X-Y$, in $\mathbf{G}$, X and Y may be separated by n hops with $n \geq 2$. The distance, i.e., the number of hops of the shortest path between vertices X and Y in graph $\mathbf{G}$ can be expressed by $d_{\mathbf{G}}(X,Y)$, then an n-order (n-hop) false-alarm edge $X-Y$ is specified by $X-Y \in \hat{\mathbf{E}}$ and $d_{\mathbf{G}}(X,Y) = n$ $(n \geq 2)$. The artificial graphs used in the simulation are all connected graphs, i.e., there is always an undirected path between any two nodes. Hence, $d_{\mathbf{G}}(X,Y)$ assumes a small number. If the graph is unconnected, we can specify a somewhat large value to $d_{\mathbf{G}}(X,Y)$. Owing to the small world property of the genetic network, this value could be relatively small, e.g., less than 6. When false-alarm edges are produced, it is desired that the inference algorithm constrains the false alarms within low-order sub-categories.

A novel comprehensive performance metric is introduced here to measure the difference between two graphs $\mathbf{G}(\mathbf{V},\mathbf{E})$ and $\hat{\mathbf{G}}(\hat{\mathbf{V}},\hat{\mathbf{E}})$. The distance from graph $\mathbf{G}$ to

$\hat{\mathbf{G}}$ is defined by

$$D(\mathbf{G}, \hat{\mathbf{G}}) = \frac{\sum_{X-Y \in \hat{\mathbf{E}}, X-Y \notin E} d_{\mathbf{G}}(X,Y) + \sum_{X-Y \in \mathbf{E}, X-Y \notin \hat{E}} 1}{|E|} \ , \tag{3.18}$$

where the first summation of the numerator corresponds to all false-alarm edges, the second summation counts the miss errors, and the denominator represents the number of edges in graph $\mathbf{G}$. It has to be noted that the metric D is not symmetric with respect to its two arguments. A good algorithm assumes a small D, provides a reduced number of misses, recovers as many as possible of the true (real) connections, and the possible false alarms belong to the lowest order (2-hop) sub-category. When the algorithm is applied on a realistic data set, we can not compute this performance metric since the actual genetic network is unknown as it represents the inference target. Therefore, there is no need to worry about its computational burden and it should not be counted into the complexity of the proposed algorithm.

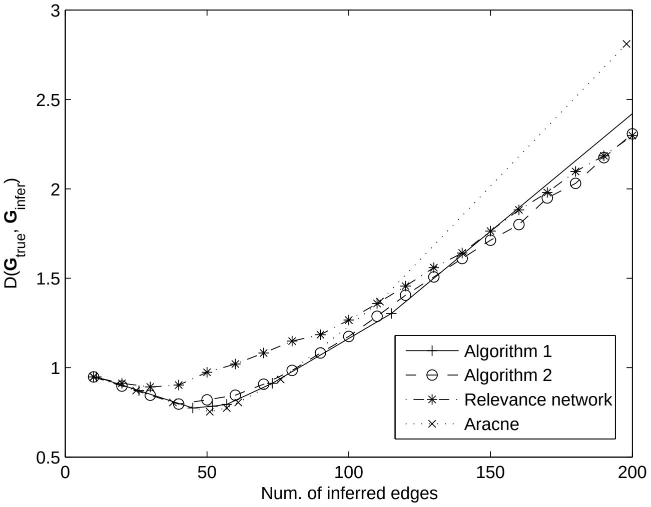
b. Simulation Results and Discussion

Four algorithms are compared in this section. These algorithms are: the proposed two algorithms, ARACNE [16] and relevance network method [9], which employs only the mutual information as a connectivity metric. Algorithms are simulated on the artificial scale-free networks generated by Mendes [48] and Bulcke [49]. Steady state data for Mendes networks are provided by Margolin and are also used in the simulation of ARACNE. Each Mendes network contains 100 genes and 200 oriented interactions, while 100 genes and 164 oriented interactions are created for Bulcke's network model. The thresholds in the four algorithms have to be tuned in order to get the specified numbers of inferred edges. Particularly in our Algorithm 1, we simply set the two thresholds to the same value and change them jointly so that the number of inferred edges matches the expected number of edges in the graph. The comparisons are categorized into the following aspects.

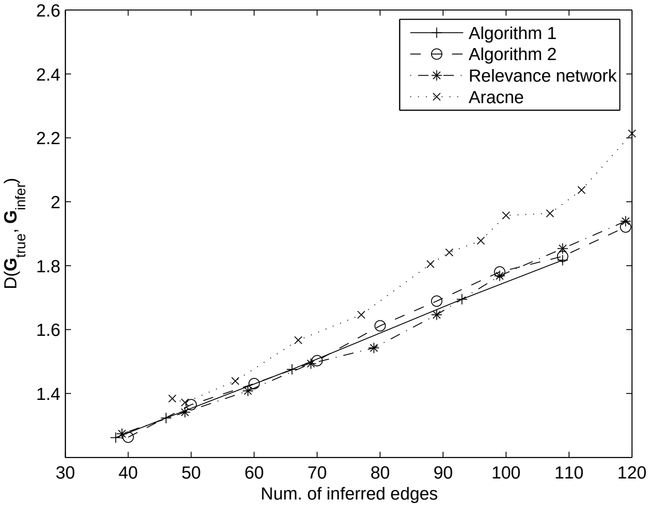
Comprehensive Performance Metric. It can be seen in Fig. 5(a) that the proposed algorithms and ARACNE outperform the relevance network method [9] when the number of inferred edges is less than 120. When the number of inferred edges keeps increasing to larger values, the performance of ARACNE deteriorates and becomes inferior to the relevance network method and to our two proposed schemes. ARACNE has to specify a proper mutual information threshold so that its performance locates in the head portion. In the full range, the proposed two schemes achieve or approximate the best performance. Fig. 5(b) presents performance results for Bulcke's data set. ARACNE exhibits poorer performance relative to the proposed schemes and relevance network method in the whole range of simulation. The proposed schemes still achieve or approximate the best performance.

True Connections. The four algorithms exhibit contrasting ability in inferring the direct connectivity for different data sets. For Margolin's data set, as shown in Fig. 6(a), when the number of inferred edges is less than 50, the proposed algorithms and ARACNE perform much better than the relevance network method and most of recovered edges are true connections. Algorithm 1 and ARACNE are the best and Algorithm 2 approximates the best. When the number of inferred edges increases, the difference between algorithms is not that pronounced. For Bulcke's data set, as shown in Fig. 6(b), all four algorithms are not successful in recovering the direct connectivity in the whole range of inferred edges. In this case, most of the inferred edges are false alarms.

False Alarms. Although the four algorithms perform inconsistently for the two data sets, the proposed algorithms and relevance network method always produce less higher-order false alarms than ARACNE. For Margolin's data set, as shown in Fig. 7(a), the proposed algorithms and ARACNE start to produce false alarms when more

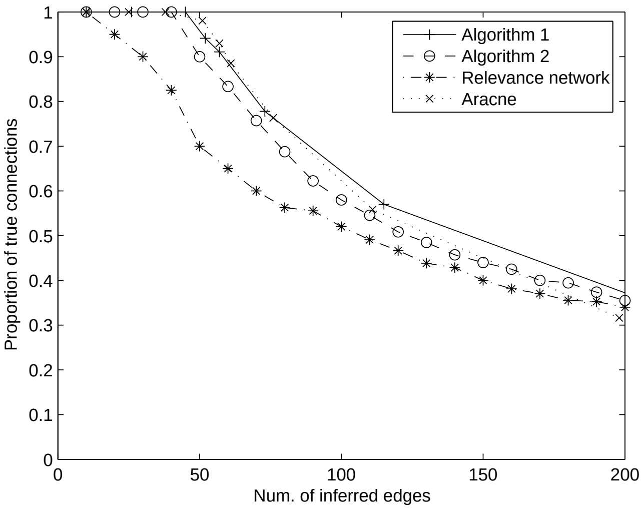


(a) Performance at presence of Margolin's data set

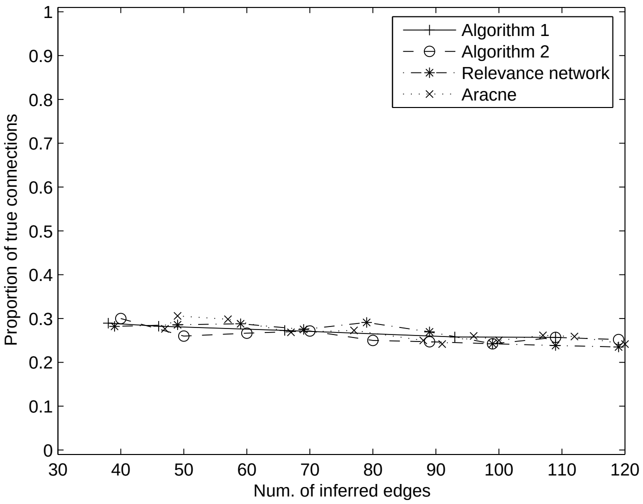


(b) Performance at presence of Bulcke's data set

Fig. 5. Comparison of algorithms in terms of the comprehensive performance metric. (a) Proposed algorithms achieve the best performance metric and ARACNE performs well in the head portion. (b) Proposed algorithms and relevance network method are better than ARACNE in terms of the performance metric.

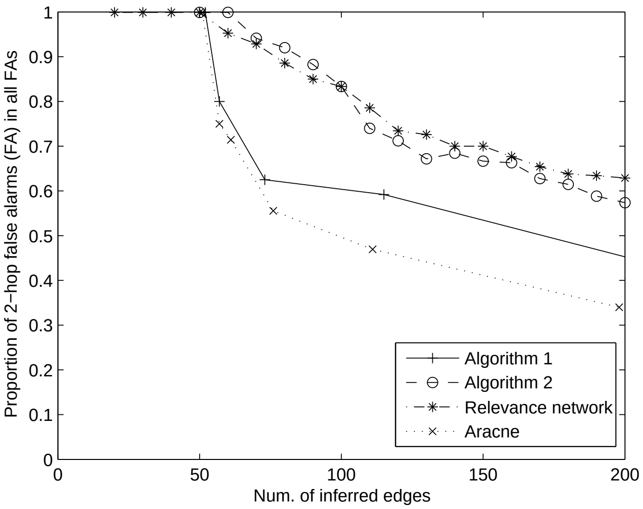


(a) Number of true positives relative to the total positives for Margolin's data

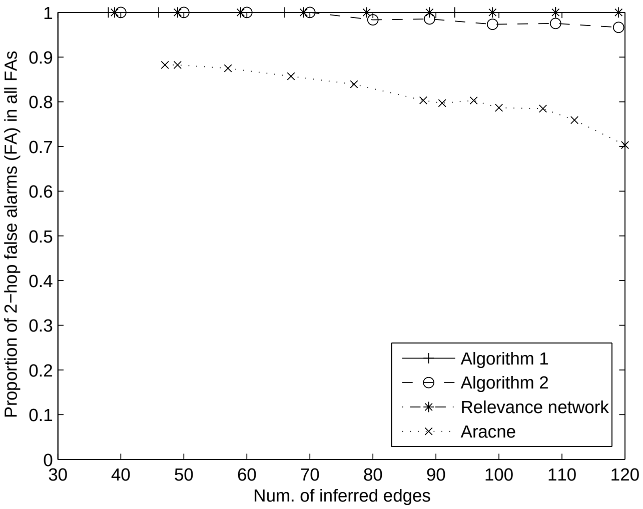


(b) Number of true positives relative to the total number of positives for Bulcke's data

Fig. 6. Comparison of algorithms in terms of the number of correctly inferred connections. (a) Proposed algorithms and ARACNE are better in inferring the direct connectivity for Margolin's data set. (b) All schemes perform similarly in recovering the direct connectivity for Bulcke's data set.



(a) Number of 2-hop false alarms relative to the total number of false alarms for Margolin's data



(b) Number of 2-hop false alarms relative to the total number of false alarms for Bulcke's data

Fig. 7. Comparison of algorithms in terms of false alarms. (a) ARACNE creates higher-order false alarms, while the other three schemes create lower-order false alarms for Margolin' data set. (b) ARACNE creates higher-order false alarms, while the other three schemes create lower-order false alarms.

than 45 edges are inferred. However, most of the false alarms in the Algorithm 2 and the relevance network are 2-hop false alarms, while ARACNE falsely connects many vertices which are actually separated by more than 2 hops. For Bulcke's data set, as shown in Fig. 7(b), the difference among the proposed algorithms, the relevance network and ARACNE still holds. Nearly all false alarms in the proposed algorithms and relevance network method are 2-hop false alarms, while an increased portion of false alarms in ARACNE are higher-order false alarms.

It has to be noted that there exist structures that can never be correctly inferred by the present computational methods. A simple example is a network that consists of 3 genes X, Y and Z. X regulates both Y and Z via the linear equations $Y = 2X$, and $Z = 0.5X$. In such a scenario, any valid method will recover a triangle instead of the true diverging case $Y \leftarrow X \rightarrow Z$. In these cases, false-alarm edges are mapped into co-expressions or co-regulations, in which case falsely connected nodes X and Y are actually separated by 2 hops. Such a semantic caveat is also described in [85]. The proposed schemes aim to differentiate direct regulations from co-expressions. When such endeavor fails, they are still successful in maintaining the connected vertices to be located closely in the actual networks.

The two artificial network generators differ in their modeling assumptions. Mendes randomly generates network topologies, while Bulcke derives topologies from established large scale genetic networks. They are also distinct in choosing interaction types and setting transition parameters. The two algorithms share the Michaelis-Menten and Hill enzyme kinetics which employ differential equations to model gene interactions. It is unknown yet which generator is more prone to be realistic. The proposed schemes consistently achieve or approximate the best performance. ARACNE runs at a risk of creating high order false alarms, while the relevance network method is not good for discovering direct connectivity. The Algorithm 2 is advantageous for simplifying the specification of thresholds and providing a more accurate metric for direct connectivity than mutual information. The DCM can also be interpreted as a metric for assessing the distance between two vertices. A large DCM means the two vertices are directly interacting, while a small DCM means more hops between the two vertices. Therefore, the DCM can be used as an alternative distance metric for mutual information.

Merging the results of these different algorithms constitutes an interesting open problem. A simple solution in this direction is to consider a majority voting scheme among the results produced by these algorithms. Such a merging can be easily achieved for algorithms assuming dichotomous decisions. However, the Algorithm 2 assumes a metric to assess the significance of gene interactions. This metric can not be easily combined with other algorithms proposed in the literature. Therefore, the fusion of multiple data sources and inference algorithms remains an open research topic, which will be discussed in Chapter V.

2. Simulation on Melanoma Data Set

The algorithms are simulated on the cutaneous malignant melanoma data set [86], which contains the expressions of 527 genes from 31 patients. Two proposed algorithms generate similar results and the network inferred by Algorithm 2 is presented. A big picture containing 470 genes and 500 connections is shown in Fig. 8. The distribution of vertex degree d, i.e., the number of edges connected with each vertex, is shown in Table I. The figure shows a proneness towards scale-free networks rather than random (Erdos-Renyi) networks since a large proportion of edges are connected with the hub genes, which are listed also in Table I. These hub genes constitute the backbone of the network and they are potential control targets.

The algorithm are also useful for recovering specific gene pathways, which contain

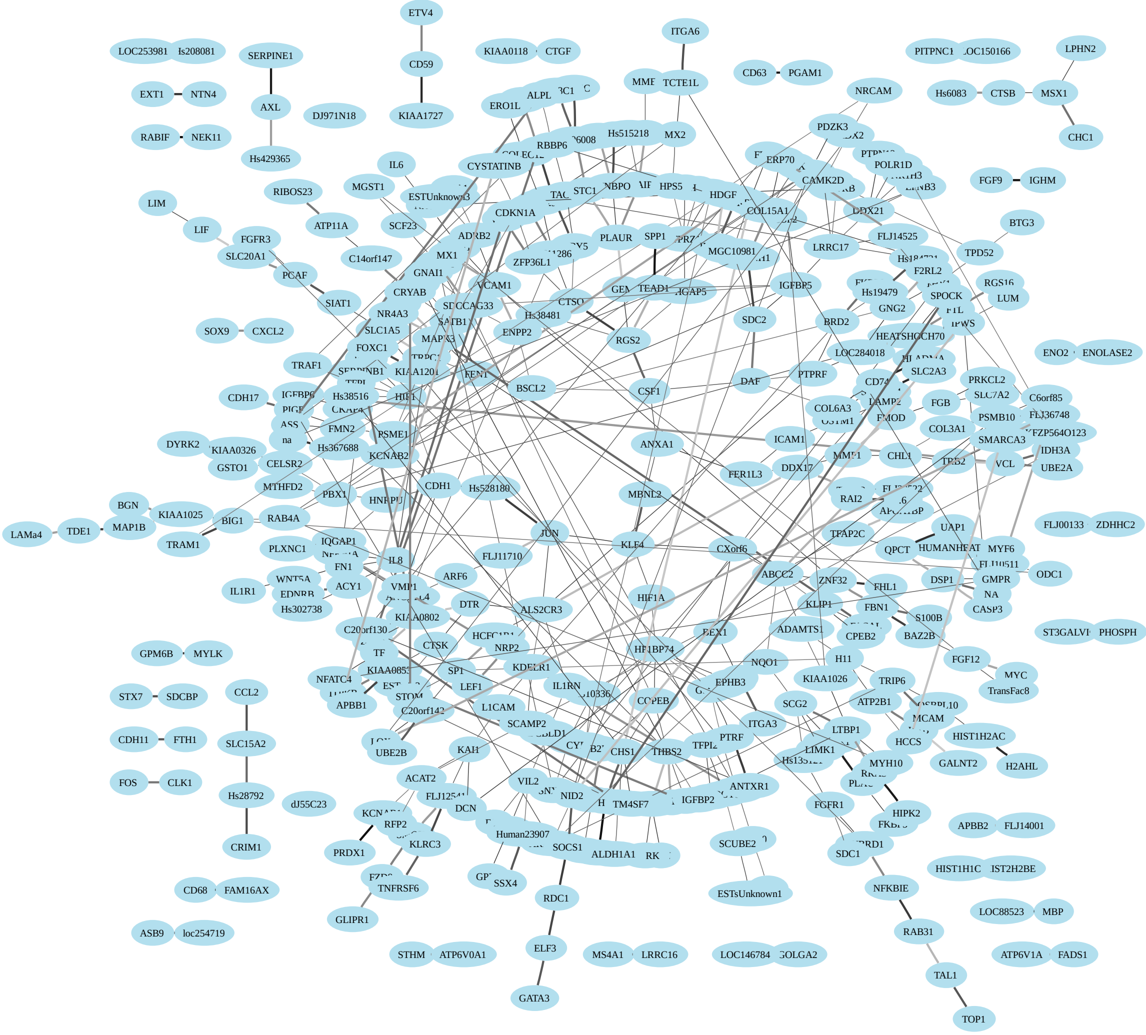


Fig. 8. Genetic regulatory network of 470 key genes. Standing alone genes are removed from the figure and duplicated genes are combined into a single vertex. Connections with top 500 DCM's are preserved. The direct connectivity metric (DCM) is represented by line's thickness and greyness. A thicker and blacker line corresponds to a greater DCM value.

Table I. Vertex degree statistic for 470-gene network

degree	$d = 0$	$d = 1$	$d = 2$	$d = 3$	$d = 4$	$d = 5$	$d \geq 6$
genes	57	204	121	74	43	14	14
hub genes	ALS2CR3 THBS2 SDCCAG33 LTBP1 SCG2 IL8						
($d \geq 5$)	C1orf29 DDX21 PTPRZ1 CCND1 NFKBIA CDH1						
	COPEB PSME1 NID2 RGS2 FEN1 HP1BP74						
	PSMB10 IPWS C20orf130 ABCC2 C5orf13						
	CHS1 HIP1 LRRC17 IGFBP5 PBX1						

Table II. Vertex degree statistic for WNT5A pathway

degree	$d = 0$	$d = 1$	$d = 2$	$d = 3$	$d = 4$	$d = 5$	$d \geq 6$
genes	13	30	30	20	16	2	6
hub genes	THBS1 Hs28792 FN1 SLC1A5						
($d \geq 5$)	SERPINB2 NR4A3 SNCA PLAUR						

gene interactions around key genes and provide integrated functions for the cell. WNT5A has been recognized as a key gene in the metastatic melanoma [87]. It affects cell motility and invasion. 20 neighbors of WNT5A are selected according to their high mutual information with WNT5A. Similarly, for each neighbor, 20 neighboring genes are selected. A second-order pathway is constructed for $WNT5A$ and 117 genes are included. The recovered pathway is shown in Fig. 9 and the degree statistic is shown in Table II.

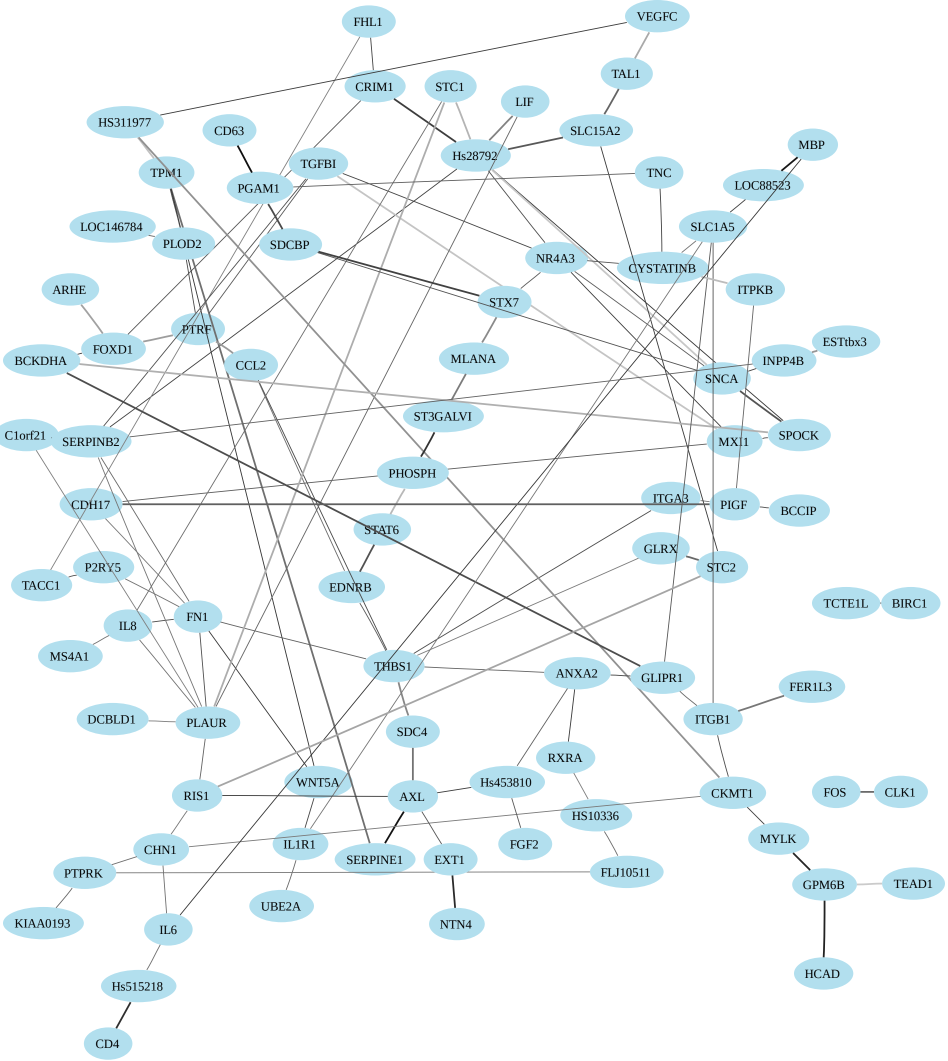


Fig. 9. $WNT5A$ pathway. 104 Genes with large mutual information with $WNT5A$ and its neighbors are displayed. Connections with the top 130 DCM's are preserved. The direct connectivity metric (DCM) is represented by line's thickness and greyness. The thicker and blacker edges correspond to interactions with greater DCM values.

An examination of hub genes can be indicative of biological phenomena. Three genes, $Plaur$ (plasminogen activator, urokinase receptor, alias $uPAR$), $Serpinb2$ (Plasminogen activator inhibitor type 2, alias $PAI-2$) and $Fn1$ (fibronectin 1) form a thick-lined triangle which is located near $WNT5A$ in the network. $Plaur$ plays a key role in tumor cell invasion, survival and metastasis in a variety of cancers [88]. $Serpinb2$ is found to be an inhibitor of $Plaur$ in the quantitative research of breast cancer kinetics [89]. The algorithm verified the direct connectivity between $Plaur$ and $Serpinb2$. $Fn1$ is related to cell growth and differentiation, and participates in the anti-tumor activity [90]. From the network viewpoint, a simultaneous control posed on the triangle $Plaur - Serpinb2 - Fn1$ is proposed here for further anti-tumor research.

Other three important hub genes are $Thbs1$ (thrombospondin 1, alias $TSP1$), $SLC1A5$ (solute carrier family 1 member 5, alias $ASCT2$) and $NR4A3$ (nuclear receptor subfamily 4, group A, member 3, alias $NOR-1$). $Thbs1$ is a critical regulator of vasculature formation [91] and has been studied in a variety of mouse model systems as an inhibitor of tumorigenesis. $Thbs1$ is strongly connected with $Fn1$ in the recovered network. The co-regulation of $Fn1$ and $Thbs1$ was identified in the study dedicated to human ovarian cancer suppression [92]. $SLC1A5$ is associated to metabolism. It is responsible for glutamine uptake in hepatoma cells and its expression is necessary for the growth of liver cancer [93]. $NR4A3$ exerts transcriptional functions through its activation and induction of downstream pathways. It is also reported as a factor of cell apotosis and carcinogenesis [94].

It can be seen that most hub genes assume important roles in the carcinoma and they have been research targets for different tumor therapies. What is presented here is a systematic view of the gene interactions. Genes cooperatively participate in the biological processes and bestow cells integrated functions. A simultaneous control on multiple hub genes could maintain a stable system and constitute a cocktail therapy, which may intervene into the cancerous organism from different aspects, e.g., cutting off the nutrition provision by inhibiting the vasculature formation, repressing metastasis by shutting down the cancerous cell proliferation. A perturbation on a single gene might also be cautiously conducted to produce cascade effects.

CHAPTER IV

TIME SERIES MICROARRAY DATA ANALYSIS [*]

A. Problem Overview

This chapter infers the structure of genetic regulatory networks by using time-course microarray data. To capture gene regulations, this chapter assumes a probabilistic network modeling framework compatible with the family of models represented by dynamic Bayesian networks (DBNs) and probabilistic Boolean networks (PBNs). As opposed to PBNs, where gene interactions are modeled explicitly in terms of binary or multi-valued logical functions, the proposed probabilistic model represents gene interactions in terms of probability tables. In addition, the proposed probabilistic network can be viewed as the transition network present in DBNs. In sum, all of these models can be considered as sharing similar basic features.

The strength of temporal relationships will be evaluated by using a cross-time mutual information metric. The minimum description length (MDL) principle [44] is utilized to determine a threshold that helps differentiate between strong and weak relationships. The MDL principle helps also to achieve a good trade-off between the network model complexity and the accuracy of data fitting. The proposed network inference algorithm is comprised of two components: encoding of the model, i.e., the network, and encoding of the time series data. After combining the network and data coding complexities, a general criterion is obtained for constructing the network so as to contain only direct and oriented interactions. The convergence of the proposed

[*] Part of the data reported in this chapter is reprinted with permission from “Inferring gene regulatory networks from time series data using the minimum description length principle,” by W. Zhao, E. Serpedin and E. R. Dougherty, 2006, *Bioinformatics*, vol. 22, pp. 2129–2135, Copyright 2006 by Oxford University Press.

MDL-based network inference algorithm is corroborated by the excellent recovery of the topology of some artificial networks and through the error rate plots obtained through extensive simulations on data sets produced by synthetic networks. When applied on real drosophila time series data sets, the proposed network inference algorithm corroborates some of the findings of Arbeitman et al. [53], and offers novel insights into the regulatory mechanisms that lie at the basis of embryonic segmentation and muscle development in drosophila melanogaster.

Historically, Tabus and Astola 2001 [95], were the first to report some preliminary results on the potential of the MDL principle in learning gene-expression networks; however, their work is limited to using the MDL principle in the prediction of gene expressions, while the present paper focuses on the more general task of learning the network structure. The mutual information has been exploited in the Reveal algorithm proposed by Liang et al. [17]. In contrast to Reveal, the proposed algorithm removes the critical assumption that all genes have to be observed, utilizes only pairwise mutual information, achieves better performance in the presence of reduced number of samples, improves greatly the computational efficiency, and requires reduced computing capabilities even in the presence of large scale networks. These information-theoretic approaches possess several attractive features: low computational complexity, novel ideas for quantifying efficiently the dependencies among a large number of genes and efficient testing (estimation) of various relationships among information-theoretic quantities (entropy, mutual information).

B. Systems and Methods

1. Genetic Network Formulation

Given a set of genes, an oriented graph $\mathbf{G}(\mathbf{V}, \mathbf{E})$, where $\mathbf{V}$ denotes the set of vertices and $\mathbf{E}$ represents the set of oriented edges, is used to map the gene interactions. Each vertex represents a specific gene and at a specific time is associated with a gene expression value. This chapter assumes discrete-valued gene expression levels but no specific limit on the number of quantization levels is enforced. Each edge of the graph denotes a directed regulation (i.e., an oriented edge with a precise temporal regulation implication). Recall the notation Π_X is used to represent the set of predecessors which regulate gene X. Similarly, the notation Ξ_X is used to represent the set of successor genes which are regulated by gene X.

Associated with a specific gene X is the regulation function $f_X(\Pi_X)$, which denotes the expression value for gene X determined by the values of the genes in the set of predecessors Π_X. For simplicity, the shorthand notation f_X will be used since Π_X is uniquely determined in the biological world. For instance, the Boolean relation *if either gene Y or gene Z is induced, gene X will be induced* can be represented by $f_X = y + z$ (with + denoting the logical *or* (summation) operator).

The gene expression is affected by many internal and external factors, e.g., other genes, environmental variables, and many other unknown factors. Since it is impossible to account for all factors, all regulation functions are assumed probabilistic to reflect this uncertainty. In addition, the gene expression values are assumed discrete-valued and the probabilistic regulation functions are represented as look-up tables. Suppose each gene expression is quantized into q levels. If X has n predecessors, i.e., $|\Pi_X| = n$, then the look-up table corresponding to regulating function f_X contains q^n rows and q columns; hence, a total of q^{n+1} entries. Each entry corresponds to a

Table III. Probability table for *or*

X:YZ	00	01	10	11
0	0.8	0.2	0.2	0.2
1	0.2	0.8	0.8	0.8

$x = y + z$ with confidence 0.8

conditional probability. For instance, with the probability 0.8, X will be induced if Y is induced and Z is repressed. By denoting the repression and induction as binary values 0 and 1, respectively, the previous regulation function can be expressed in terms of $p(x = 1|yz = 10) = 0.8$. Hence, the entry at row 3 and column 2 is filled with the value 0.8. Considering the relationship $f_X = y + z$ with probability 0.8 and $f_X = \overline{y + z}$ with probability 0.2, where the over-line denotes negation, Table III can be used to represent this probabilistic relationship.

All the functions are defined over the temporal domain, i.e., the expression values for the set Π_X at time t determine the value for gene X at time $t+1$. For this reason all functions must assume a time dependent form $x(t + 1) = f_X(\Pi_X(t))$. Given m time series samples $\boldsymbol{x}_t, \cdots, \boldsymbol{x}_{t+m}$ starting at time t, the information conveyed by these samples is represented in terms of the joint probability function $p(\boldsymbol{x}_t, \cdots, \boldsymbol{x}_{t+m})$. Estimation of joint probability functions over short time periods $k << m$, i.e., $\hat{p}(\mathbf{x}_t)$, $\hat{p}(\mathbf{x}_t, \mathbf{x}_{t+1})$, $\cdots$, $\hat{p}(\mathbf{x}_t, \cdots, \mathbf{x}_{t+k})$, can be achieved with satisfactory precision, whereas for longer time intervals it becomes more difficult.

In this chapter the concept of mutual information continues to be used to evaluate the significance of regulation, and the significance threshold is determined using the MDL principle. These two concepts (mutual information and MDL) lie at the basis of the proposed network inference algorithm.

2. Metric for Assessing Temporal Regulation

If gene Y regulates gene X at time slot t with a latency 1, X_{t+1} has to depend on Y_t. Conversely, if gene X at time slot $t+1$ is dependent on the gene expression Y at a previous time slot t , we can infer that gene Y regulates gene X in time scale 1. The cross-time dependency is considered as the metric for assessing the temporal regulation. The gene system is assumed to be event driven, i.e., all the regulations are performed step by step and in each step all regulations happen only once. Therefore, the latency parameter is set by default to a unit step.

Compared with the correlation coefficient, the mutual information is suitable for nonlinear relations and represents a good metric for evaluating the dependency between two random variables [43]. Explicit time stamps are assumed in the mutual information criterion for measuring the significance of gene Y regulating gene X in one step:

$$I(X_{t+1};Y_t) = \sum_{x_{t+1},y_t} [p(x_{t+1},y_t) \cdot log \frac{p(x_{t+1},y_t)}{p(x_{t+1}) \cdot p(y_t)}], \tag{4.1}$$

where $p(x_{t+1},y_t)$ and $p(x_{t+1})$ are cross-time joint and marginal probabilities, respectively. These probabilities are assumed time invariant. It is well known that the mutual information $I(X;Y)$ between two arbitrary random variables X and Y is always greater than or equal to zero, and it is zero if and only if X and Y are independent. Large mutual information between X_{t+1} and Y_t supports the proposition that Y regulates X in one step with a high probability. In such a case, the inference algorithm assumes an edge from Y to X on the graph. Assuming that the q-level quantization of gene expressions admits the alphabet $\mathbb{A}_q = \{0,1,\cdots,q-1\}$, the marginal and joint probabilities from m-sample time series $\{x_1, \cdots, x_m\}$ and $\{y_1, \cdots, y_m\}$ are given by:

$$\hat{p}(x = j) = \frac{1}{m} \sum_{t=1}^{m} \mathbf{1}_{\{j\}}(x_t), \tag{4.2}$$

$$\hat{p}(x_{t+1} = i, y_t = j) = \frac{1}{m-1} \sum_{t=1}^{m-1} \mathbf{1}_{\{ij\}}(x_{t+1}y_t), \quad for\ i, j \in \mathbb{A}_q. \tag{4.3}$$

where $\mathbf{1}_{\{\cdot\}}(\cdot)$ still stands for the indicator function.

The mutual information can also be defined between two groups of genes rather than a pair. Only pairwise mutual information is utilized in the proposed algorithm because of the limitation of sample size and computational complexity. It is unlikely that the number of time points available in expensive microarray measurements will rapidly increase in the near future, therefore the estimation of multivariate probability is less reliable when higher order statistics are employed. Besides, high order computations request much more memory and CPU time, which is a huge burden even for mainframe computers if very large scale networks have to be inferred.

Assume that all the cross-time mutual information between genes are collected in the entries of the regulation matrix $\mathbb{M}$, i.e., $\mathbb{M}_{y,x} = I(X_{t+1}; Y_t)$. A key problem that needs to be resolved is to find a proper threshold δ such that when $\mathbb{M}_{y,x} \geq \delta$ (or $\mathbb{M}_{x,y} \geq \delta$), then one can infer with high probability that Y regulates X (or X regulates Y) and there is potentially an oriented edge from Y to X (or from X to Y) in the network graph. On the contrary, if $\mathbb{M}_{y,x} < \delta$ and $\mathbb{M}_{x,y} < \delta$, there is no relationship between X and Y, and hence, X and Y are disconnected. Then another followup step assumes scanning of all candidate edges and trimming of all suspect connections based on a reliable criterion. Another key issue concerns the construction of unbiased and consistent estimators for mutual information in the presence of reduced number of samples. Recent progress in estimating information theoretic quantities has led to a number of good estimators in this regard, e.g. [45]-[47], [84] and [96].

3. Minimum Description Length Principle

Given a network and a data set, the MDL principle is employed to evaluate simultaneously the goodness of fit of the network and data. Intuitively, the more complicated the network is, the better the data would be fitted. However, very often models which are over-fitted relative to the actual systems are selected, which give rise to numerous errors. The merit of the MDL principle is that it achieves a good trade-off between model complexity and fitness of the data. The MDL principle aims to minimize a criterion L that consists of two parts: the model coding length L_M and the data coding length L_D.

a. Network Coding Length

The proposed network model is an oriented graph. Its coding length is positively proportional to the storage size of the graph. The proposed model's data structure involves arrays for predecessors and matrices for probability tables. For a vertex X, it is required to maintain an array that records Π_X, and if d_i bits are used to code an integer, $d_i|\Pi_X|$ bits are necessary to encode the array that records Π_X. A matrix should also be maintained for conditional probability. If d_f bits are used to represent a floating point number and each vertex is q-level quantized with the alphabet $\mathbb{A}_q$, then $d_f q^{|\Pi_X|}(q-1)$ bits are required to store the conditional probability table associated with vertex X (the multiplicative factor $q-1$ being due to the fact that one degree of freedom is lost because each row of the conditional probability table adds up to one). Supposing that any of the n vertices in the network is indexed by X_i, the network coding length (L_M) can be expressed as:

$$L_M = \Gamma \sum_{i=1}^{n} \{d_i|\Pi_{X_i}| + d_f q^{|\Pi_{x_i}|}(q-1)\}, \tag{4.4}$$

where Γ is a free parameter used to quantify the gap between the proposed network coding length and the ideal information theoretic benchmark, as well as to offer an additional control mechanism between model and data encoding complexities. In other words, this free parameter can be used to ensure that the model encoding mechanism is consistent with the data encoding mechanism. Notice further that the model encoding scheme is not unique, and there are a number of additional unknown factors (number of genes/regulation functions, selection of quantization levels and floating point arithmetic) that might still affect the model and data coding lengths. Normally, Γ should be a positive value less than one ($0 < \Gamma < 1$). As a flexible design variable, Γ can be interpreted as a simple mechanism to balance the uncertainties present in the MDL metric and to weight the relative influence of model and data encoding complexities. Simulation results illustrate that this free parameter enables also a customized trade-off between the two types of inference errors. Γ could be learned from established genetic networks, and it could also be tuned via simulations. The size of integer d_i is determined by the number of vertices $|\mathbf{V}|$. For example, the human genome contains about 25,000 genes and 16 bits are enough to code each gene's index. Therefore, d_i can be expressed as $d_i = \lceil \log_2 |\mathbf{V}| \rceil$, where $\lceil \cdot \rceil$ is the ceil function. The size of floating number d_f is determined by the sample size m. If a large sample size is available, then a relatively precise estimation of the probabilities can be achieved. Consequently, each entry in the truth table presents a higher resolution, and needs more bits to encode it. Practically d_f can be represented by $d_f = \lceil \log_2 m \rceil$.

As can be observed from the analytic dependencies present in (4.4), the network coding length is biased in favor of outgoing edges. That is, each vertex is more likely associated with a large successor set rather than a large predecessor set. However, this feature is consistent with biological findings and does not represent a weakness of the proposed probabilistic modeling framework. Guelzim [51] summarized that the number of regulating genes per regulated gene decayed exponentially while the number of regulated genes per regulating gene decayed in a power law and assumed a broader-support distribution. It is also conjectured that multiple predecessors consume more energy, hence make the coding length larger.

b. Data Encoding Length

Since the network is probabilistic, each gene can randomly commit any value in the alphabet during the next time slot. The network is associated with a Markov chain, which is used to model the transitions between states. These states are represented in terms of the n-gene expression vector $\mathbf{x}_t = (x_{1,t}, \cdots, x_{n,t})^T$. The transition probability $p(\mathbf{x}_{t+1}|\mathbf{x}_t)$ can be derived as follows:

$$p(\mathbf{x}_{t+1}|\mathbf{x}_t) = \prod_{i=1}^{n} p(x_{i,t+1}|\Pi_{X_i,t}). \tag{4.5}$$

The probability $p(x_{i,t+1}|\Pi_{X_i,t})$ can be obtained from the look-up table associated with the vertex X_i and is assumed to be time invariant. Its estimation can be obtained in a similar way to (4.2):

$$\hat{p}(x_{i,t+1} = j|\Pi_{X_i,t}) \quad = \quad \frac{1}{m-1}\sum_{t=1}^{m-1} \mathbf{1}_{\{j\}}(x_{i,t+1}|\Pi_{X_i,t}), \qquad \text{for } j \in \mathbb{A}_q. \tag{4.6}$$

Each state transition brings new information which is measured by the conditional entropy:

$$H(\mathbf{x}_{t+1}|\mathbf{x}_t) = -log(p(\mathbf{x}_{t+1}|\mathbf{x}_t)). \tag{4.7}$$

Therefore, given m time series sample points, $\{\mathbf{x}_1, \cdots, \mathbf{x}_m\}$, the total entropy is

$$L_D = H(\mathbf{x}_1) + \sum_{j=1}^{m-1} H(\mathbf{x}_{j+1}|\mathbf{x}_j). \tag{4.8}$$

The term $H(\mathbf{x}_1)$ in (4.8) is common for all models and can be omitted. The coding length for the data is given by:

$$L_D = \sum_{j=1}^{m-1} H(\mathbf{x}_{j+1}|\mathbf{x}_j). \tag{4.9}$$

Once the coding lengths for the network L_M and the sampling data L_D are obtained, the MDL criterion L is immediately obtained by summing up these two components, $L = L_M + L_D$.

c. Comparison with Other Criteria

Akaike's Information Criterion (AIC), and Bayesian Information Criterion (BIC) are two alternative model selection criteria that are widely used the literature. They can be expressed as follows:

$$\begin{aligned} AIC &= -\log \ell(\hat{\theta}|\mathbf{x}) + K, & (4.10)\\ BIC &= -\log \ell(\hat{\theta}|\mathbf{x}) + \frac{1}{2}K \log m, & (4.11)\end{aligned}$$

where $\hat{\theta}$ stands for the estimation of parameter vector, $\ell(\cdot)$ represents the likelihood function given the sample $\mathbf{x}$, K abstracts the number of parameters and m denotes the sample size. The log likelihood in essence equals the data encoding length term in the proposed MDL criterion. The differences between them lie in the penalty part, which specifies the model complexity. The AIC does not take into account the effect of sample size while BIC and the proposed MDL absorb it into the penalty part. Particularly, the proposed MDL criterion explicitly dissembles the complicated graph parameters in terms of (4.4) and provides the flexibility in trading off the two types of errors. The MDL and BIC criteria will share similar asymptotic features if the parameter K is used to represent the network storage size.

4. Network Inference Algorithm

Given m data points $(\mathbf{x}_1, \cdots, \mathbf{x}_m)$, where each point consists of n gene expressions, $\mathbf{x}_k = (x_{1,k}, \cdots, x_{n,k})^T$ $(k = 1, \ldots, m)$, the first step in the network inference algorithm is to evaluate the cross-time mutual information between any two genes, $I(X_{i,t}; X_{j,t+1})$, and to fill up the corresponding entry $\mathbb{M}_{i,j}$ of matrix $\mathbb{M}$. The next step is determination of the dependency threshold δ with the least MDL metric L, a step which is achieved over n^2 iterations, equal to the maximum number of possible connections among n vertices. Actually, the n^2-complexity can be further reduced to $O(n)$ because of a generally accepted fact in the literature: the genetic regulatory networks are sparse and the number of edges $|\mathbf{E}|$ grows linearly with the number of vertices $|\mathbf{V}|$. Such a statistic can be found for the yeast [51] and drosophila [50]. In the i^{th} iteration, the dependency threshold δ is assigned to be the i^{th} largest value in $\mathbb{M}$. The edge $X_i \to X_j$ is treated as a potential connection, and X_i is put into Π_{X_j}, if $\mathbb{M}_{i,j} \geq \delta$; otherwise, the genes X_i and X_j are treated as not being connected, and the set Π_{X_j} is left unchanged. Upon obtaining the predecessor set $\Pi_{(\cdot)}$ for each vertex, by using (4.6), the set of conditional probabilities can be estimated to fill up the corresponding probability table $\mathbb{T}_{(\cdot)}$ for each vertex. Now all the network parameters have been set up, and the network and data can be encoded to obtain $L_i = L_{M,i} + L_{D,i}$. After n^2 (or $O(n)$) iterations, all the MDL metrics $L_i's$ can be compared and the network with the least L can be selected. This preliminary network might contain false connections. Then in the last step, each edge is scanned and temporally deleted to evaluate whether such a deletion is helpful to reduce the MDL metric. If it does, then the edge is formally removed and the network is updated.

The network inference pseudo-code can be formulated in terms of the Algorithm 3, where lines 1-2 initialize all the variables, line 3 computes all the pair-wise mutual

```
1:  Input time series data set
2:  Initialize $n, \mathbb{M} \in \Re^{n \times n}, \forall j \in \{1 \cdots n\}, \Pi_{X_j} \Leftarrow \phi$;
3:  $\forall (j,k) \in \{1 \cdots n\}^2, \mathbb{M}_{j,k} \Leftarrow I(X_{j,t}; X_{k,t+1})$;
4:  $A \Leftarrow reshape(\mathbb{M}, 1, n^2)$, change the matrix into an array;
5:  $A \Leftarrow sort(A)$ in ascending order;
6:  for $i = 1$ to $n^2$ do
7:      $\delta \Leftarrow A_{(n^2-i+1)}$;
8:      $\forall (j,k) \in \{1 \cdots n\}^2$, if $\mathbb{M}_{j,k} \geq \delta$, then $\Pi_{X_k} \Leftarrow \Pi_{X_k} \cup \{X_j\}$;
9:      $\forall j \in \{1 \cdots n\}, \mathbb{T}_j \Leftarrow p(x_{j,t+1} | \Pi_{X_j,t})$ by using (4.6);
10:     compute $L_{M,i}, L_{D,i}$ by using (4.4) and (4.9) respectively;
11:     $L_i \Leftarrow L_{M,i} + L_{D,i}$;
12: end
13: $h \Leftarrow argMin_i L_i$;
14: restore network in $h^{th}$ loop, $L_{pre} = L_h$;
15: for $i = 1$ to $n$ do
16:     for $j = 1$ to $n$ do
17:         if $j \in \Pi_{X_i}$ then
18:             $\Pi_{X_i} \Leftarrow \Pi_{X_i} \setminus \{X_j\}$, exclude $X_j$ from predecessors;
19:             update $\mathbb{T}_i \Leftarrow p(x_{i,t+1} | \Pi_{X_i,t})$ by using (4.6); compute $L_M, L_D$ by using
                (4.4) and (4.9) respectively; $L \Leftarrow L_M + L_D$; if $L > L_{pre}$ then
20:                 $\Pi_{X_k} \Leftarrow \Pi_{X_k} \cup \{X_j\}$;
21:             end
22:         end
23:     end
24: end
25: Return the inferred network.
```

Algorithm 3: Network Inference Algorithm

information terms, lines 4-5 sort the mutual information terms, lines 6-12 perform a forward step by adding edges, lines 13-14 obtain the preliminary network, lines 15-27 perform a backward step by deleting possible false-alarm edges, and lines 22-24 restore the network when the deletion is invalid. Note that all function names conform to Matlab conventions.

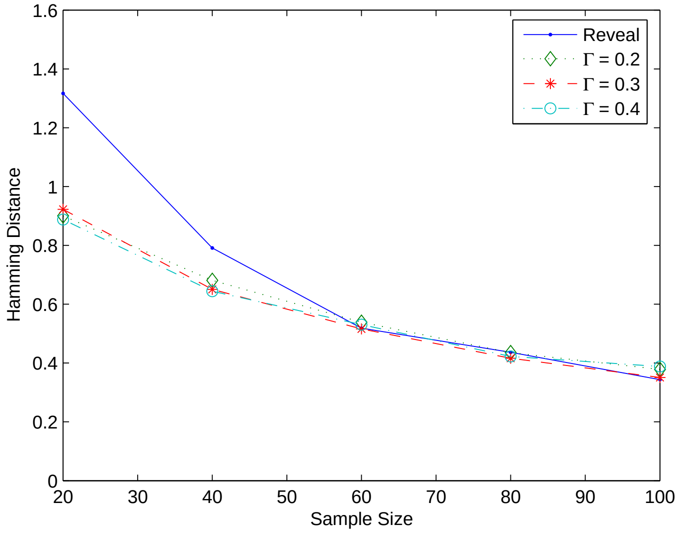
C. Results and Discussion

1. Simulation on Synthetic Networks

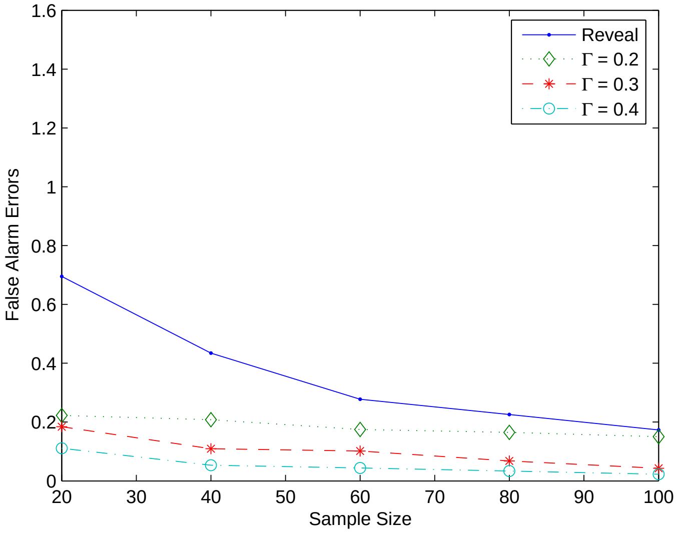
Next, the performance of the proposed network inference algorithm is evaluated on synthetic random boolean networks. The Reveal algorithm proposed by Liang [17] is used as a benchmark to illustrate the advantages of the proposed algorithm. Kevin Murphy implemented Reveal in a toolbox, which can be downloaded at `http://bnt.sourceforge.net`.

Fig.10 shows the performance for Reveal and the proposed algorithm with different Γ configurations. Fig.10(a) stands for the performance in terms of the Hamming distance. The proposed algorithm achieves much better performance when the sample size is less than 60. Avoiding high-order mutual information terms makes the proposed algorithm more accurate for small sample size. When larger sample sizes are observed, the performance of the proposed algorithm is similar to that of Reveal. The Hamming distance is not sensitive to different Γ configurations, and the performance curves for different Γ overlap. Fig.10(b) demonstrates that the proposed algorithm produces less false alarm errors than Reveal. The miss rate is sacrificed in trading for a smaller false alarm rate when Γ is adjusted to a higher value. The functionality of free parameter Γ is obvious and it serves as a good trade-off mechanism between the false alarms and misses. Currently most biological measurements assume within 20 to 50 time points, and the proposed algorithm possesses an attractive performance right in this range.

The Reveal algorithm assumes that all variables/genes can be observed. Such an assumption does not hold in the biological world due to a number of factors. In general, the biological systems are not autonomous and are always affected by environmental variables. Many genes, e.g., non-coding genes, remain still undiscovered, and hence no up-to-date microarray could measure all the genes. Finally, in gen-



(a) Hamming Distance



(b) False alarm errors

Fig. 10. The performance is obtained through averaging over 30 random networks and each network contains 20 vertices and 30 edges. Performance metrics are normalized over 30, the number of edges in synthetic networks.

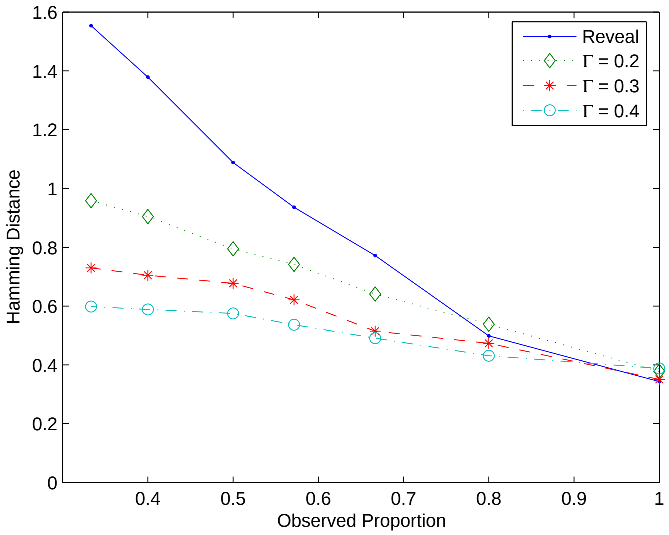


Fig. 11. Observability effects. The performance is obtained by randomly selecting 20 nodes and the associated edges from a larger scale network. The sample size is kept to 100. The Hamming distance is normalized over the number of edges present in synthetic networks.

eral a sub-network is constructed in representing a specific biological functionality. This observability effect is examined by simulating the algorithms on artificial subnetworks $\mathbf{G}_{sub}$, which are constructed by randomly selecting nodes and the associated edges from a larger scale network $\mathbf{G}_{big}$. Fig.11 explains the performance in terms of Hamming distance for both Reveal and the proposed algorithm. The performance advantage of the proposed algorithm is apparent: it is not that sensitive to the observed proportion, i.e., the ratio of the number of vertices in the subnet $|\mathbf{V}_{sub}|$ over the number of vertices in the larger network $|\mathbf{V}_{big}|$.

The proposed algorithm runs efficiently. It only employs pairwise mutual information. For an n-gene network, n^2 pairwise mutual information terms have to be estimated. Given m samples, each mutual information estimation takes $O(m)$ additions and $O(1)$ multiplications. However, if each gene is regulated by at most 3 other genes, i.e., $|\Pi_X| \leq 3$, Reveal has to estimate $\Omega(n^4)$ mutual information terms, which include pair-wise and higher order ones. This makes a big difference between the two algorithms. In practice, on Pentium IV PC with 512MB memory and both algorithms implemented in Matlab, for a network with 20 nodes, 30 edges, and 100 sample points, the proposed algorithm produces a fairly good result in 50 seconds while Reveal requires more than 600 seconds. That is more than 10 times speedup improvement.

Reveal can only deal with small networks (with less than 30 nodes on common PCs) because the space complexity grows as $\Omega(n^4)$, when $\max|\Pi_X| \geq 3$. When n approaches a large value, Reveal will be out of the capacity of even mainframe computers. However, the proposed algorithm can easily deal with a network with hundreds of nodes and its storage size grows as much as $O(n^2)$. For larger networks, we propose to divide the network into subnets and apply the algorithm on each subnet. This divide and conquer technique relies on the fact that genetic networks are prone to scale free, and the proposed algorithm is not susceptible to the observability effect.

The comparisons between Reveal and the proposed algorithm are summarized in the Table IV.

2. Simulation on the Drosophila Data Set

Measuring 74 time points, Arbeitman et al. [53], have presented transcriptional profiles for 4028 Drosophila genes through the four stages of the life cycle: embryonic, larval, pupal and adulthood. We examine our algorithm using this data set and propose a novel muscle development network.

In the first step, the original data set of ratios is quantized into binary values. Let $y_{(1)}$, $y_{(2)}$, $y_{(3)}$, $\cdots$, $y_{(n-3)}$, $y_{(n-1)}$, $y_{(n)}$ be the values of a specific gene expression

Table IV. Performance comparison

Algorithm	Reveal	The proposed alg.
Small sample performance	poor	good
Asymptotic performance	good	good
Observability effect	significant	minor
Time Complexity / Efficiency	$\Omega(n^4)$	$O(n^2)$
Largest network processable	nodes< 30	nodes>> 100

The largest network is tested on a PC with 512MB memory and Pentium IV CPU.

ordered in ascending order. The smallest two values, $y_{(1)}, y_{(2)}$, and the largest two values, $y_{(n-1)}, y_{(n)}$, are treated as outliers and discarded. The dynamic range is defined as $R = y_{(n-2)} - y_{(3)}$. The gene expressions are quantized as follows: the upper 50 percentile of the dynamic range R is treated as induced, while the lower 50 percentile as repressed. If there is a missing time point, a simple linear interpolation is used, i.e., the value of the missed time point is set to the mean of its two neighbors. When the missing point is a start or end point, it is set as its nearest observed (neighbor's) value.

A set of genes is selected to construct a novel genetic regulatory network for the muscle development process. The selected genes have been separately reported to relate with muscle development in different works, e.g. [50], [53], and [97], but no system level diagram exists yet. The inferred genetic network is shown in Fig. 12.

It can be seen in the Fig.12 that the gene *muscle specific protein 300* (*msp*-300), as its name indicates, is a hub gene and regulates *myosin alkali light chain*1 (*mlc*1), *myosin heavy chain* (*mhc*), *myosin 61F* (*myo*61*F*), *paramyosin* (*prm*), and *upheld* (*up*). All these genes except *up* belong to the *myosin* family, which encodes the motor proteins that move along *actin* filaments and are responsible for muscle contraction.

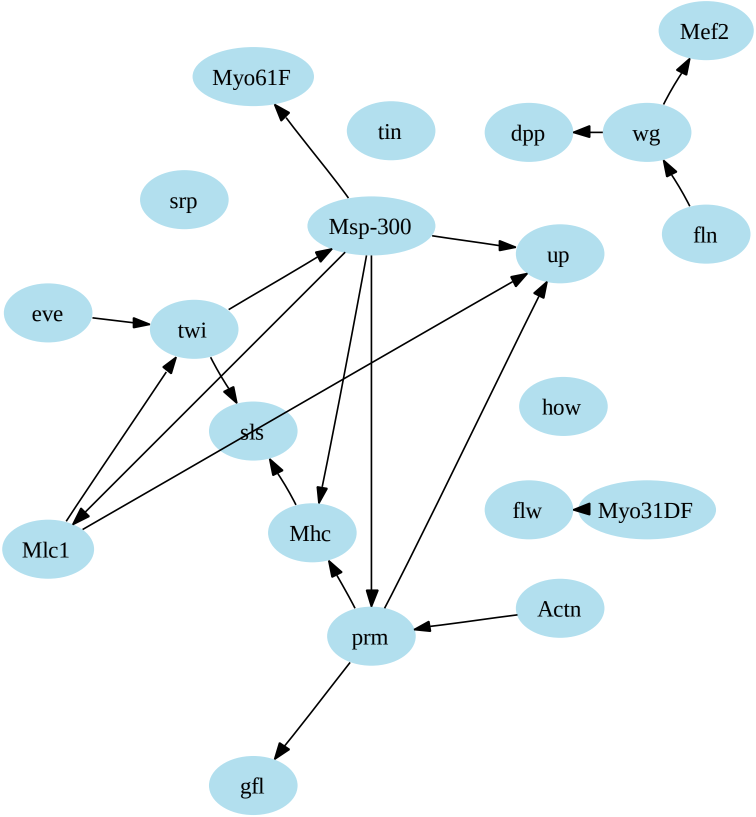


Fig. 12. Muscle development network. 20 genes are chosen according to their appearance in the literature. The free parameter Γ is 0.19 so that most nodes are connected. The network is split into two domains: muscle motor genes and muscle formation genes.

These myosin genes play important roles in cellular mechanics and stand nearby in the network.

A loop is found with genes *msp*-300, *twist* (*twi*) and *mlc*1. The boolean relations associated with this loop are: $twi \Leftarrow eve \cdot \overline{mlc1}$, $mlc1 \Leftarrow msp$-300 and msp-300 $\Leftarrow \overline{twi}$. The network might be intervened by controlling *eve* and *twi*.

The genes *flightin* (*fln*), *wingless* (*wg*), *myocyte enhancing factor 2* (*mef*2) and *decapentaplegic* (*dpp*) form a separate domain from the domain centered around *msp*-300. *Fln* has been shown as a major contributor to muscle development and function.

Wg functions during metamorphosis to coordinate wing formation and *dpp* acts as a morphogen critical for wing patterning [98]. Their cooperation and interactions can be found in the work of [99].

The proposed algorithm provides a systematic view of the drosophila's muscle development. It detaches muscle mechanic genes from formation genes. Further biological experiments are necessary for complete verification of this gene regulatory network.

CHAPTER V

INTEGRATION OF HETEROGENEOUS DATA [*]

A. Problem Overview

Inference of gene regulatory networks based solely on the information provided by microarray data is limited by a number of factors: number of available microarrays, quality of data samples, experimental noise and errors (cross-hybridizations). It is also known that post-transcriptional modifications and transcripts that are present at low levels are generally not detectable by microarrays. Since the gene activity is measured by the mRNA level, the underlying assumption is that there is a significant correlation between the mRNA level and the amount of protein associated with mRNA. However, the magnitude of such a correlation varies significantly depending on the type of protein involved. Therefore, a combined approach which besides gene expression data exploits additional data sources is likely to enhance the inference process.

The advent of in vivo Chromatin Immuno-Precipitation (ChIP) assays has enabled to test whether a protein acting as a transcription factor binds to a specific DNA segment. Hence, ChIP assays serve as a promising mechanism to examine the binding relationships. However, as discussed in Chapter I, the ChIP-chip experiments also inherit some uncertainty concerning the regulation inference since in general the binding is not equal to the regulation relationship.

A combination of both steady state microarray data and ChIP-chip data helps to

[*] Part of the data reported in this chapter is reprinted from "Recovering Genetic Regulatory Networks From Chromatin Immunoprecipitation (ChIP) and Steady State Microarray Data," by W. Zhao, E. Serpedin and E. R. Dougherty, 2008, *EURASIP Journal on Bioinformatics and System Biology*, Open Access 2008 by Hindawi Publishing Corporation.

make more accurate inferences. Intuitively, these two different types of data complement the shortcomings of each other. This motivates us to propose a Bayesian approach to analyze jointly both data sets and to establish a confidence measure of gene interactions. The proposed scheme possesses six key features which make it different from the existing algorithms. First, gene expression data in steady state are considered, while time course data are used in other works like [17], [30] and [37]. Second, most of the current schemes recover a unique genetic network represented by a graph which best fits the observed data in a certain metric, while the proposed approach determines the posterior probabilities for all gene-pair interactions and avoids to make a dichotomous decision that classifies each gene interaction as being either connected or disconnected. The proposed approach can be easily transformed into a dichotomous scheme by only preserving the highly probable gene interactions. Third, the underlying structural model is assumed to be a directed cyclic graph, which allows cycles (feedback loops) and directed acyclic graphs are treated as special cases. This contrasts to Bayesian networks, which are directed acyclic graphs. Feedback loops are a common network motif in biological processes and their function is to yield the necessary redundancy and stability for the system [2]. Therefore, methods based on Bayesian networks, e.g., [38], [100] and [101], lose their validity in the inference of cyclic graphs. Fourth, the proposed approach assumes continuous-valued variables, and this prevents the information loss incurred by data quantization. This represents an advantage compared with the discrete-valued networks such as [38], [100] and [101]. Fifth, the proposed connectivity score is oriented and has a very clear meaning, in the sense of posterior probabilities, while the existing scores based on the mutual information, e.g. [9], [16] and [15], are vague and lack orientation information. Sixth, in the proposed approach the system kinetics are assumed to be nonlinear, while linear models are commonly utilized for the purpose of simplification, e.g. [102] and [103].

Besides, the proposed scheme establishes a general framework whose components can be customized to fit the nature of the underlying biological system.

In this chapter system dynamics is presented to model the genetic expressions. The p-values of ChIP-chip experiments are translated into regulation probabilities and the inference algorithm is formulated through Bayesian analysis. The proposed algorithm and other three schemes are simulated on a set of artificial networks. Performance comparisons illustrate that the proposed algorithm exceeds in terms of several metrics. The robustness of kinetics model is also discussed via simulations. Realistic data sets are exploited in the proposed inference framework and a genetic network is presented to account for the genetic response to environmental changes.

B. System Kinetics Modeling

Genetic regulatory networks can be represented by a parameterized graph $(\mathbf{G}, \Theta)$, where $\mathbf{G}$ and Θ stand for the graph structure and parameter set, respectively. The graph structure qualitatively explains the direct gene interactions, while the parameter set quantitatively describes the system kinetics. General directed graphs (with possibly cycles) will serve as our structural model since they are consistent with the features exhibited by biological systems, in which loops account for system redundancy and stability. Given the graph structure $\mathbf{G}$, the parent set Π is specified for any gene X. For conciseness, the subscript X associated with some variables is omitted in the analysis procedure when the context has clearly specified the gene in question. Next we discuss the system kinetics and parameters defined in Θ.

The system kinetics represents the dynamics that governs the gene's mRNA concentrations in terms of gene-gene interactions. It can be modeled by a set of differential equations (DE). A simplified form is a set of linear DEs. However, we accept the more complex model which was employed previously by [41] and [104] since it is much more realistic and accounts for the expression saturation. Given a gene X, its parent set Π can be further partitioned into two disjoint subsets: the activator set A and the repressor set R, i.e., $\Pi = A \cup R$ and $A \cap R = \phi$. The kinetics of gene X can be explained by the following differential equation:

$$\frac{dx}{dt} = -\lambda x + \frac{\delta + \sum_{i=1}^{|A|} a_i^{\alpha_i}}{1 + \sum_{i=1}^{|A|} a_i^{\alpha_i} + \sum_{j=1}^{|R|} r_j^{\gamma_j}}, \tag{5.1}$$

where x is the concentration of gene X's transcriptional product, namely, mRNA. The changing rate of gene X is controlled by its activating and repressing parents, denoted individually by $a_i \in A$ and $r_j \in R$. α and γ serve as the regulating factors corresponding to each activator and repressor. α and γ assume positive values, and hence can be modeled by a Gamma distribution with shape and scale parameters (κ, β). Here we can unbiasedly assume that the activators and repressers share the same Gamma distribution for their regulation factors. Other light-tail distributions, such as Weibull and lognormal distributions, could also be employed. However, since Gamma distribution is popular in modeling the reaction rate or molecular concentration [105], the Gamma distribution is chosen here. λ stands for the gene degradation rate and the time scale can be properly chosen in order to normalize λ to the unit value ($\lambda = 1$). δ represents the expression baseline rate, i.e., the expression rate for X when there is neither activator nor repressor regulating the target gene X. Suppose y represents the observation of x, then y assumes the form $y = x + \varepsilon$, where ε incorporates all noise sources and is modeled by an additive Gaussian random variable with zero mean and variance σ^2.

As the response to environmental changes or incitations, a mature biological system always converges to a certain steady state, in which all genes stay in equilibrium and do not change their expressions. In this context, the periodic processes, e.g. cell cycle and circadian rhythm, are excluded from our research interest. By setting $dx/dt = 0$ and $\lambda = 1$, the observation y of the steady-state gene expression for gene X can be expressed as:

$$y = \frac{\delta + \sum_{i=1}^{|A|} a_i^{\alpha_i}}{1 + \sum_{i=1}^{|A|} a_i^{\alpha_i} + \sum_{j=1}^{|R|} r_j^{\gamma_j}} + \varepsilon. \tag{5.2}$$

Given a parental structure Π for gene X, the parameters in Θ can be summarized as follows:

1) For each parent $\pi \in \Pi$, a binary variable is demanded to specify whether the parent is an activator or repressor. That is, $\mathbf{1}_A(\pi)$, where $\mathbf{1}$ is the indicator function and it assumes the value 1 when $\pi \in A$, and 0 otherwise. It can be modeled by a Bernoulli random variable with known success probability ρ.

2) For each activator $a \in A$ and repressor $r \in R$, it is assumed that the regulating factors $\alpha, \gamma \sim Gamma(\kappa, \beta)$, where κ, β are known.

3) The baseline parameter δ is usually known.

4) The noise $\varepsilon \sim N(0, \sigma^2)$, where σ^2 can be set to a specific value or estimated.

It is worth to note that the choice of nonlinear differential equation and parameter priors does not influence the flow of analysis. Our scheme stands for a general framework and the detailed parameters can be easily customized to fit different scenarios. There are various mathematical models for system kinetics, such as [39, 40, 106]. The kinetics in Equation (5.1) is chosen as our dynamic model because it possess the property of saturation, a key idea of Michaelis-Menten kinetics [106]. Besides, it is fairly simple and it also takes account of most other biological properties. Therefore, in the simulation of the real data set, we are assuming the proposed kinetics is true.

C. Inference Method

Consider a system composed of n genes indexed by $\{1, 2, \cdots, n\}$. ChIP-chip experiments can be conducted to examine whether gene i's corresponding protein binds gene j's regulatory region. Usually this regulatory sequence is a promoter region which is located within 600 base pairs upstream of the coding region of gene j. The experimental results are represented in terms of p-values. In the first step, it is necessary to translate the p-value p into the probability of existence of a regulation relationship from gene i to gene j, which is denoted as $\mathcal{P}(i \rightarrow j|p)$. This probability will act as the prior knowledge to integrate gene expression data.

1. Incorporating ChIP-chip Data

The p-value is within the range of $[0, 1]$. After studying the properties of the microarray data, Allison proposed to exploit mixed Beta distribution to model the p-value [107]. If the transcription factor i regulates gene j, it is assumed that the ChIP-chip experiment produces a p-value p which conforms to a Beta distribution with parameters (ϕ, ζ),

$$f(p|i \rightarrow j) = \frac{p^{\phi-1}(1-p)^{\zeta-1}}{B(\phi, \zeta)}, \tag{5.3}$$

where $f(\cdot)$ stands for the probability density function and $B(\cdot, \cdot)$ represents the Beta function. On the other hand, if i does not regulate j, the p-value assumes a different Beta distribution with parameters (ψ, ξ):

$$f(p|i \nrightarrow j) = \frac{p^{\psi-1}(1-p)^{\xi-1}}{B(\psi, \xi)}. \tag{5.4}$$

Based on the knowledge provided by established and verified genetic networks, one can infer a prior knowledge about the probability of connectivity between arbitrary genes, denoted as $\eta(i \rightarrow j), \forall i, j$. Such statistics regarding the network connectivity can be found in the open literature, e.g., the data sets for yeast [51], and Drosophila [50]. By applying Bayes theorem, we obtain

$$\mathcal{P}(i \rightarrow j|p) = \frac{\eta B(\psi,\xi) p^{\phi-1}(1-p)^{\zeta-1}}{\eta B(\psi,\xi) p^{\phi-1}(1-p)^{\zeta-1} + (1-\eta)B(\phi,\zeta)p^{\psi-1}(1-p)^{\xi-1}}. \tag{5.5}$$

For simplicity, a uniform distribution can be alternatively employed to account for the p-value when $i \nrightarrow j$. In this case $\psi = 1, \xi = 1$ and (5.5) takes the form:

$$\mathcal{P}(i \rightarrow j|p) = \frac{\eta p^{\phi-1}(1-p)^{\zeta-1}}{\eta p^{\phi-1}(1-p)^{\zeta-1} + (1-\eta)B(\phi,\zeta)}. \tag{5.6}$$

The determination of ϕ and ζ depends on the experimental knowledge of the accuracy of selecting p-value thresholds. In the first step, a p-value threshold p_t is imposed, then the validity of all bindings with p-values less than p_t is corroborated by biological experiments. In this way, we can gain knowledge of the probability $\mathcal{P}(i \rightarrow j|p < p_t)$, which can be written in the form of

$$\begin{aligned}
\mathcal{P}(i \rightarrow j|p < p_t) &= \frac{\eta \mathcal{P}(p < p_t|i \rightarrow j)}{\eta \mathcal{P}(p < p_t|i \rightarrow j) + (1-\eta)\mathcal{P}(p < p_t|i \nrightarrow j)} \\
&= \frac{\eta \int_0^{p_t} p^{\phi-1}(1-p)^{\zeta-1}dp}{\eta \int_0^{p_t} p^{\phi-1}(1-p)^{\zeta-1}dp + p_t(1-\eta)B(\phi,\zeta)}.
\end{aligned}$$

Some works in the literature, e.g., [31], have made the observation that at a p-value threshold of 0.001, the frequency of false positives is 6%-10%, i.e., $\mathcal{P}(i \nrightarrow j|p < p_t) \in [6\%, 10\%]$. Taking into account these special points, we can determine the pair (ϕ, ζ) in a small range. In our case $\phi \approx 0.1$ and $\zeta \approx 100$. Finally, a table can be set up to map the p-value into the edge existence probability, which can be computed only once. It is an overhead for the computational system but it does not assume much computational resource in the runtime.

2. Exploiting Steady State Gene Expression Data

Assume that m observations of expression vector are obtained and stored in matrix $D^{n\times m}$. Next, we develop a computational approach to establish the posterior probability of the regulation $i \rightarrow j$, i.e., the probability of the existence of the edge $i \rightarrow j$, which is represented by $\mathcal{P}(i \rightarrow j|D,p)$. This posterior can be obtained through integration over the whole parental gene set and parameter space for gene j:

$$\begin{aligned}\mathcal{P}(i \rightarrow j|D,p) &= \sum_{\Pi_j}\int_{\Theta_j} f(i \rightarrow j, \Pi_j, \Theta_j|D,p)d\Theta_j \\ &= \sum_{\Pi_j}\int_{\Theta_j} \mathbf{1}_{\Pi_j}(i) f(\Pi_j,\Theta_j|D,p)d\Theta_j, \qquad (5.7)\end{aligned}$$

where the function $\mathbf{1}_{\Pi_j}(i)$ is the indicator function, which takes 1 if $i \in \Pi_j$ and 0 otherwise. Applying Bayes theorem, $f(\Pi_j,\Theta_j|D,p)$ can be expressed as

$$\begin{aligned}f(\Pi_j,\Theta_j|D,p) &= \frac{f(D|\Pi_j,\Theta_j,p)f(\Pi_j,\Theta_j|p)}{f(D|p)} \\ &= \frac{f(D|\Pi_j,\Theta_j)f(\Pi_j,\Theta_j|p)}{f(D)} \\ &= \frac{f(D|\Pi_j,\Theta_j)f(\Pi_j,\Theta_j|p)}{\sum_{\Pi_j}\int_{\Theta_j} f(D|\Pi_j,\Theta_j)f(\Pi_j,\Theta_j|p)d\Theta_j} \\ &= \frac{f(D_j|D_{\bar{j}},\Pi_j,\Theta_j)f(\Pi_j,\Theta_j|p)}{\sum_{\Pi_j}\int_{\Theta_j} f(D_j|D_{\bar{j}},\Pi_j,\Theta_j)f(\Pi_j,\Theta_j|p)d\Theta_j}, \qquad (5.8)\end{aligned}$$

where D_j denotes the observations of gene X_j, and $D_{\bar{j}}$ represents the collection of all the observations pertaining to all genes excluding those of gene X_j. $f(\Pi_j,\Theta_j|p)$ denotes the probability density of the high-dimensional parental model given the observation of ChIP-chip data. $f(D_j|D_{\bar{j}},\Pi_j,\Theta_j)$ stands for the gene expression likelihood given the parental values and the graphical model. It is a Gaussian distribution with known variance and mean determined by the first part of (5.2). The second equality in (5.8) holds because we believe the ChIP-chip experiment and steady state gene expression measurements are independent. By plugging (5.8) into (5.7), it can be inferred that

$$\mathcal{P}(i \to j|D,p) \;=\; \frac{\sum_{\Pi_j}\int_{\Theta_j}\mathbf{1}_{\Pi_j}(i)f(D_j|D_{\bar{j}},\Pi_j,\Theta_j)f(\Pi_j,\Theta_j|p)d\Theta_j}{\sum_{\Pi_j}\int_{\Theta_j}f(D_j|D_{\bar{j}},\Pi_j,\Theta_j)f(\Pi_j,\Theta_j|p)d\Theta_j}. \tag{5.9}$$

The integrations at the numerator and denominator of Equation (5.9) can not be generally performed in closed-form expression. However, the Monte Carlo methods enable to numerically evaluate the posterior probabilities. We can generate Monte Carlo samples based on the model probability density $f(\Pi,\Theta|p)$ and the integration can be obtained by averaging over these samples. Then the posterior probabilities can be estimated by

$$\mathcal{P}(i \to j|D,p) \approx \frac{\sum_{\Pi_j,\Theta_j}\mathbf{1}_{\Pi_j}(i)f(D_j|D_{\bar{j}},\Pi_j,\Theta_j)}{\sum_{\Pi_j,\Theta_j}f(D_j|D_{\bar{j}},\Pi_j,\Theta_j)}. \tag{5.10}$$

Assuming that the selection of a parent as an activator is performed in an independent manner, and that the selection of the regulation factor value is also performed independently, the model probability density $f(\Pi,\Theta|p)$ can be further expanded by using the chain rule:

$$\begin{aligned} f(\Pi,\Theta|p) &= f(\Theta|\Pi)\mathcal{P}(\Pi|p) \\ &= \prod_{i=1}^{|A|}[\rho f(\alpha_i)]\prod_{j=1}^{|R|}[(1-\rho)f(\gamma_j)]\mathcal{P}(\Pi|p). \end{aligned} \tag{5.11}$$

Equation (5.11) conveys the idea that the random samples of graphical models can be sequentially created and processed. First the network structure is created based on the binding probability of gene regulation obtained in the ChIP-chip experiment, then each parent is randomly assigned to represent an activator or repressor, and finally regulation factors are generated.

3. Algorithm Formulation

Our computational procedure can be briefly formulated in terms of the Algorithm 4, where the Matlab coding convention is used to write the pseudo-code. There exist n genes in the system. An $n \times n$ matrix is created to represent the p-values produced in the ChIP-chip experiment. We collect m steady state gene expression samples. The output entry C_{ij} stands for $\mathcal{P}(i \rightarrow j|D, p)$, and M denotes the number of Monte-Carlo iterations. Lines 1 and 2 deal with the ChIP-chip experimental data and translate p-values into the binding probabilities by using (5.5). The results are stored in matrix B. Lines 3 and 4 perform the preprocessing of the gene expression data. Let $y_{(1)}$, $y_{(2)}$, $y_{(3)}$, $\cdots$, $y_{(m-2)}$, $y_{(m-1)}$, $y_{(m)}$ be the values of a specific gene expression in ascending order. The smallest two values, $y_{(1)}, y_{(2)}$, and the largest two values, $y_{(m-1)}, y_{(m)}$, are treated as outliers and discarded. The dynamic range is defined as $Range = y_{(m-2)} - y_{(3)}$. The gene expressions are normalized as follows: the smallest two samples are assigned the null value and the largest two samples are assigned the unit value; the intermediary samples $y_{(i)}$ are normalized as $(y_{(i)} - y_{(3)})/Range$; if there is a missing sample, it is recovered through interpolation by gene's mean expression. Lines 12 through 16 implement the numerator of (5.10), and Line 17 computes the denominator of (5.10).

The algorithm can be easily reorganized into a parallel form so that we can exploit efficiently the distributed computational resources. The entries of output matrix C represent the posterior probabilities of regulation relationships between any two genes. It is directional (asymmetrical), and it possesses a clear probabilistic meaning compared with other vague connectivity metrics, e.g., mutual information. It grants the biologists the flexibility first to examine the most significant interactions, then to proceed with less evidenced edges. Therefore, it is advantageous relative to a

```
Input ChIP-chip data set $p^{n \times n}$;
Translate p-values to construct the binding probability matrix $B^{n \times n}$.
Input gene expression data set $D^{n \times m}$;
Normalize the expression data so that each expression is within the range of $[0, 1]$;
Initialize $n, L = 0^{1 \times n}, C = 0^{n \times n}$; for $k = 1$ to $M$ do
    Randomly create a directed graph and the adjacency matrix $J$ based on $B$;
    for $i = 1$ to $n$ do
        For gene $i$'s parents specified in $J(:, i)$, randomly assign them to be
        activators or repressers;
        For each parent, randomly create their regulation factor $\alpha$ or $\gamma$;
        $l \Leftarrow$ likelihood$(D_i | D_{\bar{i}}, \Pi_i, \Theta_i)$;
        for $j = 1$ to $n$ do
            if $j \in \Pi_i$ then
                $C_{j,i} = C_{j,i} + l$;
            end
        end
        $L(i) = L(i) + l$;
    end
end
$\forall i, j, C_{j,i} = C_{j,i}/L_i$; Return $C$.
```

Algorithm 4: Inference of Connectivity Significance

purely dichotomous scheme, in which genes are treated as being either connected or disconnected. A probability threshold can be imposed to change the algorithm into a dichotomous classifier. Since the posterior probability has a universal meaning, this threshold can be easily selected, usually within the range of [0.3,0.9]. A trade-off has also to be made for different performance metrics.

D. Simulation Results

The simulation consists of two parts. In the first part artificial networks are created and the performance of the proposed algorithm is compared with other representative algorithms available in the literature, namely the relevance network (RN) method [9], Chow-Liu algorithm [15] and ARACNE [16]. In the second part the algorithm is tested on the real Saccharomyces cerevisiae (budding yeast) data set and a biologically meaningful genetic network is inferred for the genetic response to environmental changes.

1. Simulation on Artificial Networks

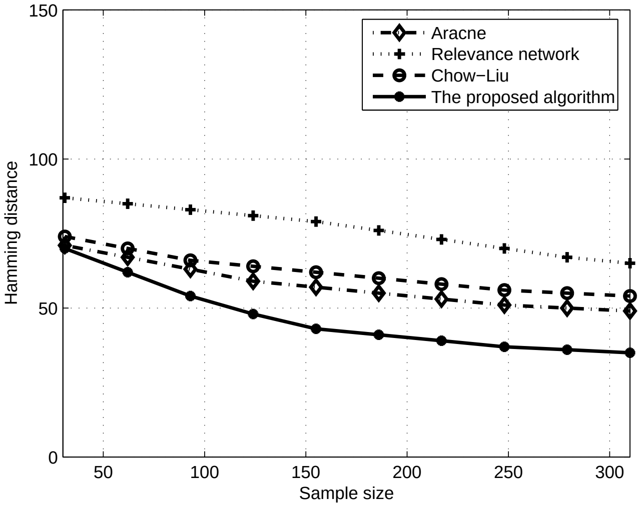
Based on gene expression data alone, the proposed algorithm is compared with other three algorithms, i.e., Relevance network [9], ARACNE [16] and Chow-Liu [15]. Because RN, Chow-Liu and ARACNE algorithms all construct undirected graphs, we have to disregard the orientation information inferred by the proposed algorithm. The synthetic and inferred graphs are represented by $G(V, E)$ and $\hat{G}(V, \hat{E})$ respectively. The two graphs share the same set of vertices but differ in the set of edges.

a. Simulation on the Proposed Kinetics

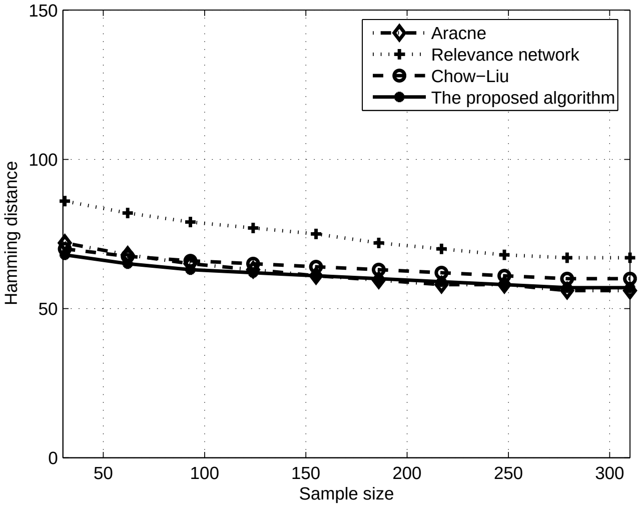
A set of artificial networks are created based on the system dynamic Equation (5.1). Each Network has 30 vertices and 60 oriented edges. Such a network scale is selected for the consideration of the computational resources and the biological network that we are going to infer. The steady state data are sampled by emulating the gene knockout experiment. A gene's expression is mandatorily forced to 0 while all other genes are free to change their expressions. The initial values of the system are randomly generated. When the system converges to the equilibrium, a Gaussian noise $N(0, 0.03)$ is added and a few samples are obtained. All genes are shut down one by one. An extra *in silico* experiment is performed and no genes are shut down. These samples correspond to the wild type strain.

Different numbers of steady-state samples were generated based on the adopted system kinetics. The transcription factor is assumed to be an activator or repressor with equal probability, i.e., $\rho = 0.5$. The baseline parameter $\delta = 0.5$ and the Gamma parameters of regulation factors are $(\kappa = 16, \beta = 0.0625)$ so that the regulation factor has a unit mean. Chow-Liu algorithm creates a spanning tree; therefore, it preserves only 29 edges, while the original synthetic network possesses 30 vertices and 60 edges. In order to make comparisons, we tune the parameters for the other three schemes so that the number of inferred edges is around 30. For the RN method, we keep the 30 edges with the highest mutual information. For ARACNE, the mutual information threshold is adjusted. In our proposed algorithm, the posterior probability thresholds are changed in the range of [0.3,0.9] so that approximately 30 edges are obtained. It has to be noted that RN, ARACNE and Chow-Liu algorithms only preserve interactions but disregard the interaction orientation. Therefore, in order to make consistent comparisons, we have to sacrifice the orientation information offered by the proposed algorithm. Besides, these three schemes have no capability of processing ChIP-chip data. Therefore, we have to configure the proposed algorithm such that any two nodes are associated with a small prior probability of connection (0.1). This reflects the fact that the connection between two arbitrary nodes in the graph is very unlikely, but not impossible. This also exemplifies how the algorithm works in the absence of the ChIP-chip data.

Fig.13(a) compares the performance in terms of Hamming distance for the four schemes assuming different sample sizes. The proposed method provides much better inference accuracy because it achieves the lowest Hamming distance. Larger sample size rewards a better inference precision. Chow-Liu's algorithm and ARACNE do not perform well. This can be attributed to the assumption of the network. Our synthetic networks actually are cyclic networks in order to reflect the real world scenario. However, cycles in the network ruin the inference precisions of Chow-Liu and ARACNE. Fig.14(a) illustrates the impact of sample size on the sensitivity. The proposed scheme outperforms the other three schemes. The sensitivities of all algorithms are less than 0.5. This is mainly due to the constraint that we pose on the number

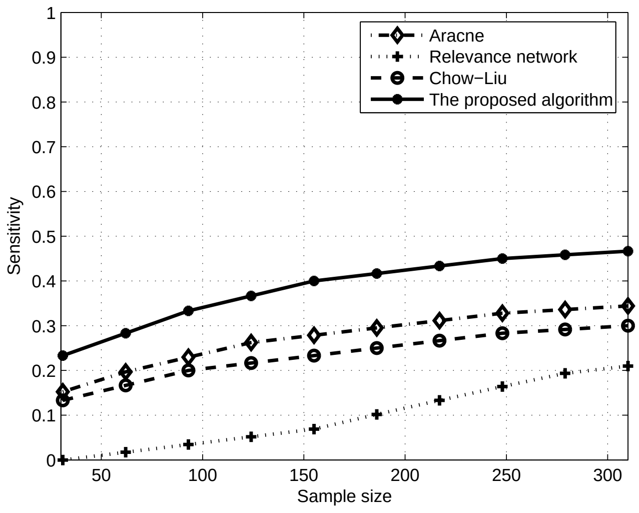


(a) Hamming distance

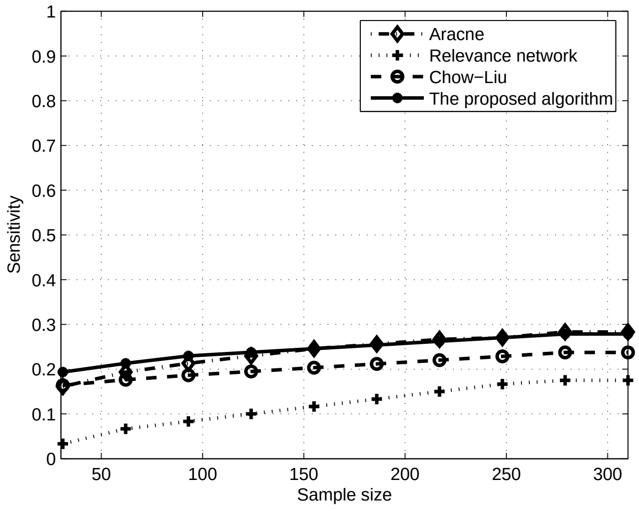


(b) Hamming distance

Fig. 13. Performance comparison in terms of Hamming distance. (a) illustrates results based on the same kinetics model employed in both data synthesization and network inference, while (b) represents results based on different kinetics models employed in the simulation process.

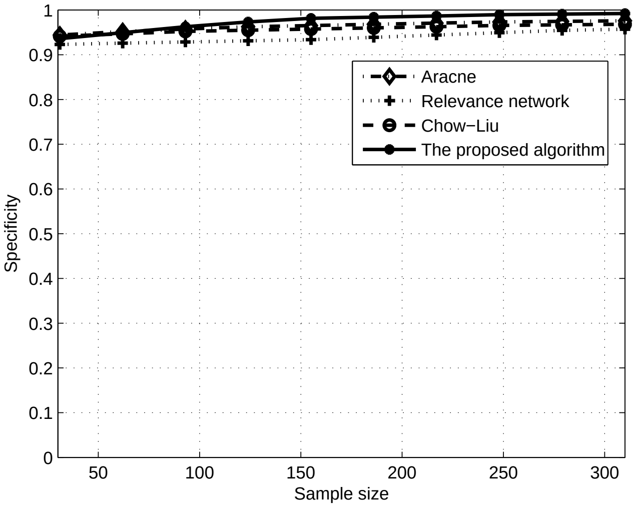


(a) sensitivity

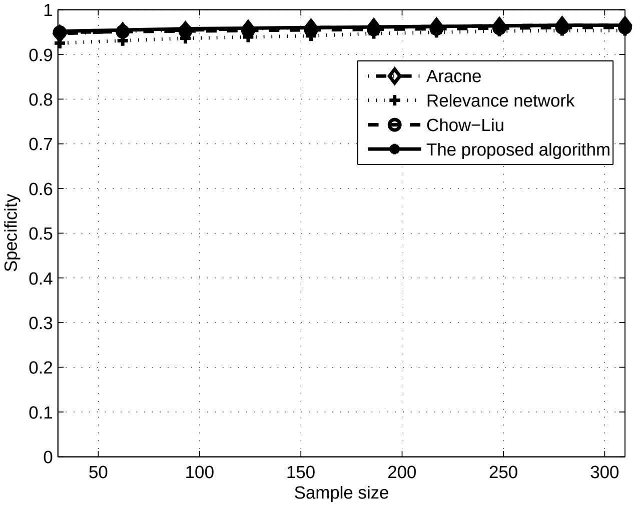


(b) sensitivity

Fig. 14. Performance comparison in terms of sensitivity. (a) illustrates results based on the same kinetics model employed in both data synthesization and network inference, while (b) represents results based on different kinetics models employed in the simulation process.



(a) specificity



(b) specificity

Fig. 15. Performance comparison in terms of specificity. (a) illustrates results based on the same kinetics model employed in both data synthesization and network inference, while (b) represents results based on different kinetics models employed in the simulation process.

of inferred edges, i.e., 30 edges. If we relax the posterior probability threshold, the sensitivity will be improved by sacrificing the specificity. Fig.15(a) depicts specificity for all schemes. All of them have high specificities, which are all greater than 0.90. The proposed scheme still exceeds. This high specificity is mainly due to the stringent constraint posed on the number of inferred edges. When considering the orientation of the edges, we find that 90% true positives inferred by the proposed algorithm are actually oriented correctly. This represents a big advantage of the proposed algorithm compared with the other three schemes.

b. Robustness of Inference

In the previous simulations, the proposed inference algorithm assumes the system dynamic as depicted by Equation (5.1). Actually, for different biological processes, there exist various mathematical models which achieve trade-offs between the sophistication of the underlying molecular reaction and the simplification of the formula description (see [40], [106] for model comparisons). Savageau [39] proposed an alternative mathematical model to account for the gene control and various forms of coupling among elementary gene circuits. This model can be denoted as

$$\frac{dx}{dt} = \lambda_A \prod_{i=1}^{|A|} a_i^{\alpha_i} - \lambda_R \prod_{j=1}^{|R|} r_j^{\gamma_j}, \tag{5.12}$$

where the two new symbols λ_A and λ_R stand for the activation and degradation coefficients, respectively, and all other symbols share the same meanings as in the Equation (5.1).

Although the proposed inference framework can "plug and play" with different models, it is still necessary to examine its robustness against the underlying model. We evaluate this model dependence by following steps: configure the model as Equation (5.12) and create a set of synthetic data, then apply the proposed algorithm based on the dynamic Equation (5.1), finally determine the performance metrics for different algorithms and compare the results with those in the previous section.

The simulation results are plotted in Figs. 13(b), 14(b) and 15(b). Each figure corresponds to a different performance metric. All algorithms exhibit different values for performance values. This shows that the inference is dependent on the particular data set and their underlying model. Compared with other three schemes, the proposed algorithm still achieves good performance in terms of the three metrics. However, the advantage of the proposed algorithm is not significant. ARACNE, Chow-Liu and the relevance method performances do not degenerate much. This is attributed mainly to the non-parametric nature of these three schemes. The persistent good performance of the proposed algorithm is due to the fact that both dynamic models have to convey the basic properties of the gene interaction kinetics, such as the activation and repression effects and the coupling of the circuitry.

2. Simulation on Saccharomyces Cerevisiae Data Sets

Saccharomyces cerevisiae (yeast) has been extensively studied in the literature of molecular biology because it is a unicellular eukaryotic organism, which shares similar cell structure with plants and animals. Also, yeast presents a short life cycle, which makes the experiments to be easily conducted. Lee [31] performed the ChIP-chip experiment, in which 141 transcription factors were tested for binding the inter-genetic regions corresponding to 6270 genes. The gene expression data were published by Mnaimneh [108], who created promoter shut-off strains for 2/3 of all essential genes. The data set contains 215 steady state cDNA microarray samples. The model parameters are assumed the same as artificial networks.

The intracellular signalling pathway in response to environmental changes has been conserved through evolution. Therefore, a study of this biological subsystem on the saccharomyces cerevisiae might help to decipher the cell survival mechanism of other organisms. We select 30 genes which are annotated to participate in the stress response process. The given ChIP-chip experiment did not provide full prior knowledge between any two genes (nodes in the graph). We believe among these genes, there are some genes whose protein products may also serve as transcription factors. Therefore, if the binding between two genes was not tested in the ChIP-chip experiment, a small probability value 0.1 is assigned as the prior knowledge. The proposed inference algorithm leads to the genetic network illustrated in Fig. 16.

The inferred genetic regulatory network shows strong proneness toward a scale-free network instead of a random network. Some genes possess especially high degree of connectivity. Three hub genes $CIN5$, $HSF1$, $MSN4$ already connect with more than 60% of all selected genes. Each of them has a connectivity degree no less than 8 while on average each gene in the network is connected with no more than 4 genes. These hub genes constitute the backbone of the network and they are potential control targets. This scale-free property is advantageous in maintaining the system robustness because a failure in one subsystem will not be propagated to the whole body.

Multiple works, e.g., [109], have identified $MSN4$ and $MSN2$ as two of the most important genes in the response to environmental changes. A recent work [110] recognized the functionality of another crucial gene $HSF1$, which is a heat shock transcription factor and functions in a different domain than the one corresponding to $MSN2/4$. Our inferred network corroborates this experimental result by showing that $HSF1$ and $MSN2/4$ regulate different set of genes except a weak connectivity between $HSF1$ and $MSN4$. $MSN2/4$ are not conserved in humans, while HSF genes have been preserved for various organisms such as Drosophila melanogaster,

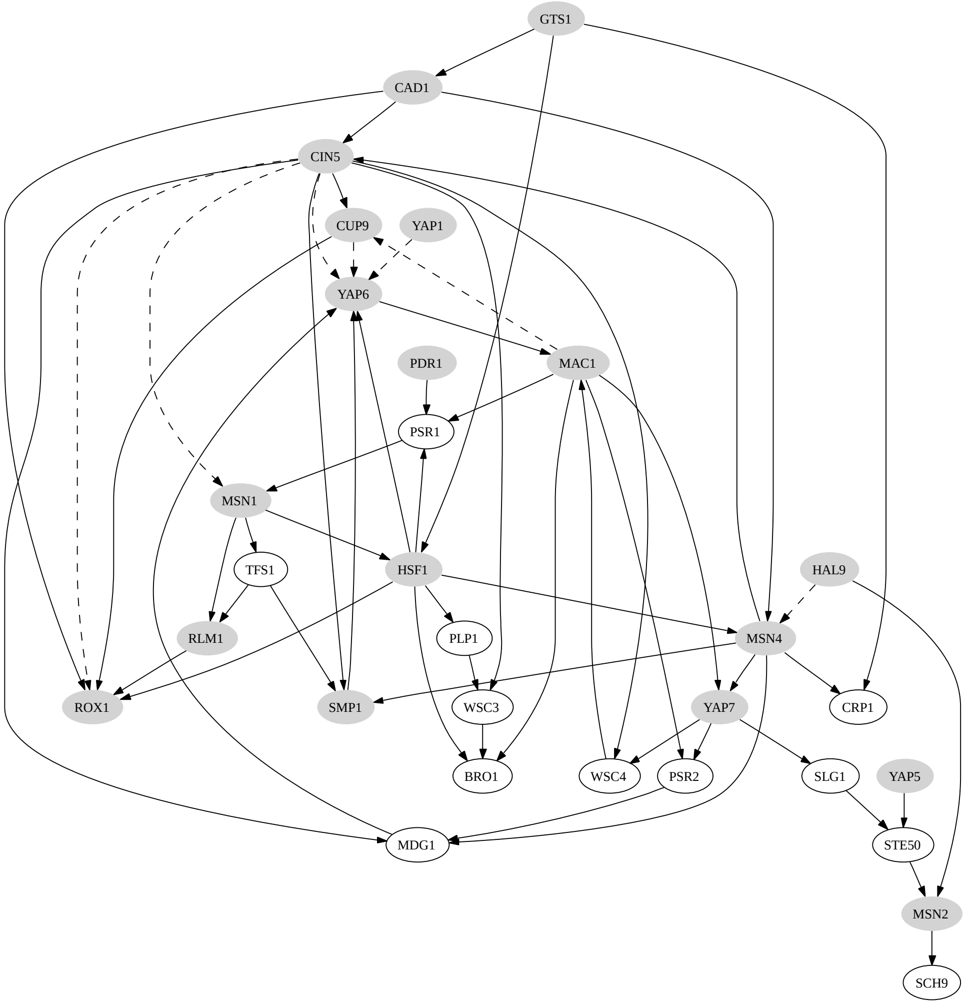


Fig. 16. Recovered genetic regulatory network for yeast stress response. The Monte Carlo iterations are 1,000,000. Dashed edges represent interactions preserved by using ChIP-chip data alone under the p-value threshold 0.001. Shadowed vertices are transcription factors tested in the ChIP-chip experiment.

chickens and mammals. Therefore, a study of the $HSF1$ pathway opens up the possibility of understanding the mechanism that governs the survival of normal cells under austere conditions.

$CIN5$ ($YAP4$) and $YAP6$ are two genes that play key roles in controlling the resistance to drugs, e.g., cisplatin [111]. $CAD1$ ($YAP2$), $CIN5$, $YAP1$ and $YAP6$ share a structure motif called basic leucine zipper ($bZIP$) and they are located closely in the network. However, they are not neighboring the other two $bZIP$ genes: $YAP5$ and $YAP7$. It is hypothesized that although they have similar molecular structures, their biological functionalities are in distinct domains.

Several edges, discovered by imposing a stringent p-value threshold 0.001 to the location data, were persevered in our inferred network. These connections constitute a small portion of the proposed network, and they are $CIN5 \rightarrow MSN1$, $CIN5 \rightarrow YAP6$, $CIN5 \rightarrow ROX1$, $YAP1 \rightarrow YAP6$, $MAC1 \rightarrow CUP9$, $CUP9 \rightarrow YAP6$ and $HAL9 \rightarrow MSN4$. Various evidences are found to corroborate the recovered interactions, which can not be obtained by employing a stringent p-value for the location data. For example, $YAP5$ is recovered to directly regulate $STE50$. This regulation relationship has also been reported in [112]. The relationship between $MSN2$ and $SCH9$ is studied in [113] in the context of extending the life span.

It is worthwhile to note that gene expression data mainly provide statistical relationships among genes, while location data offer physical binding interactions at the molecular level. By combining the two data sources, we are aiming to refine the inferred network to be biologically more meaningful. However, it also runs at a risk of confusing statistical regulatory relationships with real binding interactions. When such a case occurs, the proposed algorithm is capable of constraining the interacting genes within the same biological process and common functional relationships. A related discussion about the meaning of inferred network can also be found in [85].

CHAPTER VI

MISCELLANEOUS APPLICATIONS AND DISCUSSIONS *

A. Integration of Sequence Knowledge

The high-throughput cDNA microarray technology has enabled the simultaneous measurement of the mRNA concentrations. At the same time, multiple genome sequencing projects have been accomplished for such organisms as E. coli, yeast, fruit fly and human. The inferred networks share a problem of explaining the recovered interactions. That is, the interaction only has a statistical meaning but might have no biological justifications. Concepts are easily confusing, such as co-regulation, co-expression, direct regulation, indirect regulation. On the other hand, finding a binding site on target gene's regulatory region do not guarantee a transcription relationship since this occurrence may happen exactly by chance. Taking into account the gene expression, DNA sequence and binding site information together sheds new light on making biologically more solid inferences.

In this chapter, the genetic regulatory network is inferred based on the integration of these data sources, which helps to improve both specificity and sensitivity of the inference. The transcription factors of a target gene are determined by applying the reversible jump Markov chain Monte-Carlo (RJMCMC) algorithm to the linear regression model. The scheme is simulated on yeast data and the results provide some insight into the regulation mechanism associated with environmental changes.

*Part of the data reported in this chapter is reprinted with permission from "Recovering Genetic Regulatory Networks from Multiple Data Sources," by W. Zhao, E. Serpedin and E. R. Dougherty, 2007, in *Proceeding of 5th IEEE International Workshop on Genomic Signal Processing and Statistics (GENSIPS)*, Tuusula, Finland, Copyright 2007 by IEEE.

Our scheme is different from other schemes in that it integrates multiple sources of knowledge in a specific way, poses no constraints on the network topology and data quantification, and works well on observational data instead of controlled experiment data.

1. Model Formulation

General directed graphs G (with possibly cycles) will serve as our structural model since they are consistent with the features exhibited by biological systems, in which loops account for system redundancy and stability. Given G, the parent set Π_X is specified for any gene X, and the parameters for the system transcription model are defined in Θ.

In a system with n genes indexed by $\{1, 2, \cdots, n\}$, m observations of expression vector are obtained and stored in matrix $D^{n \times m}$. If gene i regulates gene j directly, we assume such regulation take effects by gene i's protein binding to its characteristic binding site on gene j's regulatory region. All genes' regulatory regions are stored in array $S^{n \times 1}$ and each entry of the array is a sequence with letters selected from the alphabet $\{A, T, C, G\}$ corresponding to four nucleotides. Generally a gene's regulatory region is located within L (e.g., L=600) bases upstream of the open reading frame (ORF). The binding site of the transcription factor (TF) i is denoted by B_i and contains around 5-12 bases. Consequently, gene j's expression is controlled by its TFs Π_j. Our aim is to establish the posterior probability of the regulation from gene i to gene j, which is represented by $p(i \rightarrow j | D, S, B)$.

A linear model is adopted herein to represent the relationship between the target gene j and its TFs Π_j:

$$D_{jl} = \beta_{0j} + \sum_{i \in \Pi_j} (D_{il} \cdot K_{ij} \cdot \beta_{ij}) + \epsilon. \tag{6.1}$$

where $D_{\cdot l}$ denotes the gene expression of sample l, β_{0j} is the base transcription level without any TF, ϵ is the measurement noise and can be modeled by $N(0, \sigma^2)$, K_{ij} counts how many times the binding site of the TF i occurs on the regulatory region of gene j. K_{ij} is a function of the regulatory sequence of gene j, i.e., S_j, and the transcription binding site of TF i, i.e., B_i. The K_{ij} can be represented as follows:

$$K_{ij} = \sum_{l=1}^{L-|B_i|+1} \mathbf{1}_{\{B_i\}}(S_{j,l:l+|B_i|-1}), \tag{6.2}$$

where $S_{j,l:l+|B_i|-1}$ stands for bases present on the sequence S_j from position l to position $l + |B_i| - 1$. $\mathbf{1}_A(x)$ is the indicator function which takes 1 if $x \in A$, and 0 otherwise.

Similar linear models has been employed by various works, e.g., [34]. As an enhancement of these linear models, we incorporate the TFs' expression with the binding site since our data are coming from observational experiments instead of controlled experiments. This linear model matches the intuition that the TFs controls the targets' expression through the binding sites with appropriate TF concentration in vivo.

By substituting X^l_{ij} for $D_{jl} \cdot K_{ij}$ and denoting $X^l_{0j} = 1$, Equation (6.1) can be transformed into the classical linear regression equation:

$$D_{jl} = \sum_{i \in \{0\} \cup \Pi_j} X^l_{ij} \cdot \beta_{ij} + \epsilon = X^l_{\cdot j} \beta_{\cdot j} + \epsilon. \tag{6.3}$$

Recovering the whole genetic network can be decomposed into finding TFs for each gene and that is a model selection problem. We use the reversible jump Markov chain Monte-Carlo (RJMCMC) approach to perform the a-posteriori selection between different models or candidate parent sets, as suggested by Greens [114]. The RJMCMC procedure for a specific gene can be summerized as follows:

1. At iteration t, create a new parent set $\Pi^\star|\Pi^t$ based on the proposal conditional density $g(\cdot|\Pi^t)$;

2. Create an augmenting variable $U|(\Pi^t, \Theta^t, \Pi^\star)$ from a proposal distribution $h(\cdot|\Pi^t, \Theta^t, \Pi^\star)$, define $(\Theta^\star, U^\star) = q_{t,\star}(\Theta^t, U)$, where q is an invertible one-one map and the relation $|\Pi^t| + |U| = |\Pi^\star| + |U^\star|$ holds;

3. for a proposed model $\Pi^\star$ with parameters $\Theta^\star$, the Metropolis-Hastings ratio is

$$\frac{f(\Pi^\star, \Theta^\star|y) \cdot g(\Pi^t|\Pi^\star) \cdot h(u^\star|\Pi^\star, \Theta^\star, \Pi^t)}{f(\Pi^t, \Theta^t|y) \cdot g(\Pi^\star|\Pi^t) \cdot h(u^t|\Pi^t, \Theta^t, \Pi^\star)}|J(t)|$$
$$\text{where } |J(t)| = \frac{dq_{t,\star}(\Theta, u)}{d(\Theta, u)}|_{(\Theta,u)=(\Theta^t,U)} \tag{6.4}$$

To simplify the computation of (6.4), priors can be chosen wisely as proposed in [115]. No prior bias among models is imposed hence we have $g(\Pi^\star|\Pi^t) = g(\Pi^t|\Pi^\star)$ and $g(\Pi^\star) = g(\Pi^t)$. Besides, we have $\beta_{\Pi^t} \sim N(\alpha, \sigma^2 V_{\Pi^t})$ and $\nu\lambda/\sigma^2 \sim \chi^2_\nu$, where $\alpha = (\hat{\beta}_0, 0, \cdots, 0)$, $\hat{\beta}_0 = var(D)$ and ν, λ are two hyperparameters. V_{Π^t} is a diagonal matrix with entries $(s_y^2, 1/s_1^2, ..., 1/s^2_{|\Pi^t|})$. After a series of manipulations, the Metropolis-Hastings ratio is simplified into $f(y|\Pi^\star)/f(y|\Pi^t)$, where

$$y|\Pi^t \sim t_\nu(X\alpha, (I + XVX^T)). \tag{6.5}$$

2. Simulation Results

The proposed RJMCMC is applied on the following real data sets: the microarray gene expression data reported by Gasch and Spellman [109], the sequence and biding sites information downloaded from TRANSFAC and the Saccharomyces Genome

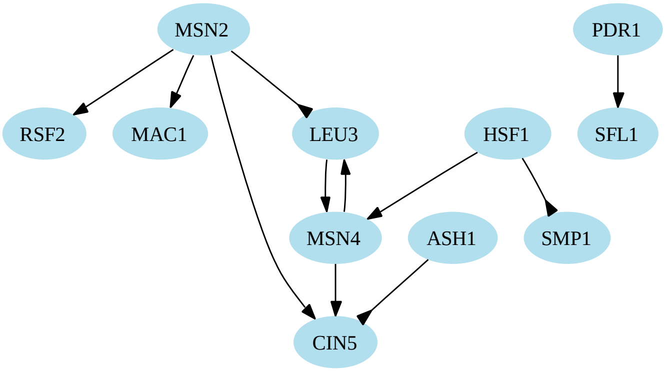


Fig. 17. Genetic regulatory network of environment-response genes. Standing alone genes are removed from the figure. The posterior threshold is chosen to be 0.5. Normal arrow head means activation while inverse arrow head stands for repression.

Database at `http://www.yeastgenome.org`. Gasch and Spellman's dataset contain 6152 genes and 137 samples. We choose a small subset of genes which are reported to respond when changes in environmental conditions occur [31]. The inferred gene regulatory network is shown in Fig. 17. The posterior threshold is set to 0.5, i.e., if the posterior connectivity probability is larger than 0.5, we assume a directed connection. Standing alone genes are removed for conciseness. We found although each gene's sequence contains at least one TF's binding site, when considering the expression data, these matches do not have a biological function and may only be inherited by chance. It shows that the techniques we employed are useful to reduce false positives in recovering gene regulatory networks.

B. Identifying Cell Cycle Genes Based on Various Knowledge

The eukaryotic cell cycle is a series of molecular-level events that lead to cell division into two daughter cells. Generally the division process presents critical formational stages known as: gap1 (G1), synthesis (S), gap2 (G2), and mitosis (M). The transcriptional events in the cell cycle can be quantitatively observed by measuring the concentration of the messenger RNA (mRNA). Based on time series microarray data, powerful approaches have been proposed to identify cell cycle genes [18] and [52]–[60]. The majority of these works deal with evenly sampled data, though the biological experiments generally output unevenly spaced measurements. To cope with this challenge, various schemes have been proposed in the signal processing literature [62]–[68]. Among them, the technically more complicated techniques, e.g., Capon and MAPES methods, aim to achieve a better spectral resolution than simpler methods, e.g., Lomb-Scargle periodogram. However, for small sample sizes, the simpler Lomb-Scargle appears to possess better performance in the presence of realistic biological data.

Most of the algorithms proposed in the literature identify the cell cycle genes by exploiting mathematical models to explain the gene's time series pattern. Employing these models and statistical tests, the periodically expressed genes are normally identified. Finally, the detected genes are compared with the genes that had been experimentally discovered to participate in the cell cycle process. Notice that these practically verified cell cycle genes only serve as a golden benchmark to evaluate the performance of the proposed identification algorithms. They are not fully exploited in the implementation of the identification algorithm. Notice also that most of the existing algorithms fail to utilize all the available information. For example, the elutriation data provided in [18] was usually discarded when performing the spectral analysis. In other experiments, some data sets were also disregarded due to either loss of synchronization or non-stationarity. Herein, we propose a novel algorithm to detect the cell cycle involved genes by integrating the gene expression analysis with the valuable prior knowledge. The prior knowledge consists of two data sets, i.e., the set of cell-cycle genes and the set of non-cell-cycle genes recognized in biological experiments. The cell-cycle genes are used to initialize the proposed algorithm and the non-cell-cycle genes are employed to control the false positives. The expression analysis is composed of the spectral estimation technique and the computation of gene expression distance. The underlying approach relies on the assumption that genes expressing similarly with cell cycle genes are also likely to be cell cycle genes. This assumption is actually exploited to apply the clustering schemes on the microarray measurements in order to partition genes into different functional groups. The proposed algorithm identifies potential cell cycle genes and guarantees that the verified cell cycle genes will be included with 100% certainty into the output gene set, and at the same time the verified non-cell-cycle genes are removed from the derived set with 100% certainty.

The proposed algorithm is composed of a spectral density analysis and a gene distance computation based on the time series microarray data. All existing spectral analysis schemes can be incorporated into the proposed algorithm. However, the Lomb-Scargle periodogram is recommended here due to its convenience of implementation and excellent performance for small sample size. The non-parametric Spearman's correlation coefficient is accepted to construct the distance measure between two genes.

1. Gene Distance Measure

A gene is identified to be a cell cycle gene if it satisfies two conditions: it passes the periodicity test which is performed on the gene expression measurements as discussed in Chapter II; or, it is within a small distance from the obtained cell cycle genes. Various distance metrics have been proposed in the clustering literature to capture the distance between genes. These include Pearson's correlation, Euclidean distance, city block distance, mutual information, etc. Because the biological observations are generally highly corrupted and the rank statistics tests usually behaves better in non-parametric environments, we accept here the Spearman's correlation coefficient as the core of our distance measure. This distance is defined for two genes X and Y between their expressions across all the available experiments:

$$d(X,Y) = 1 - |1 - \frac{6\sum_{i=1}^{n}(x_i - y_i)}{n(n^2-1)}|, \tag{6.6}$$

where $(x_i,\ y_i)$ stand for the rank pair of the measurements of genes X and Y. The parameter n counts the number of samples where both gene X and Y present available observations. This distance measure always assumes values between 0 and 1.

2. Algorithm Formulation

The proposed algorithm is formulated as the Algorithm 5. Lines 1 to 4 accept inputs and initialize the target cell cycle gene set with the spectral analysis results and the prior cell cycle genes. Lines 5 to 14 represent the iterative accumulation part. They iteratively insert into the potential cell cycle gene set the genes expressed similarly as the genes within that set. Lines 15 to 24 stand for the false positive control part. It also constructs the control set iteratively to suppress the potential false positives by using the prior knowledge. Line 25 subtracts the control set from the established target set and finalizes the cell cycle gene set. The simulation results on the yeast data set showed that the iterative accumulation part has controlled the false positives pretty well.

There are two thresholds that are to be specified. The first is the threshold for the periodicity test. Practically all genes are ranked with respect to their periodicity scores, e.g., CDC scores in [18] and maximum power spectral density, then a predetermined number of genes are conserved. Therefore, this threshold is actually a rank. This rank threshold can be determined by comparing the spectral analysis results

```
Input gene expression measurements, experimentally verified cell cycle genes (denoted as $G$) and non-cell-cycle genes (represented as $F$);
Perform power spectral analysis on gene expression data;
Perform statistical tests so that the periodically expressed genes are recognized and stored in set $C$;
$G \Leftarrow G \cup C$, $G' \Leftarrow \phi$, $F' \Leftarrow \phi$, specify the distance threshold $t$;
while $G \neq G'$ do                                  /* iterative accumulation */
    $G' \Leftarrow G$;
    for $i = 1$ to $N$ do
        for $j = 1$ to $|G|$ do                       /* $|\cdot|$ represents set size */
            if $d(x_i, g_j) < t$ then    /* $d(\cdot, \cdot)$ represents the distance between two genes */
                $G' \Leftarrow G' \cup \{x_i\}$;
            end
        end
    end
end
while $F \neq F'$ do                                  /* false positive control */
    $F' \Leftarrow F$;
    for $i = 1$ to $N$ do
        for $j = 1$ to $|F|$ do
            if $d(x_i, g_j) < t$ then
                $F' \Leftarrow F' \cup \{x_i\}$;
            end
        end
    end
end
$G \Leftarrow G - F$;
Output $G$;
```

Algorithm 5: Identifying Cell Cycle Involved Genes

with the prior knowledge. We are inclined to use a more stringent threshold, which also represents a trade-off between the number of conserved genes and the number of experimentally verified genes. The second threshold is the distance threshold. It keeps decreasing along the iteration. The initial value is assigned to be 0.25, which means high correlation by Cohen's rule of thumb [116]. Each iteration decreases this threshold by 0.05 until it reaches 0.1, then it remains constant at 0.1. This technique in practice helps to prevent the amplification of false positives.

3. Simulation Results

We still use one of the most frequently referenced time series data set published by Spellman [18]. Our prior knowledge was derived from two sources: Spellman [18] revised 104 cell cycle genes that were verified in previous biological experiments, while Lichtenberg [75] summarized 105 genes that were not involved in the cell cycle.

Spellman [18] designed a periodicity metric, namely CDC score, based on three out of four experiments. We conserved the top 400 genes with high CDC scores as the initialization set in the proposed algorithm. This means a more stringent test threshold for the spectral analysis part. The algorithm left 722 genes marked as potential cell cycle involved genes. All the detected 722 genes are hierarchically clustered in Fig. 18. The hierarchical clustering was selected mainly because it was convenient for visualization and it avoided to specify the number of desired clusters. It is worthy to note that more advanced methods, e.g., self organizing map (SOM) [117] could achieve a better clustering performance. Most clusters indicate a strong periodicity pattern, as can be discerned by the red and green regions which are positioned alternately. There is an exotic cluster, which exhibits fast oscillation in the cdc15 experiments. This cluster contains 130 genes that are illustrated in Fig.19. By examining the existing annotations for these genes, we found most of them either encode

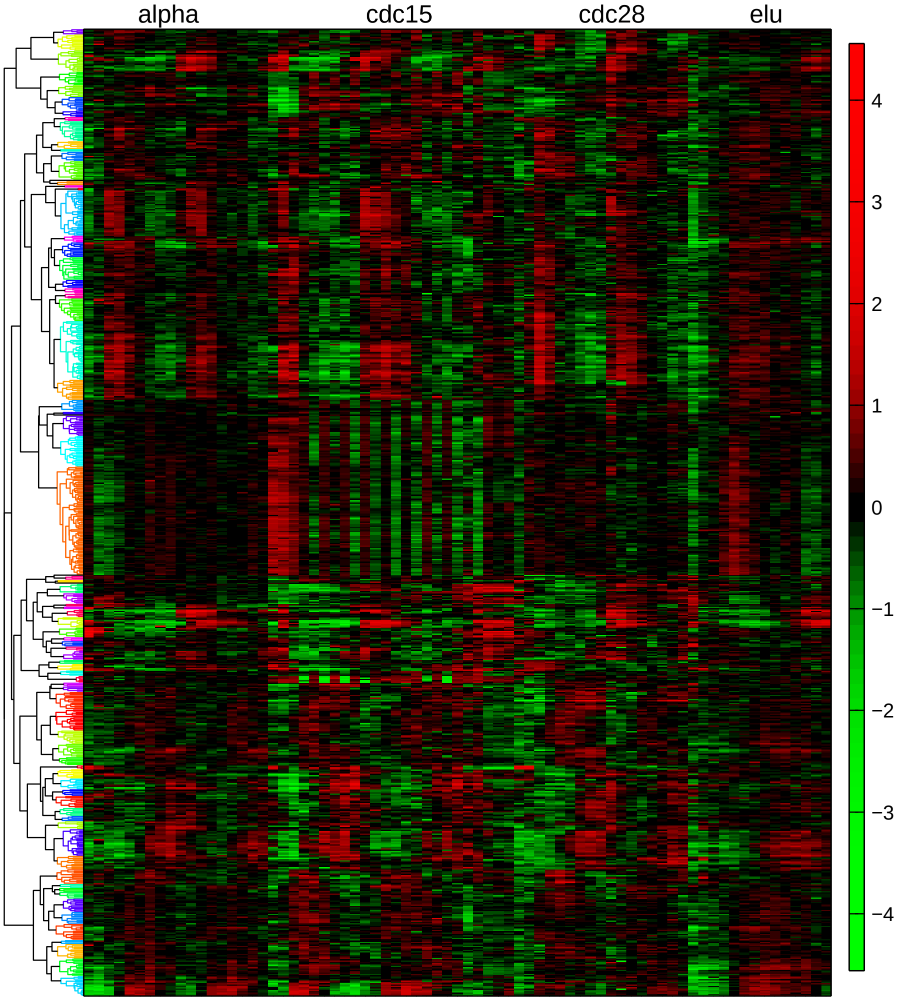


Fig. 18. Clustering analysis of identified Saccharomyces Cerevisiae genes. Gene expression levels are indicated by the heatmap. There are 722 genes identified by the proposed algorithm to participate in the cell cycle. Most genes exhibit strong periodicity, as indicated by alternately positioned red and green regions.

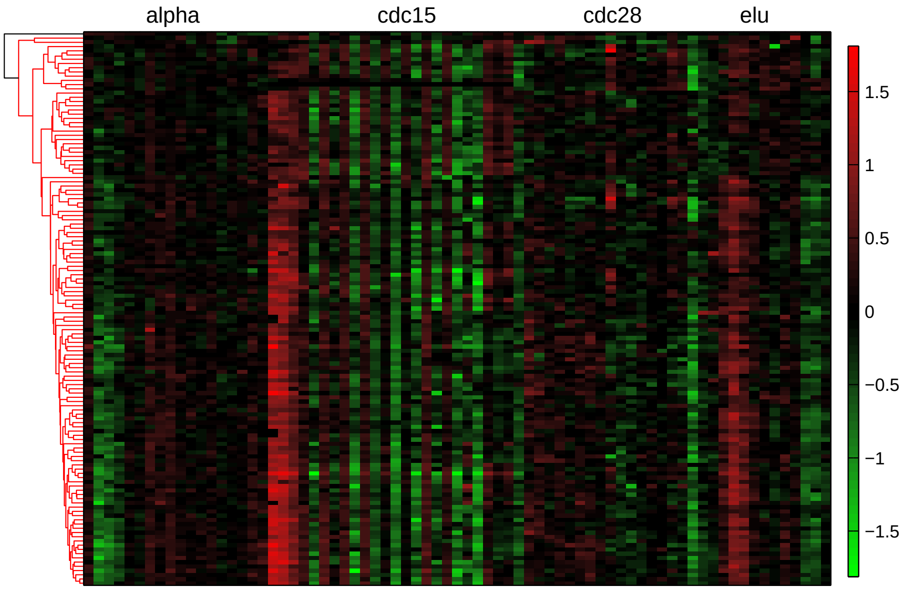


Fig. 19. The exotic clustering of identified Saccharomyces Cerevisiae genes. Gene expression levels are indicated by the heatmap. This cluster contains 130 genes. The gene expressions in the cdc15 experiment oscillate between low and high levels. Most of these genes are nucleolar genes.

nucleolar proteins or are involved in ribosome biogenesis. It has been verified that ribosome biogenesis consumes up to 80% of proliferating energy and it is linked to cell cycle in metazoan cells. However, in the yeast the ribosome biogenesis is not regulated by the cell cycle in the same manner as in advanced organisms due to the closed mitosis of the yeast [118]. Defects in nucleolar genes halt the cell at the Start checkpoint [119]. The ribosome biogenesis controls the growth of the size and inhibits the cell cycle until the cell has reached the corresponding size [120].

C. Clustering Genes Based on Spectral Information

Based on microarray measurements, clustering methods have been exploited to partition genes into subsets. Members in each subset are assumed to share specific biological function or participate in the same molecular-level process. They are termed as co-expressed genes and are supposed to be located closely in the underlying genetic regulatory networks. Eisen et al. [121] applied the hierarchical clustering to partition yeast genes, Tamayo et al. [117] exploited the self-organizing map (SOM), and Tavazoie et al. [122] employed K-means clustering to group gene expressions and then search upstream DNA sequence motifs that contribute to the co-expression of genes. Besides, Zhou et al. [123] designed a clustering strategy by minimizing the mutual information between clusters. Also, Giurcaneanu [124] exploited the minimum description length (MDL) principle to determine the number of clusters. Whether technically advanced schemes represent better solutions for real biological data is still under debate. However, usually most of the schemes provide valuable alternatives and insights to each other. Therefore, it was recommended that several clustering schemes be performed to analyze the same real data set [125] so that the difference between clusterings would capture some patterns that otherwise would be neglected by running only one method.

A straightforward application of clustering schemes will cause the loss of temporal information inherent in the time series measurements. This shortcoming was noticed by Tabus and Aastola [126], who proposed to fit the data by parametric models, depicted in terms of linear dynamic systems, and the genes in the same cluster were assumed to share common dynamics. The temporal relationships were also explored via more complex models, i.e., genetic regulatory networks, which can be constructed via more computationally-demanding algorithms, e.g., [17] and [37]. However, in general, the network inference schemes deal only with relatively small scale networks consisting of less than hundreds of genes. Genome wide analysis is beyond the computational capability of these inference algorithms. Therefore, clustering methods are usually exploited to partition genes, and the obtained subsets of genes serve as further research targets to obtain more accurate maps of the underlying biological processes.

Based on time series data, modern spectral density estimation methods have been exploited to identify periodically expressed genes, as discussed in a previous chapter. This section proposes a novel clustering preprocessing procedure which combines the power spectral density analysis with clustering schemes. Given a set of microarray measurements, the power spectral density of each gene is first computed, then the spectral information is fed into the clustering schemes. The members within the same cluster will share similar spectral information, therefore they are supposed to participate in the same temporally regulated biological process. The assumptions underlying this statement rely on the following facts: if two genes X and Y are in the same cluster, their spectral densities are very close to each other; in the time domain, their gene expressions may just differ in their phases. The phases are usually modeled to correspond to different stages of the same biological processes, e.g., cell cycle or circadian rhythms. The proposed spectral-density-based clustering actually differentiates the following two cases:

1. Gene X and Y's expressions are uncorrelated in both time and frequency domains.
2. Gene X and Y's expressions are uncorrelated in time domain, but gene X's expression is a time-shifted version of gene Y's expression.

In the traditional clustering schemes, the distances are the same for the above two cases (both assuming large values). However, in the proposed algorithm, the 2nd case is favorable and has a lower distance. Therefore, by exploiting the proposed algorithm, the genes participating in the same biological process are more likely to be grouped into the same cluster. Lomb-Scargle periodogram serves as the spectral density estimation tool since it is computationally simple and possesses higher accuracy in the presence of unevenly measured and small size gene expression data sets.

The simulation results corroborate that the proposed approach achieves a better clustering for hierarchical, K-means and self-organizing map (SOM) in most cases. Besides, it constructs a significantly different partition relative to traditional clustering strategies. When deploying the hierarchical or K-means clustering methods based on the spectral density, the Euclidean and city block distance metrics appear to be more appealing than the cosine or correlation distance metrics. The proposed preprocessing technique is valuable since it provides additional information about the temporal regulated genetic processes, e.g., cell cycle.

CHAPTER VII

CONCLUSION

This dissertation is focused on the application of statistical signal processing techniques into the emerging genomic area. The research can be categorized into three interrelated subtopics, i.e., identification of genes involving in specific processes, inference of genetic regulatory networks based on microarray measurements, either steady state or time series, and integration of heterogeneous data.

To identify specific functioning genes, particularly those in cellular cycles, three of the most representative spectral analysis methods, namely, Lomb-Scargle, Capon and missing-data amplitude and phase estimation (MAPES) methods, are compared in terms of their performance for detecting the periodically expressed genes in *Saccharomyces cerevisiae*. Our *in silico* experiments revealed that the simplest methods, in particular the Lomb-Scargle algorithm, outperforms the more sophisticated algorithms: Capon and MAPES. This discrepancy between methods is mainly attributed to the data features, such as the small sample size, large proportion of missing samples, and the presence of samples highly corrupted by noise. The computational complexity sacrificed in MAPES for achieving high resolution is not justifiable in the context of gene microarray data. In addition, a list of 149 Drosophila melanogaster genes were identified to express periodically.

The inference of the genetic regulatory network (GRN) can be performed based on time independent microarray observations. By exploiting information theoretic quantities, two algorithms together with a novel direct connectivity metric (DCM) were proposed. Simulation results show that the proposed algorithms present a satisfactory performance in the case of artificial networks. The algorithms are further applied on a realistic melanoma data set, and a 470-gene network and WNT5A pathway are recovered. The advantage of the proposed algorithms is that they not only recover the connectivity information among genes, but they also assign to each connectivity a confidence level. This provides biologists the opportunity to examine the inferred interactions starting from the most probable and valuable connectivity.

For time course microarray observations, an algorithm has been designed and implemented to reconstruct the GRN. The cross-time mutual information is employed as a metric to discern the oriented connectivity. The MDL principle is used to find the threshold for differentiating between regulation and non-regulation, and to design a network model that achieves a good trade-off between modeling complexity and data fitting accuracy. The proposed network inference algorithm is used for modeling regulatory pathways encountered in embryonic segmentation and muscle development in drosophila melanogaster. The proposed network inference algorithm is practically useful for recovering temporal regulations and can serve as an analysis tool for time series data sets.

Novel biological technology brings new data everyday. A novel algorithm is proposed to recover the GRN in the light of knowledge brought by transcriptional kinetics, ChIP-chip and gene microarray data. The analysis is based on the Bayesian methodology and Monte Carlo techniques. The proposed scheme is useful to compensate the shortcomings of utilization of only one data set alone. Our *in silico* experiments corroborate that the algorithm outperforms in specificity, sensitivity and Hamming distance relative to three state-of-the-art schemes. A budding yeast genetic regulatory network is proposed to account for the stress response.

Other applications of signal processing techniques are also proposed. These include applying the reversible jump Markov Chain Monte Carlo to incorporate sequence and binding knowledge with microarray observations, identifying cell cycle genes by combining prior experimental information, and clustering gene expressions in the frequency domain.

Through the course of the research, we have found that the difficulties come mainly from three dimensions. First, most biological experiments create small sample size, produce highly corrupted observations, and leave a large portion of crucial variables unmeasured. This demands robust and efficient stochastic analysis. Besides, schemes perform inconsistently under different circumstances. Second, biological validation remains a problem. The *in silico* results are usually mathematical significant but might not possess biological meaning. This has to be resolved via a more close cooperation with biologists and medical staffs. Third, interdisciplinary research has to be strengthened not only to incorporate different efforts in various academic areas, e.g., mathematics, statistics and engineering, but also in different domains in biology, e.g., sequence analysis, genetic network analysis, and protein structure determination.

Our research was initiated based on previous endeavors in genomic signal processing. This work represents a bridge for potential future extensions. Several other knowledge sources might be integrated into the current framework. For example, protein-protein interactions are useful to identify co-binding regulations. Protein structure knowledge can be exploited to categorize the proteins and find similar functionality. A cross-species research is also highly desirable since similar regulation mechanisms are expected to be conserved. If a gene is conserved in both humans and mice, then the knowledge of the gene's pathway in the mouse will be an excellent reference for the study of human genetic diseases. Mathematically, stochastic differential equations can be exploited to investigate the genetic kinetics in the molecular level. Various techniques, e.g., Ito integral and optimal stopping, can be applied for implementing stochastic control.

REFERENCES

[1] I. Simon and Z. Siegfried, "Combined static and dynamic analysis for determining the quality of time-series expression profiles," *Nature Biotechnology*, vol. 23, pp. 1503–1508, 2005.

[2] S.A. Kauffman, "Metabolic stability and epigenesist in randomly constructed genetic nets," *Theor. Biol.*, vol. 22, pp. 437–467, 1969.

[3] I. Shmulevich, E. R. Dougherty, and W. Zhang, "From boolean to probabilistic boolean networks as models of genetic regulatory networks," *Proceedings of IEEE*, vol. 90, pp. 1778–1792, 2002.

[4] I. Shmulevich, E. R. Dougherty, S. Kim, and W. Zhang, "Probabilistic boolean networks: A rule-based uncertainty model for gene regulatory networks," *Bioinformatics*, vol. 18, no. 2, pp. 261–274, 2002.

[5] W. Friedman, M. Linial, I. Nachman, and D. Peer, "Using Bayesian network to analyze expression data," *J. Comput. Biol*, vol. 7, pp. 601–620, 2000.

[6] P. Sebastiani, "Bayesian networks," in *The Data Mining and Knowledge Discovery Handbook*, pp. 193–230, Springer, New York, 2005.

[7] T. Dean and K. Kanazawa, "A model for reasoning about persistence and causation," *Comput. Intell.*, vol. 5, pp. 142–150, 1989.

[8] K. Murphy, "Dynamic Bayesian networks: Representation, inference and learning," Tech. report, University of California, Berkeley, Computer Science Division, 2002.

[9] A. J. Butte and I. S. Kohane, “Mutual information relevance networks: Functional genomic clustering using pairwise entropy measurements,” in *Proceedings of Pacific Symposium of Biocomputing*, Honolulu, Hawaii, January 2000, pp. 418–42.

[10] J. Pearl, *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*, San Francisco, CA: Morgan Kaufmann, 1988.

[11] H. Lahdesmaki, S. Hautaniemi, and I. Shmulevich, “Relationships between probabilistic boolean networks and dynamic Bayesian networks as models of gene regulatory networks,” *Signal Processing*, vol. 86, pp. 814–834, 2006.

[12] S. Kim, E. R. Dougherty, and M. L. Bitter, “General nonlinear framework for the analysis of gene interactions via multivariate expression arrays,” *J. Biomedical Optics*, vol. 5, pp. 411–444, 2000.

[13] X. Zhou, X. Wang, and E. R. Dougherty, “Construction of genomic networks using mutual-information clustering and reversible-jump Markov-chain-Monte-Carlo predictor design,” *Signal Processing*, vol. 83, pp. 745–761, 2003.

[14] R. Pal, I. Ivanov, A. Datta, and E. R. Dougherty, “Generating boolean networks with a prescribed attractor structure,” *Bioinformatics*, vol. 21, pp. 4021–4025, 2005.

[15] C. K. Chow and C. N. Liu, “Approximating discrete probability distributions with dependence trees,” *IEEE Transactions on Information Theory*, vol. 14, no. 3, pp. 462–467, 1968.

[16] A.A. Margolin, I. Nemanmen, K. Basso, and C. Wiggins, “Aracne: An algorithm for reconstruction of genetic networks in a mammalian cellular context,”

BMC Bioinformatics, vol. 7, pp. S7, 2006.

[17] S. Liang, S. Fuhrman, and R. Somogyi, "Reveal, a general reverse-engineering algorithm for inference of genetic network architectures," in *Proceedings of the Pacific Symposium on Biocomputing*, Honolulu, Hawaii, 1998, pp. 18–29.

[18] P. T. Spellman, G. Sherlock, M. Q. Zhang, and V. R. Iyer, "Comprehensive identification of cell cycle-regulated genes of the yeast Saccharomyces cerevisiae by microarray hybridization," *Molecular Biology of the Cell*, vol. 9, pp. 3273–3297, 1998.

[19] X. Chen, G. Anantha, and X. Wang, "An effective structure learning method for constructing gene networks," *Bioinformatics*, vol. 22, no. 11, pp. 1367–1374, 2006.

[20] K. Olesen and A. Madsen, "Maximal prime sub-graph decomposition of Bayesian networks," *IEEE Trans. Syst. Man Cybern. B.*, vol. 32, pp. 21–31, 2002.

[21] I. Beinlich, G. Suermondt, R. Chavez, and G. Cooper, "The alarm monitoring system: A case study with two probabilistic inference techniques for belief networks," in *Proceedings of 2nd European Conference on Artifical Intelligence and Medicine*, Berlin, Germany, 1989, pp. 247–256.

[22] D. Peer, A. Tanay, and A. Regev, "Minreg: A scalable algorithm for learning parsimonious regulatory networks in yeast and mammals," *The Journal of Machine Learning Research*, vol. 7, pp. 167–189, 2006.

[23] K. Murphy and S. Mia, "Modelling gene expression data using dynamic Bayesian networks," Tech. report, University of California, Berkeley, 1999.

[24] M. Zou and S. D. Conzen, “A new dynamic Bayesian network (dbn) approach for identifying gene regulatory networks from time course microarray data,” *Bioinformatics*, vol. 21, pp. 71–79, 2005.

[25] R. J. Chou, M. J. Campbell, E. A. Winzeler, L. Steinmetz, and A. Conway, “A genome-wide transcriptional analysis of the mitotic cell cycle,” *Molecular Cell*, vol. 2, pp. 65–73, 1998.

[26] S. Y. Kim, S. Imoto, and S. Miyano, “Inferring gene networks from time series microarray data using dynamic Bayesian networks,” *Briefings in Bioinformatics*, vol. 4, no. 3, pp. 228–235, 2003.

[27] “Kegg yeast cell cycle pathway,” http://www.genome.ad.jp/kegg/pathway/sce/sce04111.html; accessed May 15, 2008.

[28] J. DeRisi, V. R. Lyer, and P. O. Brown, “Exploring the metabolic and gene control of gene expression on a genomic scale,” *Science*, vol. 278, pp. 680–686, 1997.

[29] Z. Bar-Joseph, G. K. Gerber, T. I. Lee, N. J. Rinaldi, J. Y. Yoo, F. Robert, D. B. Gordon, E. Fraenkel, T. S. Jaakkola, R. A. Young, and D. K. Gifford, “Computational discovery of gene modules and regulatory networks,” *Nature Biotechnology*, vol. 21, pp. 1337–1342, 2003.

[30] A. Bernard and A. Hartemink, “Informative structure priors: Joint learning of dynamic regulatory networks from multiple types of data,” in *Pacific Symposium on Biocomputing*, Honolulu, Hawaii, 2005, pp. 459–470.

[31] T. I. Lee, N. J. Rinaldi, F. Robert, D. T. Odom, and Z. Bar-Joseph, “Transcriptional regulatory networks in Saccharomyces cerevisiae,” *Science*, vol. 298,

pp. 799–804, 2002.

[32] N. Nariai, S. Kim, S. Imoto, and S. Miyano, "Using protein-protein interactions for refining gene networks estimated from microarray data by Bayesian networks," in *Proceedings of Pacific Symposium on Biocomputing*, Honolulu, Hawaii, 2004, vol. 9, pp. 336–347.

[33] B. Xing and M. J. van der Laan, "A statistical method for constructing transcriptional regulatory networks using gene expression and sequence data," *Journal of Computational Biology*, vol. 12, no. 2, pp. 229–246, 2005.

[34] H. J. Bussemaker, H. Li, and E. D. Siggia, "Regulatory element detection using correlation with expression," *Nature Genetics*, vol. 27, pp. 167–174, 2001.

[35] E. M. Conlon, X. S. Liu, J. D. Lieb, and J. S. Liu, "Integrating regulatory motif discovery and genome-wide expression analysis," *Proc. Nat. Acad. Sci. USA*, vol. 100, pp. 3339–3344, 2003.

[36] G. R. G. Lanckriet and M. I. Jordan M. Deng, N. Cristianini, "Kernel-based data fusion and its application to protein function prediction in yeast," in *Pacific Symposium on Biocomputing*, Honolulu, Hawaii, 2004, pp. 300–311.

[37] W. Zhao, E. Serpedin, and E. R. Dougherty, "Inferring gene regulatory networks from time series data using the minimum description length principle," *Bioinformatics*, vol. 22, pp. 2129–2135, 2006.

[38] G.F. Cooper and E. Herskovits, "A Bayesian method for the induction of probabilistic networks from data," *Machine Learning*, vol. 9, pp. 309–347, 1992.

[39] M. A. Savageau, "Rules for the evolution of gene circuitry," in *Proceedings of Pacific Symposium on Biocomputing*, Honolulu, Hawaii, 1998, pp. 54–65.

[40] L. F. Wessels, E. P. van Someren, and M. J. Reinders, "A comparison of genetic network models," in *Proceedings of Pacific Symposium on Biocomputing*, Honolulu, Hawaii, 2001, pp. 508–519.

[41] M. K. S. Yeung, J. Tegner, and J. J. Collins, "Reverse engineering gene networks using singular value decomposition and robust regression," *Proc. Natl. Acad. Sci USA*, vol. 99, no. 9, pp. 6163–6168, 2002.

[42] W. Zhao, K. Agyepong, E. Serpedin, and E.R. Dougherty, "Identifying drosophila cell-cycle regulated genes from irregular microarray data," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, Las Vegas, Nevada, 2008, pp. 633–636.

[43] T.M. Cover and J.A. Thomas, *Elements of Information Theory*, New York: Wiley-Interscience, 1991.

[44] J. Rissanen, "Modeling by shortest data description," *Automatica*, vol. 14, pp. 465–471, 1978.

[45] J. Beirlant, E. Dudewicz, L. Gyorfi, and E. van der Meulen, "Nonparametric entropy estimation: An overview," *Int. J. Math. Stat. Sci.*, vol. 6, pp. 17–39, 1997.

[46] L. Paninski, "Estimation of entropy and mutual information," *Neural Comput.*, vol. 15, pp. 1191–1253, 2003.

[47] L. Paninski, "Estimating entropy on m bins given fewer than m samples," *IEEE T. Inform. Theory*, vol. 50, pp. 2200–2203, 2004.

[48] P. Mendes, W. Sha, and K. Ye, “Artificial gene networks for objective comparison of analysis algorithms,” *Bioinformatics*, vol. 19, pp. ii122–ii129, 2003.

[49] T. V. D. Bulcke, K. V. Leemput, B. Naudts, P. van Remorte, H. Ma, A. Verschoren, B. De Moor, and K. Marchal, “Syntren: A generator of synthetic gene expression data for design and analysis of structure learning algorithms,” *BMC Bioinformatics*, vol. 7, pp. 43, 2006.

[50] L. Giot, J. S. Bader, C. Brouwer, A. Chaudhuri, and B. Kuang, “A protein interaction map of drosophila melanogaster,” *Science*, vol. 302, pp. 1727–1736, 2003.

[51] N. Guelzim, S. Bottani, P. Bourgine, and F. Kepes, “Topological and causal structure of the yeast transcriptional regulatory network,” *Nature Genetics*, vol. 31, pp. 60–63, 2002.

[52] M. L. Whitfield, G. Sherlock, A. J. Saldanha, and J. I. Murray, “Identification of genes periodically expressed in the human cell cycle and their expression in tumors,” *Molecular Biology of the Cell*, vol. 13, pp. 1977–2000, 2002.

[53] M. N. Arbeitman, E. M. Furlong, F. Imam, E. Johnson, B. H. Null, B. S. Baker, M. A. Krasnow, M. P. Scott, R. W. Davis, and K. P. White, “Gene expression during the life cycle of Drosophila melanogaster,” *Science*, vol. 297, pp. 2270–227, 2003.

[54] C. D. Giurcaneanu, “Stochastic complexity for the detection of periodically expressed genes,” in *Proceedings of IEEE International Workshop on Genomic Signal processing and Statistics (GENSIPS)*, Tuusula, Finland, 2007, pp. 1–4.

[55] M. Ahdesmaki, H. Lahdesmaki M., R. Pearson, H. Huttunen, and O. Yli-Harja,

"Robust detection of periodic time series measured from biological systems," *BMC Bioinformatics*, vol. 6, no. 117, pp. 1–18, 2007.

[56] Y. Luan and H. Li, "Model-based methods for identifying periodically expressed genes based on time course microarray gene expression data," *Bioinformatics*, vol. 20, pp. 332–339, 2004.

[57] X. Lu, W. Zhang, Z. S. Qin, K. E. Kwast, and J. S. Liu, "Statistical synchronization and Bayesian detection of periodically expressed genes," *Nucleic Acids Research*, vol. 32, pp. 447–455, 2004.

[58] S. Wichert, K. Fokianos, and K. Strimmer, "Identifying periodically expressed transcripts in microarray time series data," *BMC Bioinformatics*, vol. 20, pp. 5–20, 2004.

[59] T. Bowles, A. Jakobsson, and J. Chambers, "Detection of cell-cyclic elements in mis-sampled gene expression data using a robust capon estimator," in *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Montreal, Quebec, Canada, May 2004, pp. V417–420.

[60] U. de Lichtenberg, L. J. Jensen, A. Fausboll, and T. S. Jensen, "Comparison of computational methods for the identification of cell cycle-regulated genes," *Bioinformatics*, vol. 21, pp. 1164–1171, 2005.

[61] P. Stoica and R. L. Moses, *Introduction to Spectral Analysis*, Upper Saddle River, NJ: Prentice Hall, 1997.

[62] N. R. Lomb, "Least-squares frequency analysis of unequally spaced data," *Astrophysics and Space Science*, vol. 39, pp. 447–462, 1976.

[63] J. D. Scargle, “Statistical aspects of spectral analysis of unevenly spaced data,” *The Astrophysics Journal*, vol. 263, pp. 835–853, 1982.

[64] E. F. Glynn, J. Chen, and A. R. Mushegian, “Detecting periodic patterns in unevenly spaced gene expression time series using lomb-scargle periodograms,” *Bioinformatics*, vol. 22, pp. 310–316, 2006.

[65] A. Schwarzenberg-Czerny, “On the advantage of using analysis of variance for period search,” *Monthly Notices of the Royal Astronomical Society*, vol. 241, pp. 153–165, 1989.

[66] M. Ahdesmaki, H. Lahdesmaki, A. Cracey, I. Shmulevich, and O. Yli-Harja, “Robust regression for periodicity detection in non-uniformly sampled time-course gene expression data,” *BMC Bioinformatics*, vol. 8, no. 233, pp. 1–16, 2007.

[67] P. Stoica and N. Sandgren, “Spectral analysis of irregularly-sampled data: parallelling the regularly-sampled data approaches,” *Digital Signal Processing*, vol. 16, pp. 712–734, 2006.

[68] Y. Wang, P. Stoica, J. Li, and T. L. Marzetta, “Nonparametric spectral analysis with missing data via the EM algorithm,” *Digital Signal Processing*, vol. 15, pp. 191–206, 2005.

[69] L. Eyer and P. Bartholdi, “Variable stars: Which nyquist frequency?,” *Bioinformatics*, vol. 135, pp. 1–3, 1999.

[70] J. Fan and Q. Yao, *Nonlinear Time Series: Nonparametric and Parametric Methods*, New York: Springer-Verlag, 2003.

[71] R. A. Fisher, “Tests of significance in harmonic analysis,” *Proceeding of Royal Society A*, vol. 125, pp. 54–59, 1929.

[72] P. J. Brockwell and R. A. Davis, *Time Series Theory and Methods, 2nd edition*, New York: Springer-Verlag, 1987.

[73] J. D. Storey, “A direct approach to false discovery rates,” *Journal of the Royal Statistical Society Series B*, vol. 64, pp. 479–498, 2002.

[74] J. D. Storey, “The positive false discovery rate: A Bayesian interpretation and the q-value,” *Annals of Statistics*, vol. 31, pp. 2013–2035, 2003.

[75] U. de Lichtenberg, R. Wernersson, T. S. Jensen, and H. B. Nielsen, “New weakly expressed cell cycle-regulated genes in yeast,” *Yeast*, vol. 22, pp. 1191–1201, 2005.

[76] L. T. Reiter, L. Potocki, S. Chien, M. Gribskov, and E Bier, “A systematic analysis of human disease-associated gene sequences in drosophila melanogaster,” *Genome Research*, vol. 11, no. 6, pp. 1114–1125, 2001.

[77] S. Cooper, “Rethinking synchronization of mammalian cells for cell cycle analysis,” *Cellular and Molecular Life Sciences*, vol. 60, pp. 1099–1103, 2003.

[78] I. Shmulevich and O. Yli-Harja, “Inference of genetic regulatory networks under the best-fit extension paradigm,” in *IEEE-EURASIP Workshop on Nonlinear Signal and Image Processing (NSIP-01)*, Baltimore, MD, June 2001, pp. 3–6.

[79] J. Yu, V. A. Smith, P. P. Wang, A. J. Hartemink, and E. D. Jarvis, “Using Bayesian network inference algorithms to recover molecular genetic regulatory networks,” in *proceedings of 3rd International conference on systems biology*, Stockholm, Sweden, December 2002, pp. 1.

[80] G. F. Cooper, “A simple constraint-based algorithm for efficiently mining observational databases for causal relationships,” *Data Mining and Knowledge Discovery*, vol. 1, pp. 203–224, 1997.

[81] P. Brazhnic, A. de la Fuente, and P. Mendes, “Gene networks: How to put the function in genomics,” *Trends in Biotechnology*, vol. 20, pp. 467–472, 2002.

[82] M. Gustafsson, M. Hornquist, and A. Lombardi, “Constructing and analyzing a large-scale gene-to-gene regulatory network - lasso-constrained inference and biological validation,” *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 2, no. 3, pp. 254–261, 2005.

[83] D. W. Scott, *Multivariate Density Estimation:Theory, Practice, and Visualization*, New York: John Wiley & Sons, 1992.

[84] D. Wolpert and D. Wolf, “Estimating functions of probability distributions from a finite set of samples,” *Phys. Rev. E*, vol. 52, pp. 6841–6854, 1995.

[85] A. J. Hartemink, “Reverse engineering gene regulatory networks,” *Nature Biotechnology*, vol. 23, pp. 554–555, 2005.

[86] M. Bittner, P. Meltzer, Y. Chen, Y. Jiang, E. Seftor, and J. Trent, “Molecular classification of cutaneous malignant melanoma by gene expression profiling,” *Nature*, vol. 406, pp. 536–540, 2000.

[87] A. T. Weeraratna, Y. Jiang, G. Hostetter, and K. Rosenblatt, “Wnt5a signaling directly affects cell motility and invasion of metastatic melanoma,” *Cancer Cell*, vol. 3, pp. 279–288, 2002.

[88] S. M. Pulukuri and C. S. Gondi, “Rna interference-directed knockdown of urokinase plasminogen activator and urokinase plasminogen activator receptor

inhibits prostate cancer cell invasion, survival, and tumorigenicity in vivo," *J. Biol. Chem.*, vol. 280, pp. 36529–36540, 2005.

[89] F. Al-Ejeh, D. Croucher, and M. Ranson, "Kinetic analysis of plasminogen activator inhibitor type-2: Urokinase complex formation and subsequent internalisation by carcinoma cell lines," *Exp Cell Res.*, vol. 297, pp. 259–271, 2004.

[90] A.L. Pang E.T. Ifon, W. Johnson, and K. Cashman, "U94 alters fn1 and angptl4 gene expression and inhibits tumorigenesis of prostate cancer cell line pc3," *Cancer Cell Int.*, vol. 22, pp. 5–19, 2005.

[91] R. S. Watnick and Y. N. Cheng, "Ras modulates myc activity to repress thrombospondin-1 expression and increase tumor angiogenesis;ras modulates myc activity to repress thrombospondin-1 expression and increase tumor angiogenesis," *Cancer Cell*, vol. 3, pp. 219–231, 2003.

[92] H. R. Abeysinghe, Q. Cao, J. Xu, S. Pollock, Y. Veyberman, N. L. Guckert, P. Keng, and N. Wang, "Thy1 expression is associated with tumor suppression of human ovarian cancer," *Cancer Genet. Cytogenet.*, vol. 143, pp. 125–132, 2003.

[93] B. C. Fuchs, J. C. Perez, J. E. Suetterlin, S. B. Chaudhry, and B. P. Bode, "Inducible antisense rna targeting amino acid transporter atb0/asct2 elicits apoptosis in human hepatoma cells," *Am. J. Physiol Gastrointest Liver Physiol.*, vol. 286, pp. G467–478, 2003.

[94] Q. X. Li, R. Sundaram N. Ke, and F. Wong-Staal, "Nr4a1, 2, 3-an orphan nuclear hormone receptor family involved in cell apoptosis and carcinogenesis," *Histol. Histopathol.*, vol. 21, pp. 533–540, 2006.

[95] I. Tabus and J. Astola, "On the use of mdl principle in gene expression prediction," *Journal of Applied Signal Processing*, vol. 2001, no. 4, pp. 297–303, 2001.

[96] A. Treves and S. Panzeri, "The upward bias in measures of information derived from limited data samples," *Neural Comput.*, vol. 7, pp. 399–407, 1995.

[97] "Drosophila interaction database," http://portal.curagen.com/cgi-bin/interaction/flyHome.pl; accessed May 15, 2008.

[98] J. Shen and C. Dahmann, "Extrusion of cells with inappropriate dpp signaling from drosophila wing disc epithelia," *Science*, vol. 307, pp. 1789–1790, 2005.

[99] H. H. Lee and M. Frasch, "Nuclear integration of positive dpp signals, antagonistic wg inputs and mesodermal competence factors during drosophila visceral mesoderm induction," *Development*, vol. 132, no. 6, pp. 1429–42, 2005.

[100] A. J. Hartemink and D.K. Gifford, "Combining location and expression data for principled discovery of genetic regulatory network models," in *Proceedings of Pacific Symposium on Biocomputing*, Honolulu, Hawaii, 2002, pp. 437–449.

[101] D. Heckerman, D. Geiger, and D. Chickering, "Learning Bayesian networks: The combination of knowledge and statistical data," *Machine Learning*, vol. 20, no. 3, pp. 197–243, 1995.

[102] I. T. Luna, Y. Yin, and Y. Huang, "Uncovering gene regulatory networks using variational bayes variable selection," in *Proceedings of IEEE Genomic Signal Processing and Statistics (GENSIPS)*, College Station, Texas, 2006, pp. 111–112.

[103] S. Rogers and M. Girolami, "A Bayesian regression approach to the inference of regulatory networks from gene expression data," *Bioinformatics*, vol. 21, no. 14, pp. 3131–3137, 2005.

[104] J. J. Rice, Y. Tu, and G. Stolovitzky, "Reconstructing biological networks using conditional correlation analysis," *Bioinformatics*, vol. 21, no. 6, pp. 765–773, 2005.

[105] N. Friedman, L. Cai, and X. Xie, "Linking stochastic dynamics to population distribution: An analytical framework of gene expression," *Phys. Rev. Lett.*, vol. 97, pp. 168302, 2006.

[106] L. Edelstein-Keshet, *Mathematical Models in Biology*, New York: Random House, 1988.

[107] D.B. Allison, G.L. Gadbury, M. Heo, J. R. Fernandez, C.-K. Lee, T. A. Prolla, and R. Weindruch, "A mixture model approach for the analysis of microarray gene expression data," *Computational Statistics & Data Analysis*, vol. 39, pp. 1–20, 2002.

[108] S. Mnaimneh, A. Davierwala, J. Haynes, and J. Moffat, "Exploration of essential gene functions via titratable promoter alleles," *Cell*, vol. 118, pp. 31–44, 2004.

[109] A. P. Gasch and P. T. Spellman, "Genomic expression programs in the response of yeast cells to environmental changes," *Molecular Biology of the Cell*, vol. 11, pp. 4241–4257, 2000.

[110] D. L. Eastmond and H. C. M. Nelson, "Genome-wide analysis reveals new roles for the activation domains of the saccharomyces cerevisiae heat shock

transcription factor (hsf1) during the transient heat shock response," *J. Biol. Chem.*, vol. 281, no. 43, pp. 32909–32921, 2006.

[111] T. Furuchi, H. Ishikawa, N. Miura, M. Ishizuka, and K. Kajiya, "Two nuclear proteins, cin5 and ydr259c, confer resistance to cisplatin in saccharomyces cerevisiae," *Mol Pharmacol.*, vol. 59, no. 3, pp. 470–474, 2001.

[112] C. E. Horak, N. M. Luscombe, J. Qian, and P. Bertone, "Complex transcriptional circuitry at the g1/s transition in saccharomyces cerevisiae," *Genes Dev*, vol. 16, pp. 3017–3033, 2002.

[113] P. Fabrizio, F. Pozza, S. D. Pletcher, C. M. Gendron, and V. D. Longo, "Regulation of longevity and stress resistance by sch9 in yeast," *Science*, vol. 292, no. 5515, pp. 288–290, 2001.

[114] P. J. Greens, "Reversible jump Markov chain Monte Carlo computation and Bayesian model determination," *Biometrika*, vol. 82, no. 4, pp. 711–732, 1995.

[115] A. E. Raftery, D. Madigan, and J. A. Hoeting, "Bayesian model averaging for linear regression models," *Journal of The American Statistical Association*, vol. 92, pp. 179–191, 1997.

[116] J. Cohen, *Statistical Power Analysis for the Behavioral Sciences (2nd ed.)*, Hillsdale, NJ: Lawrence Erlbaum Associates, 1988.

[117] P. Tamayo, D. Slonim, J. Mesirov, and Q. Zhu, "Interpreting patterns of gene expression with self-organizing maps: Methods and application to hematopoietic differentiation," *Proc. Natl. Acad. Sci. USA*, vol. 96, pp. 2907–2912, 1999.

[118] K. A. Bernstein and S. J. Baserga, "The small subunit processome is required for cell cycle progression at g1," *Molecular Biology of the Cell*, vol. 15, pp.

5038–5046, 2004.

[119] K. A. Bernstein, F. Bleichert, J. M. Bean, F. R. Cross, and S. J. Baserga, "Ribosome biogenesis is sensed at the start cell cycle checkpoint," *Molecular Biology of the Cell*, vol. 18, pp. 953–964, 2007.

[120] G. Thomas, "An encore for ribosome biogenesis in the control of cell proliferation," *Nature Cell Biology*, vol. 2, pp. E71–E72, 2000.

[121] M. B. Eisen, P. T. Spellman, P. O. Brown, and D. Botstein, "Cluster analysis and display of genome-wide expression patterns," *Proc. Natl. Acad. Sci. USA*, vol. 95, pp. 14863–14868, 1998.

[122] S. Tavazoie, J. D. Hughes, M. J. Campbell, R. J. Cho, and G. M. Church, "Systematic determination of genetic network architecture," *Nature Genetics*, vol. 22, pp. 281–285, 1999.

[123] X. Zhou, X. Wang, E. R. Dougherty, D. Russ, and E. Suh, "Gene clustering based on clusterwide mutual information," *Journal of Computational Biology*, vol. 11, pp. 147–161, 2004.

[124] C. D. Giurcaneanu, I. Tabus, J. Astola, J. Ollila, and M. Vihinen, "Fast iterative gene clustering based on information theoretic criteria for selecting the cluster structure," *Journal of Computational Biology*, vol. 11, pp. 660–682, 2004.

[125] P. D'haeseleer, "How does gene expression clustering work," *Nature Biotechnology*, vol. 23, pp. 1499–1501, 2005.

[126] I. Tabus and J. Astola, "Clustering the non-uniformly sampled time series of gene expression data," in *Proceedings of ISSPA 2003, International Symposium*

on Signal Processing and Applications, Paris, France, July 2-5, 2003, pp. 61–64.

APPENDIX A

144 DROSOPHILA PERIODICALLY EXPRESSED GENES

Gene Name	CG3140	CG6455	CG13279	CG5345	CG8684	CG5174
q-value	0.03	0.03	0.03	0.03	0.03	0.03
Gene Name	CG11242	CG13319	CG10248	CG10621	CG9126	CG6673
q-value	0.03	0.03	0.03	0.03	0.03	0.03
Gene Name	CG1091	CG14808	CG1471	CG5413	CG8357	CG6398
q-value	0.03	0.03	0.03	0.03	0.03	0.03
Gene Name	CG4316	CG6714	CG7780	CG7469	CG10658	CG5466
q-value	0.03	0.03	0.04	0.04	0.04	0.04
Gene Name	CG4928	CG8006	CG5253	CG2055	CG1408	CG7122
q-value	0.04	0.04	0.04	0.04	0.04	0.04
Gene Name	CG9047	CG7717	CG3770	CG8250	CG7082	CG4144
q-value	0.04	0.04	0.04	0.04	0.05	0.05
Gene Name	CG1523	CG17148	CG4443	CG8676	CG10602	CG9319
q-value	0.05	0.05	0.05	0.05	0.05	0.05
Gene Name	CG4071	CG9796	CG9858	CG11771	CG11836	CG1514
q-value	0.06	0.07	0.07	0.07	0.07	0.07
Gene Name	CG9216	CG4920	CG5871	CG11055	CG9763	CG9779
q-value	0.07	0.07	0.07	0.08	0.08	0.09
Gene Name	CG1090	CG10997	CG6510	CG2867	CG8187	CG2060
q-value	0.09	0.09	0.09	0.09	0.09	0.09
Gene Name	CG8947	CG7048	CG10916	CG3268	CG8739	CG8507
q-value	0.09	0.09	0.09	0.09	0.09	0.09

Gene Name	CG3992	CG12737	CG4759	CG9057	CG4608	CG7319
q-value	0.1	0.1	0.1	0.1	0.1	0.1
Gene Name	CG2221	CG3239	CG2903	CG10277	CG18662	CG12177
q-value	0.1	0.1	0.11	0.11	0.11	0.11
Gene Name	CG9089	CG3305	CG17818	CG10171	CG3365	CG8286
q-value	0.11	0.11	0.11	0.11	0.11	0.11
Gene Name	CG12236	CG15309	CG3492	CG1021	CG9071	CG18627
q-value	0.11	0.11	0.12	0.12	0.12	0.12
Gene Name	CG12263	CG1942	CG12131	CG9916	CG9581	CG2694
q-value	0.12	0.12	0.12	0.12	0.12	0.12
Gene Name	CG11120	CG9848	CG15433	CG5486	CG10977	CG12251
q-value	0.12	0.12	0.12	0.12	0.12	0.12
Gene Name	CG9392	CG1868	CG3756	CG6605	CG14045	CG1105
q-value	0.12	0.12	0.13	0.13	0.13	0.13
Gene Name	CG7563	CG4905	CG1891	CG11591	CG9804	CG3262
q-value	0.13	0.13	0.13	0.14	0.14	0.14
Gene Name	CG8954	CG3881	CG9140	CG11259	CG6302	CG7197
q-value	0.14	0.14	0.14	0.14	0.14	0.15
Gene Name	CG3460	CG1980	CG1193	CG7359	CG18539	CG11010
q-value	0.15	0.15	0.15	0.15	0.15	0.15
Gene Name	CG1583	CG17184	CG1462	CG4710	CG11440	CG4294
q-value	0.15	0.16	0.16	0.16	0.16	0.17
Gene Name	CG1963	CG6433	CG4897	CG9769	CG5555	CG7841
q-value	0.17	0.17	0.17	0.17	0.18	0.18
Gene Name	CG7096	CG6936	CG9553	CG4556	CG11186	CG3045
q-value	0.18	0.18	0.18	0.18	0.18	0.19

VITA

Wentao Zhao received the B.S. and M.S. degrees in Electrical Engineering from the Tsinghua University, Beijing, China, in 1999 and 2002, respectively. He enrolled in Department of Electrical and Computer Engineering at Texas A&M University in College Station in June 2002 and obtained a Master degree in August 2005. He then joined the Genomic Signal Processing group as a Ph.D. student under the supervision of Dr. Erchin Serpedin and Dr. Edward. R. Dougherty. He was conferred the Ph.D. degree in August 2008. His research interests included information theory and pattern recognition in genomic signal processing.

His address is:

Zachry 214
Department of Electrical and Computer Engineering
Texas A&M University
College Station, Texas 77843-3128

The typist for this thesis was Wentao Zhao.