INFERRING REGULATORY INTERACTIONS IN TRANSCRIPTIONAL REGULATORY NETWORKS

By

Sherine Awad Mahmoud

A DISSERTATION

Submitted to
Michigan State University
in partial fulfillment of the requirements
for the degree of

Computer Science - Doctor of Philosophy

2013

ABSTRACT

INFERRING REGULATORY INTERACTIONS IN TRANSCRIPTIONAL REGULATORY NETWORKS

By

Sherine Awad Mahmoud

Living cells are realized by complex gene expression programs that are moderated by regulatory proteins called transcription factors (TFs). The TFs control the differential expression of target genes in the context of transcriptional regulatory networks (TRNs), either individually or in groups. Deciphering the mechanisms of how the TFs control the expression of target genes is a challenging task, especially when multiple TFs collaboratively participate in the transcriptional regulation. Recent developments in biotechnology have been applied to uncover TF-target binding relationships to reconstruct draft regulatory circuits at a systems level. Furthermore, to identify regulatory interactions in vivo and consequently reveal their functions, TF single/double knockouts and over-expression experiments have been systematically carried out. However, the results of many single or even double-knockout experiments are often non-conclusive, since many genes are regulated by multiple TFs with complementary functions.

To predict the TF combinations that the knocking out of them are most likely to bring about the phenotypic change, we developed a new computational tool called TRIM that models the interactions between the TFs and the target genes in terms of both the TF-target interaction's function (activation or repression) and its corresponding logical role (necessary and/or sufficient). We used DNA-protein binding and gene expression data to construct regulatory modules for inferring the transcriptional regulatory interaction models for the TFs and their corresponding target genes. Our TRIM algorithm is based on an HMM and a set of constraints that relate gene expression patterns to regulatory interaction models. However, TRIM infers up to 2-TFs interactions. Inferring the

collaborative interactions of multiple TFs is a computationally difficult task, because when multiple TFs simultaneously or sequentially control their target genes, a single gene responds to merged inputs, resulting in complex gene expression patterns. We developed mTRIM to solve this problem with a modified association rule mining approach. mTRIM is a novel method to infer TF collaborations in transcriptional regulation networks. It can not only identify TF groups that regulate the common targets collaboratively but also TFs with complementary functions. However, mTRIM ignores the effect of miRNAs on target genes.

In order to take miRNAs' effect into considerations, we developed a new computational model called TmiRNA that incorporates miRNAs into the inference. TmiRNA infers the interactions between a set of regulators including both TFs and miRNAs and the set of their target genes. We used our model to study the combinatorial code of Human Cancer transcriptional regulation.

To Prophet Mohammed (PUH)

ACKNOWLEDGMENTS

I also would like to thank my committee members; Dr. Gregg Howe, Dr. Rong Jin, and Dr. Titus Brown for their valuable comments and feedbacks.

Special thanks to Dr. Titus Brown for his great effort in one of the most benefical classes I ever had (CSE891, Open problems in Bioinformatics). Thanks to Dr. Jason Pell (former colleague in GED lab) for his help and support. I also would like to thank our helpful grad director Dr. Eric Torng for his support. I also would like to thank Dr. Radha Hyder and Dr. Hassan Khalil for their kind advice and support.

Many thanks to Dr. Rami Halloush (former colleague in Chen lab) for his generous help, advice, and continous encouragements.

I would like to thank Dr. Yuhua Jiao for her support and advice. encouragements. I also would like to thank Dr. Bruce Rosa and Jiajie Peng (former colleagues in Chen lab) for their help and advice.

I am extremely grateful to my parents and brothers for their continuous support and patience.

Many thanks to my friends Shireen Ahmed, Ghada Abdel Samie and Azza Sallam for their overseas support and encouragments. Many thanks to my friend Donna Fernandez for the great time I spent with her in MSU, her generous advice, and encouragements.

Many thanks to my spirtual fathers and professors, Dr. Ahmed Sharaf Eldin and Dr Hamed Nassar for their conitnous encouragments and care.

Finally, I would like to thank my generous funding of my PhD by the Egyptian government.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter 1

Introduction

Genetic information encoded in a cell's genome can determine the properties of a cell. In order to understand how such information define distinct cell types and cellular states, we need to understand how such information is differentially and dynamically retrieved. Transcriptional factors (TFs) and microRNAs (miRNAs) represent the most numerous factors that control the expression of genomic information. Hence, understanding the transcription regulation and how a set of genes are regulated in concert by a group of TFs is important to decipher such information.

Understanding transcription regulations is a key to understand cellular differentiation and how an organism responds to a stimuli. In addition, numerous human disease have been associated with mutations in transcriptional regulatory elements. It is also recently discovered that chromosomal rearrangement involving regulatory elements result in variety of cancers. Moreover, many human diseases are caused by complex interactions of multiple genes and not only caused by mutations in single genes. Therefore, understanding transcriptional modulation can have therapeutic benefits. Furthermore, engineering transcriptional activators can be used as therapeutic agents in diseases caused by loss of gene expression [9, 10].

In Section 1.1 and Section 1.2, we explain both the experimental and computational methods used to study transcription regulation respectively. In Section 1.3, we provide a brief introduction to the computational models used in this dissertation. In Section 1.4, we define our problem and the focus of this dissertation. We briefly summarize the contributions of the dissertation in Section 1.5. Finally, in Section 1.6, we provide the organizational structure of the dissertation.

1.1 Experimental Methods For Understanding Transcriptional Regulation

Information about a genetic network may be revealed experimentally by applying a directed perturbation to the network [4, 17, 61, 79]. This helps observing the steady-state expression levels of every gene in the network in the presence of the perturbation [4, 17, 61, 79]. Perturbations may be genetic, or biological. In genetic perturbation, the expression levels of one or more genes are fixed by deletion or over-expression. In biological perturbation, one or more non-genetic conditions are altered, such as a change in growth media, a temperature increase [4, 17, 61, 79]. Single gene perturbation has the following advantages; 1) It is possible to observe directly the functional effect of controlled changes, 2) it easy to interpret the data, 3) it leads to gene variants that have more easily detectable effects, 4) introduces qualitatively different proteins and thereby uncovers the range of gene function. For the above reasons, single gene perturbation has been proven to be successful in extending our knowledge of pathway components and interactions [4, 61, 79]. However, if there is no one-to-one correspondence between elements and functions, single perturbation experiments will fail to reveal the functional structure of a system [4, 61, 79]. For example, if a gene A can be activated by two routes, in one route, gene A is activated directly be gene C and in another route, gene A is activated by gene B and gene C. In such case, only double knockout can lead to the loss of function phenotype for gene C [4, 61, 79]. Hence, multiple perturbations is essential for understanding moderately complex biological systems [4, 61, 79]. To verify the TF-target gene relationships and to detect the TF functions in vivo, TF knockouts and/or over-expression experiments are usually carried out [34]. Knockout experiments can also detect whether a TF is necessary to its targets. Over-expression experiments can also detect whether a TF is sufficient to its targets. However, single knockout or over-expression may not provide statistically significant evidence due to redundancy or confounding signals from indirect regulatory feedback [49]. For example, it has been shown that approximately 73% (about 4,500) of the known genes of Saccharomyces cerevisiae (yeast) are non-essential [32]. The results of the single knockout or over-expression experiments are therefore often non-conclusive, as it is highly likely that multiple non-essential genes can be involved. This has led to the development of automated experimental methods for double-knockouts to provide more statistically significant determination of the TF functions [29]. However, to systematically knockout (or over-express) all possible combinations of the TFs in the whole genome is still challenging. Given an organism with k TFs, the total number of possible double-TF combinations is [13]:

$$k*(k-1)/2 \tag{1.1}$$

For complex organisms, k can be easily in the range of thousands. Instead of blindly trying out all possible TF pairs for double-knockout experiments, one solution is to select the TF pairs that are most likely to bring about the phenotypic change. To do so, we need to understand the interaction models employed by the TFs to influence the regulatory patterns of the target genes in the network. In other words, we need to uncover the models of TF-target regulatory interactions of the TF pairs and the target gene in terms of the TF-target interactions' directions (activation or repression) and their corresponding logical roles (necessary and/or sufficient).

A fundamental step in transcriptional regulation is a transcription factor binding DNA[19,20]. Recent advances in DNA sequencing and microarray technologies lead to a huge increase in the amount of information that can be accumulated about this process [28, 30]. Particularly, ChIP-chip and ChIP-seq experiments enabled genome-wide mapping of TF binding sites in a single experiment [28, 30, 73]. The knowledge of these binding sites is a key step in determining which gene is regulated by which TFs. However, the binding sites alone are not sufficient to infer transcriptional regulation [28, 48].

In simple organisms, such as bacteria or yeast, transcriptional regulation usually occurs by a TF binding in a promoter region, near the transcription start site of a gene [28]. However, in more complex organisms, regulation occurs through long-range enhancers, often spanning many tens of thousands of base pairs [28, 30]. In such case, it is not obvious which gene a specific binding site might be regulating [28, 30]. This is because the enhancer regulation is independent of the binding site orientation corresponding to the regulated gene and transcription factors bound to a particular gene may regulate a different gene through an enhancer [28, 30]. Additionally, binding sites in gene-rich locations of a genome have a large number of potential targets in its neighborhood [28, 30]. Although we might have an indication of the maximum range at which an enhancer is likely to function, any gene within that distance is considered a potential target [28, 30]. In addition, the presence of a regulator at a promoter region indicates binding but not function or regulation [6]. For the above reasons, binding information is not sufficient to infer transcriptional regulation [6, 28]. Therefore, binding data can be combined with expression data from microarrays, and with sequence motif searches for transcription factor binding sites. When two or all the three experiments agree, we expect to have a reliable assignment of a transcription factor to a gene [58, 76].

1.2 Computational Methods For Understanding Transcriptional Regulation

A significant effort has been devoted to modeling and learning regulatory networks. These studies modeled regulatory interactions and dynamics with various degrees of details, flexibility, and assumptions. Such models can be hierarically classified based on the model of regulatory interactions. Differential equations and stochastic models can be found on the highest level as they provide detailed descriptions of regulatory systems and provide simulations of systems dynamics [6, 23, 62]. However, these models require accurate estimations of a large number of parameters and hence they are computationally demanding. At the next level lies Boolean networks. Boolean networks simply considers genes as on or off gates. They also include logic interactions including AND, OR, XOR,..etc. Though, the assumption held by Boolean networks is simple, Boolean networks can be learned with fewer data. In addition, Boolean networks are robust and easy to interpret. Linear model are intermediate approaches in terms of complexity and robustness. Linear models describe the expression level of a gene as a weighted sum of the levees of its predictors [8, 67].

Reducing the dimensionality of the search space before inference can simplify the process of learning and/or modeling regulatory networks. One approach to this reduction is to group co-regulated genes into clusters where gene expression clustering is commonly used. However, co-expressed genes does not correspond to co-regulated genes specially under different experimental conditions as genes are often regulated differently under different conditions. For this reason, Biclustering was developed to find co-regulated genes on the basis of coherence under subsets of observed conditions [11].

In order to infer the causal relationship among different elements such as genes, proteins, metabolites, neurons and so on, based upon multi-dimensional temporal data, Granger causality and dynamic Bayesian network inference are used and massively reported in research. The dynamic Bayesian network inference performs better than the Granger causality when the data size is short, while Granger causality proved better performance when the data size is long [85].

The previous briefly explained approaches have been widely used to decipher the mechanisms of how the TFs control the differential expression of a target gene in a TRN. In order to increase the understanding of the regulatory mechanism, we need to decipher how multiple TFs collaborate in regulating their targets. In the previous section, we have showed that systematically knockout (or over-express) all possible combinations of the TFs in the whole genome is still challenging. Connecting the expression patterns to the interaction model can help revealing the role of each TF instead of blindly trying each individual TF [6, 30]. Deciphering the roles of TFs under different conditions, help understanding the dynamic mechanisms of TRNs.

Several approaches have been studied to reveal the dynamic mechanism of TRNs. Dynamic Regulatory Events Miner (DREM) integrates timeseries expression data and ChIP-chip or motif information to infer an annotated global temporal map [24]. This global map uncovers the transcriptional regulatory events that can help discovering the timeseries expression patterns and the factors controlling these events during a cells response to stimuli [24]. DREM takes a unique approach by focusing its modeling of temporal regulatory interactions on bifurcation points. Bifurcation events occur when a set of genes share a similar expression path up to a certain time point diverge [24]. This approach may lead to better insights regarding the system being studied, this is because TFs may bind to different genes at different time points, and DREM can derive dynamic maps that associate TFs with the genes they regulate and their activation time points. These insights help identifying master and secondary regulators respectively that control the initial response, and responsible for more specific pathways respectively. These insights can also explain several aspects of the observed response, for example the condition-specific activity of factors and the activation of certain network motifs [24]. Although this approach can identify master and secondary regulators, it cant infer the logical roles of the TFs; it also can’t infer the combinatorial interactions of multiple transaction factors [24, 79]. Another approach used a computational approach to capture combinatorial effects of multiple transcription factors in transcription control. Binding and expression data were used to identify regulatory modules including subsets of regulators and genes together with a regulatory program [79]. However, this approach reduced the combinatorial functions of TFs to the properties of individual TFs and it did not consider the combinatorial interactions of TFs. In addition, the method uses an incremental approach that does not study the functions of the TFs simultaneously because of the scalability issue introduced by the greedy search.

However, the previous approaches do not take the effect of microRNAs (miRNAs) into considerations. Le et al. in [40] developed a protein interaction-based MicroRNA Modules (PIMiM) , a new model that integrates sequence, expression and interaction data to identify modules of mRNAs controlled by sets of miRNAs. PIMiM combines regression with network information to discover these modules. PIMiM is used to infer condition-specific regulation of miRNAs and their targets. Le et al. in [72] proposed an approach to infer the causal knockdown effects of each single miRNA on the regulations of its targets, using expression profiles of miRNAs and mRNAs without taking into consideration prior target information. However, the proposed approach considers the effect of a single miRNA on each of its targets and it does not consider the level of regulations that a group of miRNAs has on a particular gene.

1.3 Computational Models

In this section, we provide a brief background for the computational methods used in the dissertation.

1.3.1 Hidden Markov Model (HMM)

A hidden Markov model (HMM) is a statistical Markov model in which the system being modeled is assumed to hold the Markov process property with unobserved (hidden) states. A stochastic process is said to have the Markov property if the conditional probability distribution of its future states depends only upon the current state. Hidden Markov model is very popular machine learning tool in bioinformatics [19, 22]. To formalize the notation of hidden Markov models, let p be the path or the sequence of states. This path follows a simple Markov chain which means that the probability of a state depends solely on the previous state [19, 22]. This chain can be represented as shown in Equation 1.2.

$$a_{kl} = P(\Pi_i = l|\Pi_{i-1} = k) \tag{1.2}$$

Where Π_i is the i th state in the path and a_{kl} is the transition probability from state k to state l. A begin state is used to model the beginning of sequences in Markov chains. The transition probability a_{0k} presents the probability for the model to transit from this begin state to state k, this intrinsically means the probability that the Markov chain starts at state k. Similarly, we can use an end state to model the ending of a state sequence to transit the sequence into an end state. In addition, each state can produce a symbol from a distribution over all possible symbols. Hence, an emissions probability $e_k(b)$ in Equation 1.3 represents the probability that symbol b is seen when the model is in state k [19, 22].

$$e_k(b) = P(x_i = b|\Pi_i = k) \tag{1.3}$$

There are three fundamental problems associated with hidden Markov models are introduced in the following sub-sections.

Considering the underlying states to find out what the sequence of observations means is called decoding. Several approaches have been proposed for decoding. We will present here the most common among them, the Viterbi algorithm. The Viterbi algorithm is a dynamic programming algorithm for finding the most likely sequence of hidden states called the Viterbi path that results in a sequence of observed events. If we have to choose only one path for our prediction, then the one with the highest probability is the most probable path to be chosen [19, 22].

$$\Pi^* = argmax_{\Pi} P(x, \Pi) \tag{1.4}$$

Since many different state paths can produce the same sequence x, we need to compute of a particular observation sequence x as in Equation 1.5.

$$P(x) = \sum_{\pi} P(x, \Pi) \tag{1.5}$$

But the number of possible paths Π increases exponentially with the length of the sequence. This makes the brute force enumeration for all possible paths inapplicable. This probability can be computed using a similar approach to the Viterbi algorithm, by replacing the maximization steps to summations [19, 22].

The aim of HMM learning is to determine the emission and transition probabilities from the training samples. The forward-backward algorithm can be used to determine the model parameters. The algorithm works basically by updating the weights in order to better explain the observed training sequences. The algorithm first computes the probability of $A_{kl}$and $E_k(b)$ by looking at the probable path of the training sequence using the current a_{kl} and $e_k(b)$. Equation 1.6 and Equation 1.7 are used to update the values of the transition and emission probabilities respectively. The process is repeated till a stopping criterion is reached [19, 22].

$$a_{kl} = \frac{A_{kl}}{\sum_{l'} A_{kl'}} \tag{1.6}$$

$$e_k(b) = \frac{E_k(b)}{\sum_{b'} E_k(b')} \tag{1.7}$$

1.3.2 Bayesian Inference

Bayesian models provide full joint probability distributions over both observed data and unobserved model parameters. Bayesian statistical inference is carried out using Bayes rule. For a probabilistic model, the parameters of the model are to be inferred from data. Assume we would like to infer parameters for a model M from a set of D. Suppose there is a probability distribution over the parameters . If we condition on M , gives the version of Bayes theorem as in Equation 1.8.

$$P(\Theta|D,M) = \frac{P(D|\Theta,M)P(\Theta|M)}{P(D|M)} \tag{1.8}$$

Where $P(\Theta|M)$ is the prior probability which has to be chosen in a reasonable manner, $P(\Theta|D,M)$is the posterior probability, and $P(D|\Theta,M)$ is the likelihood probability [19, 22].

The EM algorithm is an efficient iterative procedure to compute the Maximum Likelihood (ML) estimate if there is some missing or hidden data. In ML estimation, we would like to estimate the model parameter(s) for which the observed data are the most likely. Each iteration of the EM algorithm consists of two processes: The E-step, and the M-step.

E-step: In this step, the missing data are estimated given the observed data and current estimation of the model parameters. This is achieved using the conditional expectation, explaining the choice of terminology [19, 22]. M-step: in this step, the likelihood function is maximized under the assumption that the missing data are known. The estimation of the missing data from the E-step is used in behalf of the actual missing data. Convergence is assured since the algorithm is guaranteed to increase the likelihood at each iteration [19, 22].

The fact that each iteration improves the likelihood of (Θ), in addition to the simplicity and ease of implementation make the EM algorithm so useful. However, the rate of convergence can become excruciatingly slow as approaching local optima. In addition, EM works best when the percentage of missing information is small and the dimensionality of the data is not too large. Hence, EM can require much iteration, and higher dimensionality can significantly slow down the E-step [19, 22].

1.3.3 Association Rules Mining

Association analysis is a useful methodology for discovering interesting relationships hidden in a large datasets where the uncovered relationships are represented in the form of association rules or frequent items. Formally, an association rule is an expression of the form A ⇒ B, where A, and B are disjoint itemsets, such that A ∩ B =∅. The significance of an association rules can be measure using the support and the confidence of the rule. The support measures how often the rule occurs in a dataset. While the confidence measures how frequently items in B are found in transactions that contains A. Formally,

$$Support, s(A \Rightarrow B) = \frac{s(A \cup B)}{N} \tag{1.9}$$

$$Confidence, c(A \Rightarrow B) = \frac{s(A \cup B)}{s(A)} \tag{1.10}$$

where N is the number of transactions in the dataset. When a rule has a very low support, it means it happens by chance and hence less interesting. On the other side, confidence of a rule reflects the reliability of the inference. If A⇒ B has a high confidence, then it is more likely that B occurs in the transactions that contain A. Association rules mining can be mathematically defined as follows: Given a set of transactions T, we need to find all the rules with support ≥ ms, and confidence ≥mc, where ms and mc are the minimum support and minimum confidence thresholds respectively and are computed using Equation 1.9 and Equation 1.10 respectively. The basic strategy used by many association rule mining algorithms is to divide the problem into two subtasks:

1. Frequent Itemset Generation: The goal of this step is to find all the frequent itemsets. Frequent itemset is an itemset that satisfy the minimum support threshold.

2. Rule Generation: The goal of this step is to find all the rules that have confidence higher than the minimum confidence.

Apriori is one of the oldest algorithms that is widely used in mining association rules [1]. Apriori applies the Apriori principle to prune the exponential space. Apriori principle states that if an itemset is frequent, then all of its subsets must also be frequent, or if an item set is infrequent then all its supersets must also be infrequent.

Apriori algorithms performs several passes over the data in order to mine association rules. In the first pass, the algorithm simply counts the number of items occurrence to determine the frequent 1-itemsets, L_1. In pass k, the algorithm uses the L_{k-1}-itemsets found in the k-1 pass, to generate the k-candidate itemsets (C_k) using an Apriori generator function. The Apriori generator works as follows:

1. Join L_{k-1} with L_{k-1}.

2. Delete every $c \in C_k$ such that any $k-1$-subset of c is not in L_{k-1}

Then the database is scanned again to count the support of candidates in C_k.

Although many association rule mining algorithms use support and confidence, support and confidence have some limitations. Many potentially interesting rule that have low support items might be eliminated because of the given minimum support. The confidence ignores the support of the itemset of the rule consequent. Hence, various measures have been used to qualify the significance of an association rule. In addition, Apriori suffers from a number of inefficiencies. One major trade-off of Apriori is that the candidate generation step generates large numbers of subsets.

1.4 Problem Statement

We define a regulatory module R (TF; G; I) as a set of genes G regulated in concert by a group of one or more TFs that govern the target genes' behaviors via appropriate TF-target regulatory interactions in I [7]. There are two types of regulatory modules. In an independent regulatory module, the target genes are solely regulated by one TF. A TF's regulatory interaction model can be defined in terms of two properties: the TF's functional role as an activator or a repressor, and its logical role as being necessary or sufficient. Table 1.1 shows such a regulatory interaction model as proposed by previous researchers [3, 7] . Note that the categories in the TF's functional and logical roles can be combined. A TF-target interaction model in R can be Activator Necessary (AN), Activator Sufficient (AS), or Activator Necessary and Sufficient (ANS). Similarly for TFs that are repressors, they can be RN, RS, or RNS [5]. A multiple regulatory interaction is defined as a group of TFs that collaborate to control the expression levels of the same target genes. The directions of all the TFs in the group, therefore, form a transcriptional regulation pattern of the target genes. In a multiple-TF regulatory interaction, a group of TFs collaborate to control the expression levels of the same target genes.

This dissertation focuses on identifying TF groups that are collaboratively responsible for target gene expressions by uncovering the regulatory interactions in terms of their directions and corresponding logical roles from gene expressions and TF-DNA binding data.

1.5 Dissertation Contribution

The following is a summary of the contributions of this dissertation:

1. We developed an HMM based model called TRIM that relate gene expression patterns to

Table 1.1: TF-Target interactions can be modeled in terms of the TF's functional role as an activator (up-regulates the target gene's expression) or a repressor (down-regulates the target gene's expression), and the logical role of the TF as being necessary and/or sufficient. The two categories (functional and logical roles) can be combined.

Role	**Concept**	**Description**
Functional	Activator	Response of target gene is inline with the expression change of TF
	Repressor	Response of target gene is opposite to the expression change of TF
Direction of Logical	Necessary	Decreasing TF's expression level leads to responses opposite to its functional role
	Sufficient	Increasing TF's expression level leads to the responses consistent with its functional role
	Necessary and Sufficient	Increasing TF's expression level leads to responses consistent with its functional role, and decreasing the TF's expression level leads to the responses opposite to its functional role

regulatory interaction models based on a set of constraints. TRIM infers the regulatory interactions of one or two TFs. Although, multiple TFs can collaboratively regulate their set of target genes, TRIM still solves a sub-problem where target genes are regulated by 2 TFs.

2. In order to infer *k*-TFs regulatory interactions, we developed mTRIM, an integration of an EM-based Bayesian inference and a new association rule mining approach built on a set of basic constraints that relate gene expression patterns to regulatory interactions. mTRIM infers the regulatory interactions of *k*-TFs that collaboratively regulate a set of target gene with no limitations on the size of *k*. Clearly, mTRIM has higher scalability than TRIM. In addition, mTRIM gives an insight on the replacements between TFs.

3. To better study transcription regulation and to enhance the performance, we incorporated miRNAs into the inference. We designed TmiRNA, a sequential mining approach that infer the interactions of *k* regulators including TFs and miRNAs and their set of target genes. Instead of constructing regulatory modules based on binding network and gene expressions,

TmiRNA provides a new approach to construct regulatory modules based on regulatory interactions.

1.6 Dissertation Outline

In this chapter, we provided a brief background about transcriptional regulation. We briefly described the the experimental methods that can be used to study transcriptional regulation. Furthermore, we explained some of the theoretical bounds of those experimental methods, and the need to have new computational models to study transcriptional regulation. Then, we explained briefly the basics of the computational models used through out the dissertation. Based on this solid foundation, we stated the main problem we aim to solve in this dissertation.

In Chapter two, we describe our first approach (TRIM) to infer TFs interaction models. This approach is limited to 2-TFs.

In Chapter three, we describe a new approach (mTRIM) to infer k-TFs interaction models, with no bound on the size of k.

In Chapter four, we incorporate miRNA into the inference. We described TmiRNA, a new sequential mining approach, to infer k-regulators interaction models, the regulators are a combination of TFs and miRNAs. We also describe how TmiRNA constructs regulatory modules based on interaction models.

Finally, in Chapter five, we conclude our work and shed light to our future work.

Chapter 2

Inferring 2-TF Regulatory Interactions

2.1 Introduction

The complex gene expression programs in living cells are moderated by regulatory proteins called transcription factors (TFs) that control the transcription of genes in the context of transcriptional regulatory networks (TRNs) [56]. The TFs interact with their target genes in the TRNs to up or down-regulate gene expression. They can act independently or collaboratively with other TFs, leading to different *TF-target interaction models* that influence the regulation patterns of target genes in different ways [22, 79].

Various experimental methods have been developed to unravel the complex regulatory mechanisms behind biological processes. Recent developments in biotechnology (*e.g.,* chromatin immunoprecipitation, yeast one-hybrid and next-generation sequencing) have been used to indirectly or directly uncover TF binding relationships [16, 58] to reconstruct draft regulatory circuits at a systems level [22, 57, 79]. To verify the TF-target gene relationships and to detect the TF functions *in vivo*, TF knockouts and/or overexpression experiments are usually carried out [29].

However, single knockout or overexpression may not provide statistically significant evidence due to redundancy or confounding signals from indirect regulatory feedback [28]. For example, it has been shown that approximately 73% (about 4,500) of the known genes of *S. cerevisiae* (yeast) are non-essential [30]. The results of the single knockout or overexpression experiments are therefore often non-conclusive, as it is highly likely that multiple non-essential genes can be involved.

This has led to the development of automated experimental methods for double-knockouts to provide more statistically significant determination of the TF functions [73]. However, to systematically knockout (or overexpress) all possible combinations of the TFs in the whole genome is still challenging. Given an organism with k TFs, the total number of possible double-TF combinations is as in Equation 1.1. For complex organisms, k can be easily in the range of thousands.

Instead of blindly trying out all possible TF pairs for double-knockout experiments, one solution is to select the TF pairs that are most likely to bring about the phenotypic change. To do so, we need to understand the interaction models employed by the TFs to influence the regulatory patterns of the target genes in the network. In other words, we need to uncover the models of TF-target regulatory interactions of the TF pairs and the target gene in terms of the TF-target interactions' directions (activation or repression) and their corresponding logical roles (necessary and/or sufficient).

In this chapter, we design a set of constraints that relate gene expression patterns to regulatory interaction models, and propose an algorithm TRIM (Transcriptional Regulatory Interaction Model Inference) [3] to systemically infer the regulatory interaction models between individual TFs, as well as any two TFs, and their target genes, from wild-type time-series gene expression data. Our TRIM algorithm is based on a hidden Markov model (HMM). Experimental results on yeast data showed that TRIM outperformed the existing algorithms for inferring the regulatory interaction models of TFs and their target genes for individual TFs as well as pairs of TFs that are in collaborative regulatory modules. In addition, on an individual Arabidopsis binding network, we showed that the target genes' expression correlations can be significantly improved by incorporating the TF-target regulatory interaction models inferred by TRIM into the expression data analysis, which may introduce new knowledge in transcriptional dynamics and bioactivation. We define a regulatory module $R(TF,G,I)$ as a set of genes G regulated in concert by a group of one or more

TFs that govern the target genes' behaviors via appropriate TF-Target regulatory interactions in *I* [61]. There are two types of regulatory modules. In an *independent regulatory module*, the target genes are solely regulated by one TF. In a *collaborative regulatory module*, the target genes are regulated by multiple TFs. In this chapter, we focus on collaborative regulatory modules with up to 2-TFs. In the following chapters, we will show high order regulatory modules.

A TF's regulatory interaction model can be defined in terms of two properties: the TF's functional role as an activator or a repressor, and its logical role as being necessary or sufficient. Table 1.1 shows such a regulatory interaction model as proposed by previous researchers [61, 79].

Note that the categories in the TF's functional and logical roles can be combined. A TF-target interaction model in *R* can be Activator Necessary (*AN*), Activator Sufficient (*AS*), or Activator Necessary and sufficient (*ANS*). For example, pheromone response elements are necessary and sufficient for basal and pheromone-induced transcription of the FUS1 gene of yeast[26]. Similarly for TFs that are repressors, they can be *RN*, *RS* or *RNS* [4].

Yeang and Jaakkola [79] attempted to characterize the combinatorial regulatory models of multiple TF-target interactions using a heuristic approach to measure how well *R* fits the associated binding and gene expression data with a log likelihood function. The regulatory module's likelihood is maximized with a greedy approach by incrementally adding genes to the module and monitoring the predictions of the TF-target interactions for optimality. However, this incremental approach does not study the functions of the TFs simultaneously because of the scalability issue introduced by the greedy search. Their method also uses a p-value based approach to calculate the significance of the combinatorial property of a TF, determined by the gap of log likelihood scores between their model and a model built on the randomized gene expression data based on the entire time frame. However, as stated in Ernst *et al* [22], a TF usually functions at specific "activation timepoints" instead of throughout the entire time frame. This means that the identification of TF-target interaction modules should be focused on such activation timepoints rather than comparing with random gene expression data of the entire time frame. In this chapter, we include a step to recognize the activation timepoints of the target genes in TRIM. Our experimental results will show that the target genes' expression correlation are indeed markedly improved by taking the actual activation timepoints into consideration.

Another related algorithm is DREM, which was proposed to derive dynamic regulatory networks that associate TFs with target genes and their activation timepoints [22]. To uncover transcriptional regulatory events leading to the observed temporal expression patterns and the underlying factors that control these events during a cell's response to stimuli, DREM integrates time-series gene expression data and protein-DNA binding data to build a global temporal map. The method mainly works by identifying bifurcation timepoints where the expression of a subset of genes diverges from the rest of the genes. The bifurcation points are then annotated with the TFs regulating these transitions, which result in a unified temporal map. The method can therefore facilitate the determination of the time when TFs are exerting their influence, and assigns genes to paths in the map based on their expression profiles and the TFs that control them. Unlike the method by Yeang and Jaakkola [79], DREM's ability to derive dynamic maps that associate TFs with the genes they regulate and their activation timepoints has indeed led to better insights for the regulatory module being studied. For example, one can identify master regulators that control the initial response and secondary regulators that are responsible for specific pathways. Numerous aspects of the observed response, including the condition-specific activity of factors and the activation of certain network motifs, can also be explained using DREM. However, unlike our method, DREM does not infer the logical roles of the TFs; for example, whether a specific master or secondary TF is necessary or sufficient for regulating a set of target genes. Such knowledge are essentially useful for understanding the complex regulatory mechanisms of many biological processes. We will show in this chapter that by incorporating the knowledge of the regulatory interaction model, we can significantly improve the computation of gene expression correlation.

2.2 Concepts

The kernel of our TRIM method is an HMM, which is a stochastic model that assumes the Markov property holds and all the states are unobserved (hidden). A stochastic process is said to have the Markov property if the conditional probability distribution of its future states depends only upon the current state, as shown in Equation 2.1:

$$p(x(t+1)|x(0),....,x(t)) = p(x(t+1)|x(t)) \tag{2.1}$$

An HMM consists of a set of hidden states and each state has a probability distribution over the possible outputs [18]. In an HMM, a state k_i transits to another state k_j with a probability $P(k_j(t+1)|k_i(t))$. A state k can emit an output b with emission probability $e_k(b) = P(output = b|state = k)$ [18, 19, 51]. See Section 1.3.1 for more details about HMM.

In this chapter, each state refers to a possible TF-target interaction model, while the output emitted by a state will indicate whether a particular interaction model is true [19].

By taking advantage of the HMM, our TRIM algorithm can consider the influence from multiple TFs simultaneously. Prior biological knowledge on the TFs such as gene perturbation experiments can also be incorporated in the setting of the initial emission probabilities, while the time-dependent regulatory relations can be effectively captured by preserving the time dependency within the HMM.

In this chapter, we assume that a TF-target interaction is consistent in the context of transcriptional control as long as the experimental conditions are unchanged. We also assume that the activity of a TF is proportional to its mRNA abundance over time. Although these assumptions may be violated in practice, existing algorithms for inferring TF-target interaction models at different levels of complexity [6, 22, 61, 79] have all been developed with these assumptions.

2.3 Methods

Given a large-scale TRN, we design our TRIM algorithm for inferring TF-target interaction models systematically from large-scale TRNs. A TRN can be represented as a directed graph, in which each node is a TF or a gene, and each edge represents a regulation relationship between a TF and a target gene. The framework of TRIM, as shown in Figure 2.1, consists of two steps: the first step constructs the regulatory modules from the TRN; the second step infers the TF-target interaction models in each of the regulatory modules.

Figure 2.4 shows that the genes in yeast, one of the most well studied eukaryotic organisms, are regulated mostly by individual TFs (69.6%) or pairs of TFs (18.2%). Therefore, in this chapter, we focus on independent regulatory modules and collaborative regulatory modules with two TFs (we call these “2-TF collaborative modules”).

2.3.1 Constructing Regulatory Modules

Given a large-scale TRN, a TF may regulate multiple target genes simultaneously but with different types of TF-target interactions. To construct regulatory modules, we extract the subnetworks using gene expression clustering followed by graph partitioning (Figure 2.1 step 1).

We first cluster all the target genes in a TRN based on their gene expression values with Cluster 3.0 (specifically, k-means), which uses Pearson correlation coefficient for gene similarity metric [20]. The clusters are then evaluated with Gene Ontology enrichment analysis using Bingo, and un-

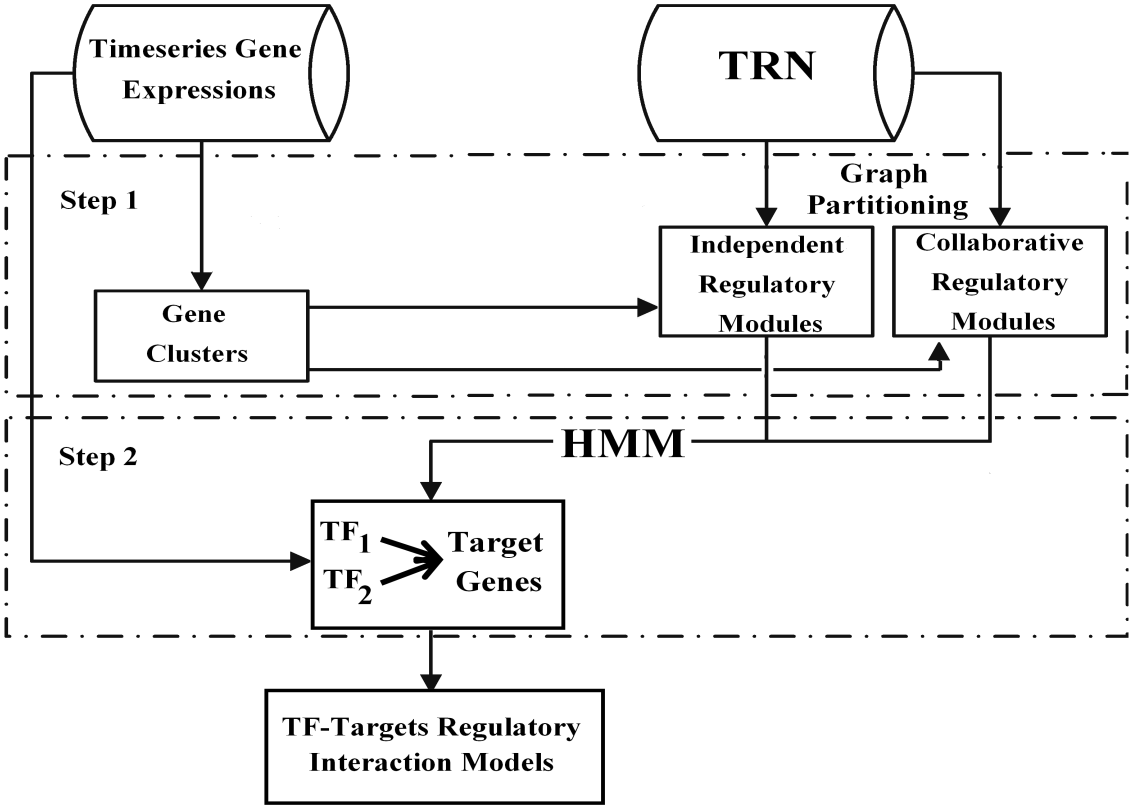

Timeseries Gene Expressions
TRN
Step 1
Graph Partitioning
Independent Regulatory Modules
Collaborative Regulatory Modules
Gene Clusters
Step 2
HMM
TF1
TF2
Target Genes
TF-Targets Regulatory Interaction Models

Figure 2.1: TRIM framework for inferring TF-target regulatory interaction models. TRIM has two main steps: 1) a regulatory module construction step that includes both a clustering of co-expressed genes and a topological analysis to classify independent and collaborative regulatory modules from a given large-scale TRN, and 2) a HMM model to infer TF-target interaction models in the independent and collaborative regulatory modules identified in step 1.

enriched clusters are discarded [47]. Genes in the same cluster are considered to be co-expressed. And the co-expressed and co-regulated genes are usually weighed to be regulated by the same TF(s) with a similar interaction model. So for the target genes that are regulated by the same single TF (or the same TF pair), we partition them based on whether they are in the same cluster to construct independent regulatory modules (or collaborative regulatory modules). An illustrative example is shown in Figure 2.2 TF_1 and TF_2 regulate genes g_1 and g_2, and g_1 and g_2 belong to the same gene expression cluster, so this regulatory module contains TF_1, TF_2, g_1 and g_2.

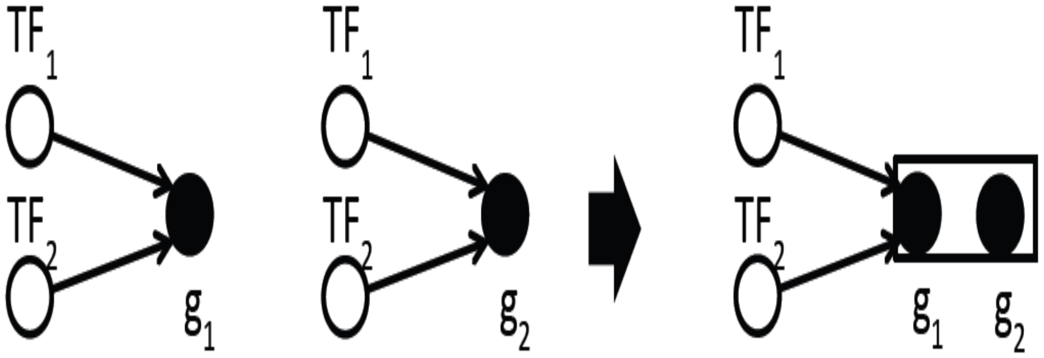


Figure 2.2: An illustrative example of regulatory module construction. TF_1 and TF_2 regulate genes g_1 and g_2, and g_1 and g_2 belong to the same gene expression cluster, so this regulatory module contains TF_1, TF_2, g_1 and g_2.

2.3.2 Designing the HMM model

In the next step (Figure 2.1 step 2), we design a new HMM model [51] to infer the regulatory interaction models for the TF-target interactions in every regulatory module detected above. For the regulatory modules with two TFs, we run the HMM model directly. For the regulatory modules with a single TF, we add a dummy TF with its expression value constantly zero. Some researchers have pointed out that designing an HMM model is a sort of art [19]. In the following text, we describe the details of how we design the structure, set the initial probabilities, and develop the updating method for emission probabilities and transition probabilities for our HMM for 2-TF collaborative regulatory modules.

Structure. To model 2-TF collaborative modules, our HMM consists of two TFs, TF_1 and TF_2, where each TF has four states (*i.e., AS*, *AN*, *RS* and *RN*), as shown in Figure 2.3. Each state emits two possible outputs, active or inactive. One can view a state as a representation of whether a particular regulatory interaction model for an individual TF-target interaction is valid (active) or invalid (inactive). In the training process, if one of the four states of TF_1 emits an active output, the

HMM will transit to a state belonging to TF_2 based on the constrains (to be introduced later), and *vice versa*. **Initial Probability.** The initial emission probabilities of the states in the HMM are set equally (if we do not have any prior information), or they can be determined by prior knowledge obtained from TF perturbation experiments. In addition, we use uniform transition probabilities, if there is an arrow in Figure 2.3

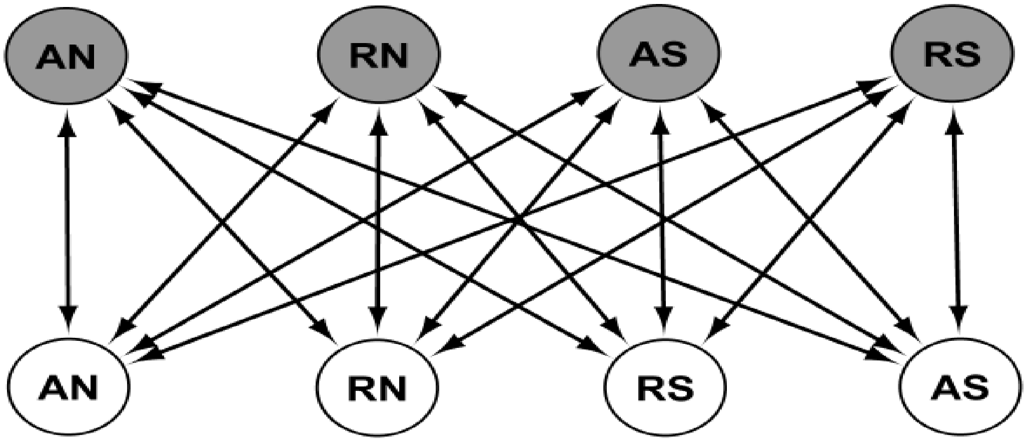


Figure 2.3: A simplified representation of the HMM model for the collaborative regulatory module with 2-TF, TF_1 and TF_2, where each TF has four states (*i.e., AS*, *AN*, *RS* and *RN*). Each state emits two possible outputs, active or inactive. One can view a state as representing whether a specific interaction model of an individual TF-target relation is valid or invalid. In the training process, if an active output is emitted from one of the four states of TF_1, the model will transit to a state belonging to TF_2 and *vice versa*.

Emission and Transition probabilities. We train the HMM model with discretized gene expression data. We discretized the gene expression levels as up or down-regulation instead using absolute absence or presence, by comparing the expression changes between two consecutive time-points to determine whether there was increase $(+1)$ / decrease (-1) [52]. For the regulatory module with multiple target genes, all the gene expression changes are used sequentially for HMM training.

In a mRNA transcriptional process, the expression change of a TF is usually earlier than the

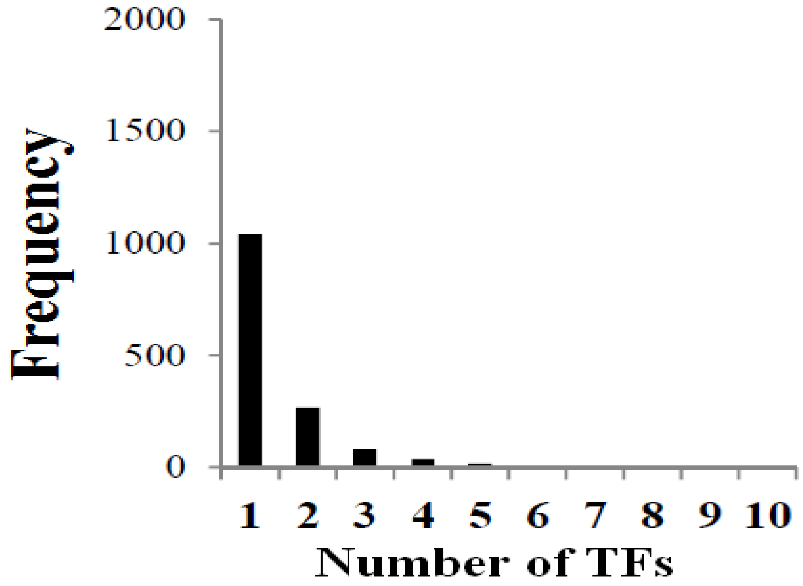


Figure 2.4: In-degree distribution of the yeast TRN for Reimand dataset.

change of its targets. Therefore, we adopt the concept of time lagging in this HMM model training process [59].

At timepoint n, the input to the HMM is a set of gene expression changes of the TFs from timepoint $n-l$ to n, where l is a time lag that is defined by the user to capture the effects of the TFs at the earlier timepoints $(n-l,\ldots,n)$ on the target gene at timepoint n. To update the emission probabilities properly, we design a set of constraints that relate the gene expression patterns of a TF and its target to the regulatory interaction model (see Table 2.1). The emission probability of a state (active/non-active) can then be updated with Equation 2.2, where b and b' are the outputs of state k and $E_k(b')$ is the probability for emitting output b' from state k [19]. The final emission probability of each state reflects the likelihood of the state being active or inactive.

$$e_k(b) = \frac{E_k(b)}{\sum_{b'} E_k(b')} \tag{2.2}$$

In a HMM, a path is a sequence of states that follows the Markov chain of hidden states, in which the probability of a state depends only on the probability of the previous state. In this way, we are able to consider the regulatory effects of the two TFs simultaneously, unlike previous works. The Viterbi algorithm is applied to effectively find the most probable path [18, 51]. The final path contains two states with a state for each TF's TF-target interaction model. For example, a path "AS-RN" means that the first TF is activator sufficient and the second TF is repressor necessary for the same target genes.

Note that the HMM is a probabilistic model which cannot assume combined events or none of the events to occur. Therefore, the model described above does not include a state for "Neither" or "Both Necessary and Sufficient (N+S)". To infer neither/N+S regulatory models, a post-processing step is required. We use the distribution of coefficient of variation (CV) of all the emission probabilities to determine whether a regulatory interaction model can be N+S or neither: if none of the probabilities are significant, the model outputs "neither"; if the probabilities of both N and S states are significant, and if there is a significant difference between the probabilities of the two states, TRIM outputs the more significant state, otherwise our model outputs N+S.

Table 2.1: Constrains for the inferring TF-target interaction models in a 2-TF collaborative regulatory module. They are based on the TF's expression direction and the response of the target gene.

TF_1	TF_2	Target-gene expression	TF-target Interaction Model
Up	Down	Up	TF_1 is Activator Sufficient or TF_2 is Repressor Necessary
Up	Down	Down	TF_1 is Repressor Sufficient or TF_2 is Activator Necessary
Up	Up	Up	At least one TF is Activator Sufficient
Down	Down	Up	At least one TF is Repressor Necessary
Up	Up	Down	At least one TF is Repressor Sufficient
Down	Down	Down	At least one TF is Activator Necessary
Up	-	Up	TF_1 is Activator Sufficient
Up	-	Down	TF_1 is Repressor Sufficient
Down	-	Up	TF_1 is Repressor Necessary
Down	-	Down	TF_1 is Activator Necessary

We show an illustrative example of our HMM in Table 2.2. In this example, TF_1 and TF_2 regulate the same target gene g. For simplicity, no time lag is used in the example ($l = 0$). Given the gene expression changes of the two TFs and the target gene, we can infer the regulatory interaction models for both of the TF-target pairs using the HMM as follows. We first initialize the active emission probabilities of all the states equally to 0.5. At time t_0, as none of the expression changes is significant, nothing is done. At time t_1, the down-regulation of both TF_2 and g triggers the active emission probability of state AN of TF_2 (Table 2.1 Row 10). The active emission probability of state (TF_2,AN) is updated by adding the new frequency and then being normalized, *i.e.,* $(0.5+1)/2 = 0.75$, while the inactive emission probability of (TF_2,AN) is 0.25. Meanwhile, the active emission probabilities of all other states are decreased to 0.25 and the inactive emission probabilities of them are increased to 0.75. See Table 2.3 for the updated emissions. At time t_2, the up-regulation of TF_1 and the down-regulation of g trigger the state RS of TF_1 (Table 2.1 Row 8). The active emission probability of (TF_1,RS) is updated to be 0.625 and the other active emission probabilities are adjusted accordingly. At time t_3, state RS of TF_1 and state AN of TF_2 are triggered given the two-TF constraint in Table 2.1 Row 2. The active emission probability of (TF_2,AN) is updated to $(0.375+1)/2 = 0.688$ and the active emission probability of (TF_1,RS) to $(0.625+1)/2 = 0.813$. See Table 2.3 for the emission probabilities of the four observations. The training process continues until all the gene expression observations are processed. For this example, the RS's active emission probability (0.675) is the highest amongst all the active probabilities of TF_1 and the RS's active emission probability (0.203) is the highest amongst all the active probabilities of TF_2 in the end. The final output for the TF-target interaction model is subject to the distribution of all the active emission probabilities in all the regulatory modules. In this example,

Table 2.2: An illustrative example of HMM training process. There are 10 observations of gene expression changes for a collaborative regulatory module with 2-TF: 0 means no significant gene expression change; 1 means significant up-regulation and -1 means significant down-regulation.

Time	**t_0**	**t_1**	**t_2**	**t_3**	**t_4**	**t_5**	**t_6**	**t_7**	**t_8**	**t_9**
TF_1	0	0	1	1	1	1	0	1	1	1
TF_2	0	-1	0	-1	0	1	-1	0	-1	0
g	0	-1	-1	-1	1	-1	0	1	0	-1

we conclude that the most possible regulatory interaction model for (TF_1, g) is repressor sufficient (RS) and for (TF_2, g) is Neither.

Table 2.3: The emission probabilities of the first four observations of gene expression changes in the illustrative example shown in Table 2.2. After processing these four observations, the model suggests that the most possible regulatory interaction model for (TF_1, g) is *RS* with probability 81% and for (TF_2, g) is *AN* with probability 68%.

	AS	**AN**	**RS**	**RN**	**AS**	**AN**	**RS**	**RN**
	t_0				t_1			
TF_1	0.50	0.50	0.50	0.50	0.25	0.25	0.25	0.25
TF_2	0.50	0.50	0.50	0.50	0.25	0.75	0.25	0.25
	t_2				t_3			
TF_1	0.125	0.125	0.625	0.125	0.062	0.062	**0.813**	0.062
TF_2	0.125	0.375	0.125	0.125	0.062	**0.688**	0.062	0.062

2.4 Experimental Results

We evaluate the performance of TRIM using the yeast regulatory network. For comparison, we applied DREM v3.0 on the same dataset. We did not compare TRIM with Yeang and Jaakkola [79] since their objective is to build a reliable TRN, and the method does not output the TF function, which is the focus of our work.

Table 2.4: Yeast cell cycle time series data that are used for model training and testing. .

Data type	Time-points
α-factor synchronization	Every 7 minutes for 2 hours
CDC-28 heat-based synchronization	Every 10 minutes for 2.67 hours
CDC-15 heat-based synchronization	Every 20 minutes for the first hour, every 10 minutes for the next 3 hours, and then every 20 minutes for the final hour
Elutriation synchronization	Every 30 minutes for 6.5 hours

2.4.1 Data preparation

Using the same statistical approach in Reimand *et al* [57], we generated a large-scale yeast regulatory network with 2,230 regulatory relations between 268 TFs and 1,509 target genes by filtering the yeast ChIP-chip binding data [42] and the binding-cite predictions [21, 46]. The in-degree distribution (number of TFs per gene) in Figure 2.4 shows that 87.8% of the target genes are regulated by one or two TFs, indicating that studying independent and 2-TF collaborative regulatory models is sufficient to cover the majority of the yeast TRN.

To train and evaluate TRIM, four widely used time-series microarray datasets from yeast cell cycle studies were collected [68]. These datasets contain 73 timepoints in total. In these experiments, yeast cells were first synchronized to the same cell cycle stage, released from synchronization, and then the total RNA samples were taken at even intervals for a period of time (see details in Table 2.4).

To decide whether a gene is significantly up or down regulated, we used a gene expression change cutoff at 0.35. In this experiment, we used three of them (CDC15, CDC28 and elu) as the training data and one (Alpha) as the testing data, since the Alpha dataset contains the most available gene expression values after applying the cutoff. A time lag $l = 2$ was used in this experiment.

For evaluating the inferred regulatory interaction models for the TF-target interactions of the independent regulatory modules, single TF knockout microarray data for yeast were collected [30]. A p-value cut-off at 0.05 was applied to determine whether a gene is significantly affected by a TF knockout [30].

2.4.2 Evaluation criteria

To test whether our method can correctly predict the TF-targets interaction models, we adopt a similar approach as used in Segal *et al* [61] to examine the distribution of the differentially expressed genes in the modules. With the regulatory interaction models learned from training data for the TF-target interactions in a 2-TF collaborative regulatory module, we can obtain the active timepoints on the testing data at which a TF functions. If the inferred regulatory interaction model is correct and the TF has the consistent function on the training and the testing data, then the gene expression correlations on the active timepoints should be significantly higher than on the whole time frame of the testing data.

Mathematically, the set of activation timepoints T_x is defined as follows: let e_i be the gene expression change (1, -1 or 0) of a TF at timepoint t_i; for each TF and its TF-target interaction model x, timepoint t_i is in T_x if and only if $Indicator(t_i) = 1$ or $Indicator(t_{i+1}) = 1$. (See Equation 2.3).

$$Indicator(t_i) = \begin{cases} 1 & \text{if } x = \text{sufficient and } e_i = 1, or \\ & x = \text{necessary and } e_i = -1, or \\ & x = \text{(necessary and sufficient) and } |e_i| = 1 \\ 0 & \text{otherwise} \end{cases} \tag{2.3}$$

To compute the gene expression correlation, we group together the target genes regulated by the same TF with the same interaction model. We then compute the gene expression correlation on each group of the target genes over their activation timepoints, and normalize them. Mathematically, given a set of genes G and a set of timepoints T, if TF TF_k is predicted to be necessary for genes in G_n, sufficient for genes in G_s and both necessary and sufficient for genes in G_b, then the correlation score of all the genes regulated by TF_k is computed with Equation 2.4.

$$COR(TF_k) = \frac{\sum\limits_{g_i, g_j \in G_n} C(g_i, g_j, T_n) + \sum\limits_{g_i, g_j \in G_s} C(g_i, g_j, T_s) + \sum\limits_{g_i, g_j \in G_b} C(g_i, g_j, T_b)}{|G_n \cup G_s \cup G_b|} \tag{2.4}$$

where $G_x \subseteq G$ and $T_x \subseteq T$; $C(g_i, g_j, T_x)$ is the Pearson correlation score between g_i and g_j ($i \neq j$) at all the activation timepoints in T_x (x can be n, s or b).

For comparison, we applied the same approach on DREM prediction results. Since DREM did not predict the effectiveness of TFs as necessary or sufficient, we grouped the target genes regulated by a TF based on the TF's dependence (activation or repression). The Pearson correlation of the same genes (grouped by TRIM prediction results) on all the timepoints were also computed.

2.4.3 Evaluation of TRIM on yeast data

In the yeast regulatory network, the 1,509 target genes were grouped into 50 clusters with Cluster 3.0 [20]. We then partitioned the yeast regulatory network into 1,051 independent regulatory modules and 275 collaborative regulatory modules with two TFs. Finally, by combining the regulatory modules accordingly (see Section 2.3.1), the total number of the independent regulatory modules is reduced from 1,051 to 640, while the number of 2-TF collaborative regulatory modules is reduced

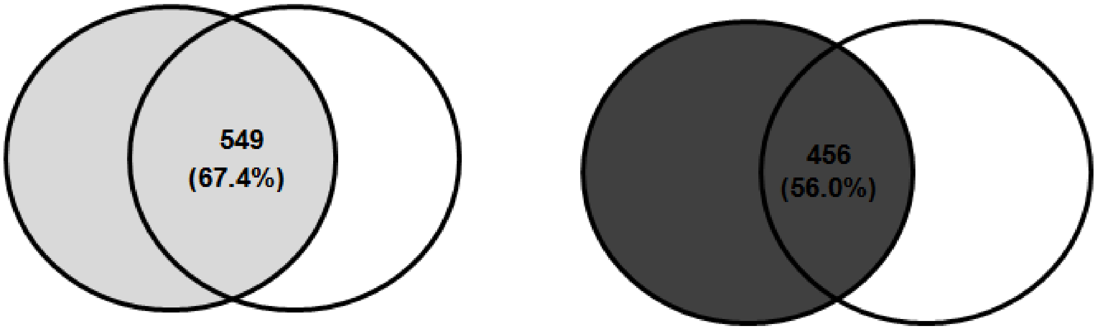


Figure 2.5: In the independent regulatory modules, the overlap between the single TF knockout supported TF-target pairs (815) and TRIM prediction results (833) for the functional roles (activation or repression) is 549, greater than the overlap between knockout and the results of DREM (873), which is 456.

Table 2.5: Summary of TRIM's predictions for independent and two TFs regulatory modules on yeast and Arabidopsis.

	Activator	**Repressor**	**Unknown**
A) Independent regulatory modules of yeast			
Necessary	37	35	3
Sufficient	68	50	17
Necessary & Sufficient	228	179	216
Neither	218		
B) 2-TF collaborative regulatory modules of yeast			
Necessary	74	61	0
Sufficient	121	61	0
Necessary & Sufficient	6	5	0
Neither	212		
C) 2-TF collaborative regulatory modules of of Arabidopsis			
Necessary	180	136	0
Sufficient	126	101	0
Necessary & Sufficient	6	0	0
Neither	87		

from 275 to 257. In the 640 independent regulatory modules, there are a total of 1,051 TF-target pairs. TRIM was able to infer the TF-target interaction models for 833 pairs (see Table 2.5A). On the same dataset, DREM was able to infer 873 TF-target relations with its p-value cut-off at 0.05.

We used the single TF knockout microarray data to directly verify our inferred TF-target interaction models for the independent regulatory modules. First, for the functional roles (activation or repression) of TFs, our TRIM successfully predicted 549 out of 815 models with a successful rate of 67.4%, which is clearly higher than the results of DREM (56.0%), as shown in Figure 2.5. Second, for evaluating the inference performance of the logical roles of TFs, with knockout data, we can only look at the "necessary" logical role: a TF is necessary for its target genes if the expression values of the target genes are significantly changed when the TF is knocked out. In the single TF knockout data, a total of 815 independent regulatory modules were considered as necessary. Among them, TRIM predicted 682 pairs to be necessary or necessary and sufficient with a success rate of 83.6%. (We are unable to perform a similar comparison for DREM since DREM does not predict necessary TFs).

In addition, the gene expression correlation scores of the TF-target pairs that are predicted by both TRIM and DREM are shown in Figure 2.8 (see Section 2.4.2 for the score computation). The median correlation score for TRIM (0.52) is clearly higher than that of DREM (0.46) and than using all the timepoints (0.29). To further evaluate the performance of TRIM on independent modules, we estimated the mRNA synthesis rate of each gene by applying decay adjustments [50] on the training and testing data:

$$\mu_g = e_g(\alpha + \lambda_g) \tag{2.5}$$

where μ_g is the synthesis rate of gene g; e_g is the gene expression of g; α is a consistent variable representing the growth rate of yeast; λ_g is the decay rate of each gene g, which could be estimated with the mRNA half-life (L_g) for each gene with equation $\lambda_g = ln(2)/L_g$, where the mRNA half-life data was obtained from [50]. For the growth rate of yeast (α), we used the same value as used in [50], *i.e.,* $ln(2)/cell\_cycle\_length$ and the cell cycle length is set to 150 minutes. After applying the decay adjustments, the median of the expression correlation scores of TRIM further increased from 0.52 to 0.57, while the scores of DREM and all-timepoints are almost unchanged (0.40 and 0.31 respectively, see Figure 2.10, indicating that by focusing on the mRNA synthesis rates, TRIM model is able to predict the TF-target interaction models more precisely. (the detailed correlation values are shown in Figure 2.6). In the 540 2TF-target pairs in the 257 2-TF collaborative regulatory modules, TRIM was able to infer the interaction models for 328 pairs. Table 2.5B shows the summary of the number of interaction models predicted by TRIM for 2-TF modules. On the same dataset, DREM was able to infer 440 TF-target pairs.

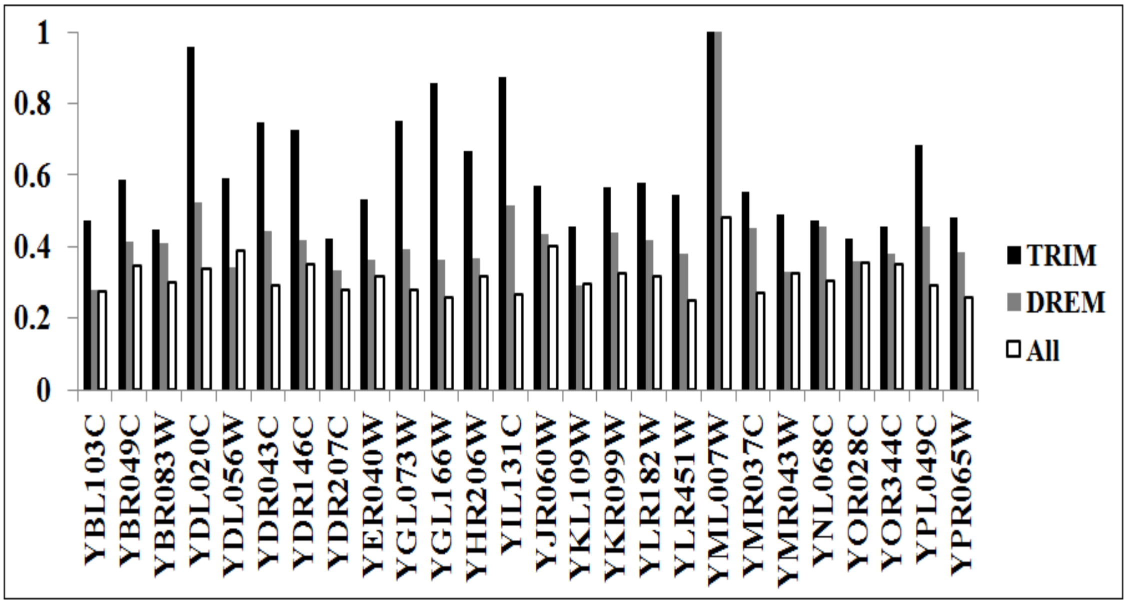


Figure 2.6: A histogram of gene expression correlation of 26 TFs in the independent regulatory modules of yeast using TRIM, DREM or all time-points after applyng decay adjustments. These 26 TFs are the intersection between the results of DREM and TRIM.

Figure 2.9 shows the expression correlation scores of the 2TF-target pairs that are predicted by both TRIM and DREM. The median correlation score for TRIM (0.87) is significantly higher than that of DREM (0.54) and than using all the timepoints (0.27) We also applied similar decay estimation approach on the 2-TF collaborative modules and the result is consistent with the single TF module (see Figure 2.7).

In summary, the experimental results on the yeast data showed that TRIM can successfully

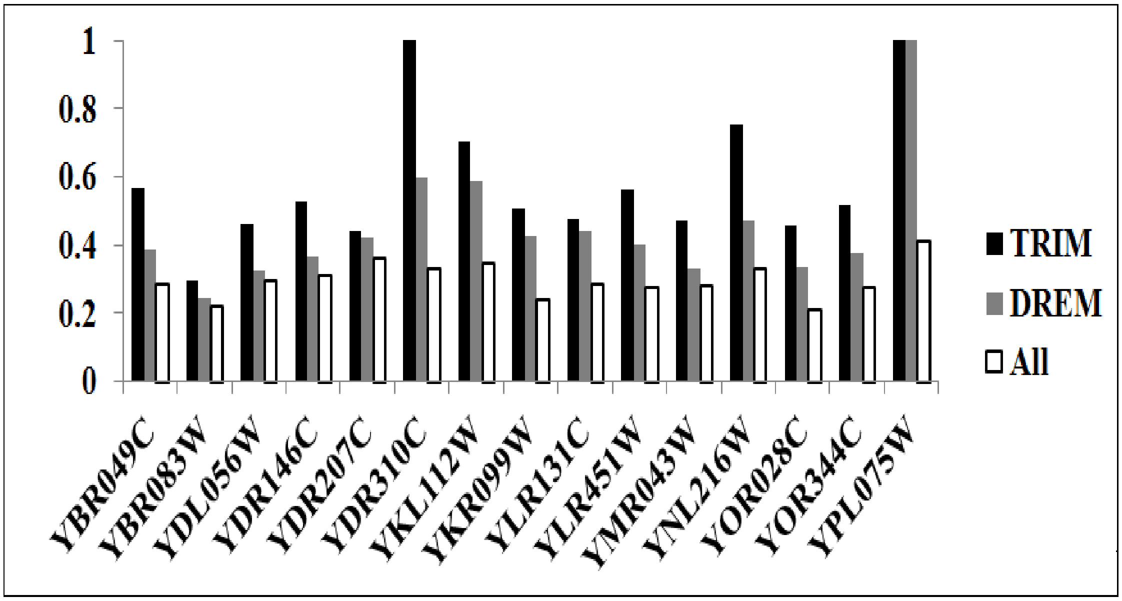


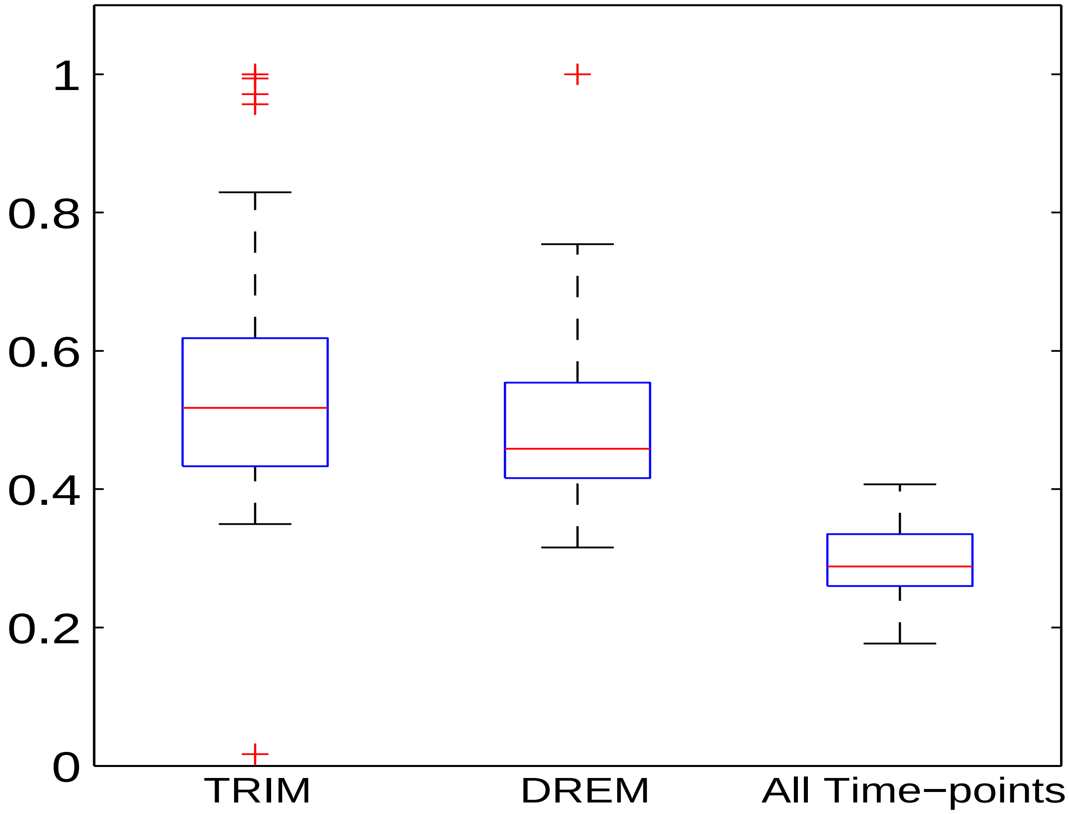
Figure 2.7: A histogram of gene expression correlation of 15 TFs in the 2-TF collaborative regulatory modules of yeast using TRIM, DREM or all time-points after applying decay adjustments. These 15 TFs are the intersection between the results of DREM and TRIM.

predict TF-target interaction models for both independent and 2-TF collaborative modules.

2.5 Application of TRIM on Arabidopsis data

In order to maintain a stable intracellular environment, living cells utilize complex and specialized transcriptional regulatory systems to react against a variety of external perturbations, such as temperature change, drought, UV, *etc.* Many of the adaptive

mechanisms contributing to cellular homeostasis operate through TRNs to regulate the expression of anti-stress genes [82]. The key step to understand the TRN behavior is therefore to explore the roles of relevant TFs by using time-series gene expression data under different stress conditions. In this study, we applied TRIM to infer the roles (i.e. interaction models) of TFs in 2-TF regulatory modules of Arabidopsis TRN under 8 different abiotic stress conditions. The objective of this study is to infer the TF-target interaction models in Arabidopsis TRN with all the available

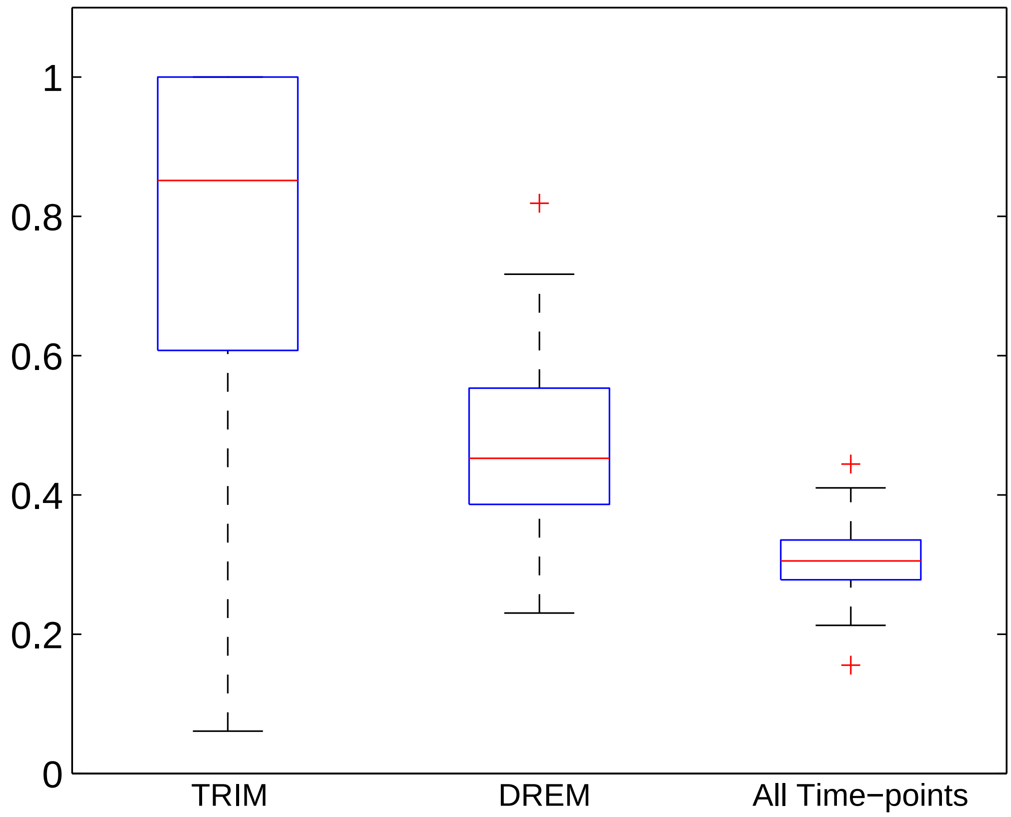


(a)

Figure 2.8: Distribution of gene expression correlation of the independent regulatory modules. For interpretation of the references to color in this and all other figures, the reader is referred to the electronic version of this dissertation.

abiotic stress data, so that by looking at the gene expression patterns (as shown in Figure 2.11), we are able to tell whether a TF is a general abiotic stress TF, a specific abiotic stress TF, or a TF that does not function under abiotic stress. In addition, since our TRIM (like all the other current algorithms) assumes that TF-target interaction models will remain consistent under different abiotic stress conditions, we also need to verify whether this assumption holds for the various biological functions of the target genes.

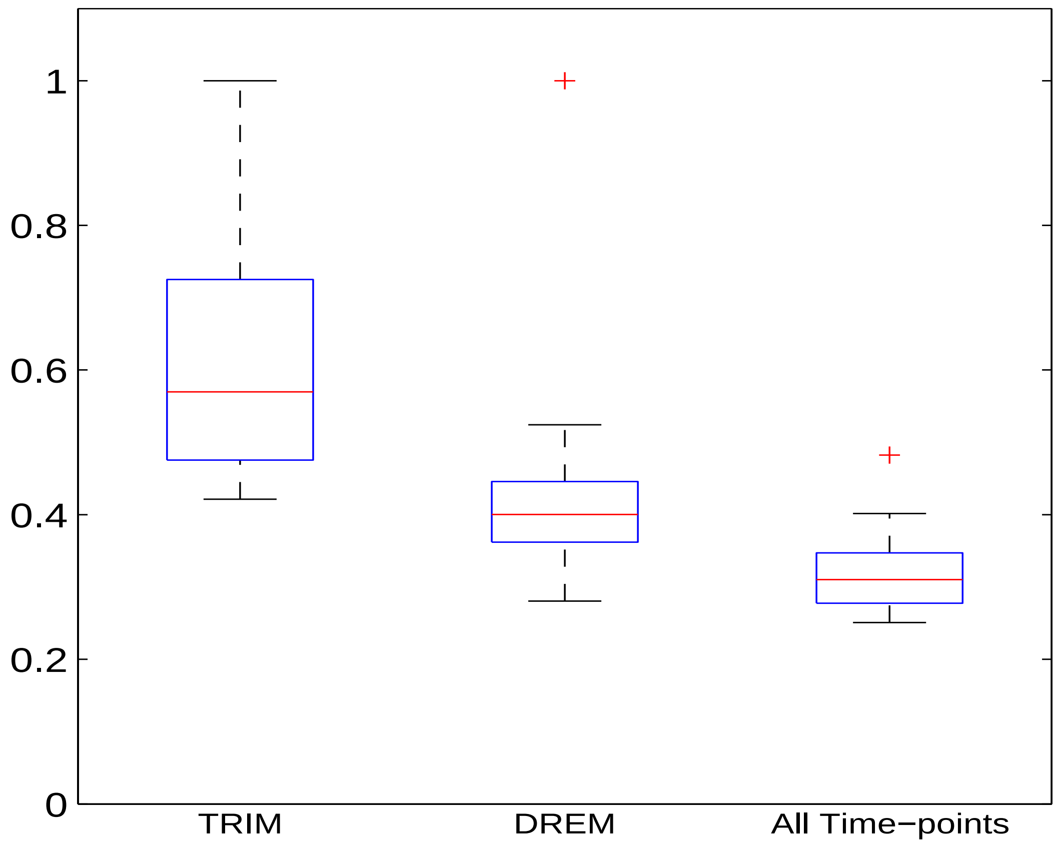
We obtained Arabidopsis regulatory data from AtRegNet, which contains 11,355 direct interactions between TFs and target genes [53]. In AtRegNet, a direct interaction means either a TF binds directly to the target gene (detected by electromobility shift assay, yeast one-hybrid, or ChIP), or



(a)

Figure 2.9: Distribution of gene expression correlation of 2-TF collaborative regulatory modules

a TF directly regulates the target gene based on use of transgenic plants expressing an inducible TF-GR fusion protein. We partitioned the Arabidopsis genes in AtRegNet into 50 clusters with Cluster 3.0. A total of 8 abiotic stress data sets (Cold, Drought, Genotoxic , Heat, Osmotic, Salt, UVB and Wounding) were collected from AtGenExpress to train and test our TRIM's performance in Arabidopsis [38]. The datasets were filtered for significant variability in mRNA expression using Bonferroni corrected p-values at 0.05. For each experimental run, three of the data sets were reserved to evaluate the output of the model, while the remaining five were used for model training. In the experiment, six runs were conducted each with a different combination of training and testing data (see detail in Table 2.6). A time lag $l = 2$ was used in this experiment. Note that in combining data from different abiotic stress experiments, we have assumed that the TF-target



(a)

Figure 2.10: The independent regulatory modules after decay adjustments. For interpretation of the references to color in this and all other figures, the reader is referred to the electronic version of this dissertation

Table 2.6: The design of the 6 experiments on Arabidopsis abiotic stress data. X: testing data; T: training data

Round	ID	Cold	Drought	Genotoxic	Heat	Osmotic	Salt	UVB	Wounding
1	DWG	T	X	X	T	T	T	T	X
2	WSG	T	T	X	T	T	X	T	X
3	CSG	X	T	X	T	T	X	T	T
4	CSH	X	T	T	X	T	X	T	T
5	DOG	T	X	X	T	X	T	T	T
6	DOU	T	X	T	T	X	T	X	T

interaction models will remain consistent under different abiotic stress conditions. This is also a common assumption by previous researchers [38, 39]. In this study, we compare the results of the TRIM model across multiple combinations of training and testing data. If our hypothesis is true, the interaction model and subsequent correlation scores given by our model should be consistent across all groups and the results would not be sensitive to how we grouped the data. Otherwise, it means that the TF-target interaction model may vary under different conditions. In this study, we found that the hypothesis is true for selected classes of TFs, *e.g.,* developmental TFs.

On average, our TRIM identified 318 2-TF collaborative regulatory modules involving 32 TFs (the prediction results is shown in Table 2.5C). 10 of these TFs had sufficient data to be analyzed by gene expression correlation. Detailed results are shown in Table 2.7. In all, we found that the developmental TFs were the most consistent in terms of prediction and scoring, though their correlation scores were not always the highest on average. The only exception to this rule, APETALA2, is involved in seed development and thus, we would expect it to be more differentially expressed, since the production of seeds involves a reproductive decision which is likely to be stress sensitive. Other TFs, such as specific stress response genes and those associated with photosynthetic processes, are less consistent.

We would also expect general stress response factors to show very consistent predictions and score highly with the given data, yet they are absent from our dataset. However, we missed these TFs because they are involved in more complicated regulatory modules with 3 or more TFs and extending TRIM to *k*-TF modules. In the following chapter, we propose a new model (mTRIM) that handles *k*-TF modules, that would capture their behavior.

In addition to grouping these TFs by function, we analyzed 3 individual TFs (ATGL1, WRKY53 and RD26) in depth. We show their Pearson correlation distributions in Figure 2.11. The TF with the highest average correlation score, ATGL1, a developmental gene involved in trichome pattering, is an example of a case where our hypothesis held, as $\sim$80% of the interactions that could be predicted were either AN or RN across all six experiments. The correlation scores were the highest when the variable modules were predicted as either AN or RN. Alternatively, the TFs with

Table 2.7: The prediction results of the 10 TFs on Arabidopsis abiotic stress data with TRIM

Locus	**Gene**	**High** (> 0.6)	**Low**(< 0.3)	**Gene Function**
AT3G27920	ATGL1	DWG, WSG, CSG, CSH, DOG	N/A	Development,Trichome Pattering
AT1G14350	FLP/MYB124	N/A	N/A	Development, Guard Cell Differentiation
AT1G24260	SEPAL LATA3	DOG	N/A	Development, Flower Development
AT2G20180	PIF1	N/A	N/A	Development, Regulation of Seed Germination
AT4G36920	APETA LA2	WSG, CSH, DOG	DWG, CSG	Development, Seed Development
AT3G47640	PYE	WSG, DOG, DOU	DWG, CSG, CSH	Stress Response, Iron Starvation
AT4G23810	WRKY53	N/A	DOG,DOU	Stress Response, Response to Chitin and Bacteria
AT4G27410	RD26	DOG, DOU	DWG, WSG, CSG, CSH	Stress Response, Response to Dessication
AT1G09530	PIF3	DWG,WSG	CSG, DOU	Photosynthetic, Red Signalling /Anthrocyanian Metabolism
AT5G11260	HY5	DWG	DOG*	Photosynthetic, Red Signalling (via PHYA)

the two lowest average scores, WRKY53 and RD26, biotic and drought stress associated regulators, showed very inconsistent prediction patterns. The highest Pearson correlation scores of RD26 occur when drought was included in the evaluation set. This is confirmed by the fact that there is little difference in the correlation whether we apply TRIM or use all timepoints to calculate the score. Hence, RD26 is an example of condition sensitive TF which violates our assumption.

WRKY53, on the other hand, has low and similar correlation scores whether we use activation timepoints predicted by TRIM or all the timepoints, which is true across all six runs. Therefore, the issue here is most likely not sensitivity to particular grouping of data, but rather that the data is uninformative for this particular case. This conclusion is not unexpected given WRKY53 responds primarily to biotic rather than abiotic stress.

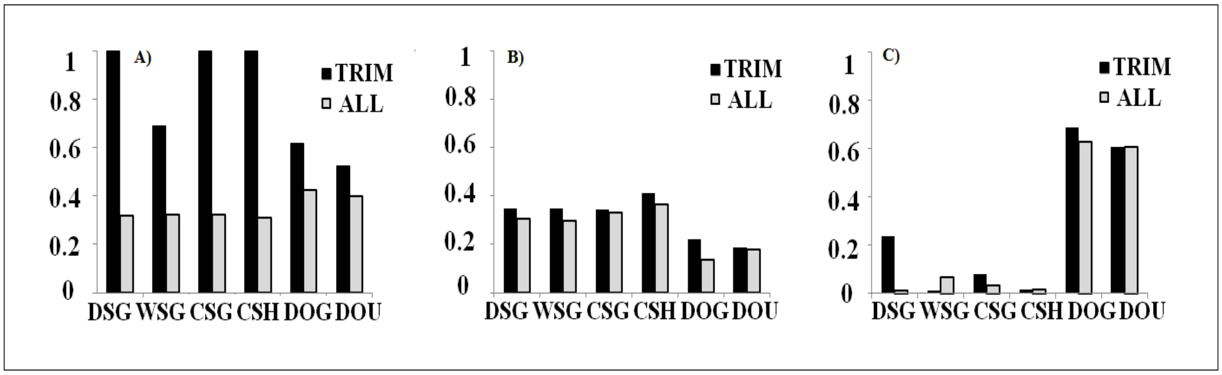


Figure 2.11: Pearson correlation of target genes of (A) ATGL1, (B) WRKY53 and (C) RD26 using different combinations of training and testing data with TRIM.

2.6 Conclusions

Revealing the mechanisms of the transcriptional regulatory programs in TRNs is essential for understanding the complex control by which genes are expressed in living cells. In this chapter, we model the interactions between the TFs and the target genes in terms of both the TF-target interaction's function (activation or repression) and its corresponding logical role (necessary and/or sufficient). Based on the characterizations proposed by Yeang and Jaakkola [79], we define the combinatorial regulatory interaction models for possibly multiple TF-target interactions in TRNs.

We used DNA-protein binding and gene expression data to construct regulatory modules for inferring the transcriptional regulatory interaction models for the TFs and their corresponding target genes. Our TRIM algorithm is based on a HMM and a set of constraints that relate gene expression patterns to regulatory interaction models. We have shown in this chapter how to apply TRIM to infer the transcriptional regulatory interaction models for TFs in collaborative regulatory modules involving two TFs. It can thus be used to help predict the phenotype of TF double-knockouts to reduce the number of double knock-out or over-expression experiments needed.

Chapter 3

Inferring K-TF Regulatory Interactions

3.1 Introduction

In the previous chapter, we defined an individual-TF regulatory interaction in terms of two properties: the TF's functional role as an activator or a repressor, and its logical role as being necessary or sufficient (see Table 1.1) [61, 79]. The categories in the TF's functional and logical roles are combinable; they can be activator necessary (AN), activator sufficient (AS), or activator necessary and sufficient (ANS).

We explained how recent biotechnology (such as ChIP [54] and yeast one-hybrid [15]) have been applied to uncover TF-target binding relationships [16, 58] to reconstruct draft regulatory circuits at a systems level [22, 57, 79]. For example, to identify regulatory interactions *in vivo* and consequently reveal their functions, TF single/double knockouts and over-expression experiments have been systematically carried out [29]. We also showed that high order genetic variations are needed for precise inference of transcriptional regulations.

Considering the prohibitive costs and the tremendous number of possible combinations of higher-order gene knockouts, it is currently impossible for researchers to examine all of possible gene knockout combinations experimentally. One solution to this problem is to select only the TF groups that are most likely to bring about the phenotypic change. To do this, we need to understand the regulatory interactions between multiple TFs to regulate their set of common target genes. However, this is also a difficult task, because when multiple TFs simultaneously or sequentially control their target genes, a single gene responds to merged inputs, resulting in complex gene expression patterns [5, 6]. The exhaustive approach requires enumerating all TF combinations, which, given the high complexity of combinatorial, is simply impractical at the whole genome level.

In the previous chapter, we described TRIM [3], a Hidden Markov model that relate gene expression patterns to regulatory interactions, in order to solve a relatively simpler sub-problem that considers only two TFs.

In this chapter, we extend the 2-TF regulatory interaction to multiple-TF regulatory interaction. A multiple regulatory interaction is defined as a group of TFs that collaborate to control the expression levels of the same target genes. The directions of all the TFs in the group, therefore, form a transcriptional regulation pattern of the target genes. Similarly for TFs that are repressors, they can be RN, RS or RNS [4]. In a multiple-TF regulatory interaction, a group of TFs collaborate to control the expression levels of the same target genes. The directions of all the TFs in the group, therefore, form a transcriptional regulation pattern of the target genes.

To predict regulatory interactions for all possible collaborative TFs, we propose an algorithm called "mTRIM" (multiple Transcriptional Regulatory Interaction Mechanism) in this chapter. By uncovering the regulatory interactions in terms of their directions (activation or repression) and corresponding logical roles (necessary and/or sufficient) from gene expression and TF-DNA binding data, mTRIM identifies TF groups that are collaboratively responsible for target gene expressions. Such inferences may provide high-quality candidate sets for further experimentally detecting the collaborative functions of gene regulations that are largely unknown [5].

Yeang and Jaakkola [79] attempted to characterize the combinatorial regulation of multiple-TF regulatory interactions using a heuristic approach to measure how well a regulatory module fits the associated binding and gene expression data with a log-likelihood function (See details in Section 2.1). However, as stated in [22], a TF usually functions at specific “activation time points” instead of throughout the entire time course, meaning that the identification of regulatory interaction modules should be focused on activation time-points rather than the entire time frame.

To derive dynamic regulatory networks that associate TFs with target genes at their activation time-points, an algorithm called DREM was proposed [22]. he method mainly works by identifying bifurcation time-points where the expression of a subset of genes diverges from the rest of the genes (See details in Section 2.1). However, DREM does not infer the logical roles of the TFs (*i.e.,* whether a specific TF is necessary or sufficient for regulating a set of target genes). Such knowledge is extremely useful for designing high-order genetic variation experiments to understand the complex regulatory mechanisms of biological processes.

In this chapter, we demonstrate that by inferring the logical roles of the TFs, the target gene co-expression correlation increased significantly.

TRIM is an HMM based model which was developed to infer the collaboration of at most 2 TFs that regulate the same target genes [3]. In the HMM, the functions of a TF are hidden states. The model starts with random priors, and then is iteratively trained using EM till convergence. Since each possible function of a TF is a node in the HMM, there are four nodes (AS, AN, RS, and RN) for each TF. With the design of HMM (and the limited training data), the number of TFs the model can handle is greatly limited.

The enumeration of all TF combinations is clearly an NP problem. Therefore, we focused on the most important biological problem (i.e., 2-TF combination) and therefore “hardcoded the problem in TRIM. In this paper, alternatively, we solve the efficient problem by using association rule mining algorithms which is capable to handle a large amount of data or high-level combinations.

In this chapter, we propose a new model mTRIM [2]for inferring regulatory interactions for multiple TFs with an EM-based Bayesian inference approach [18, 71] and a modified bottom-up association rule mining method. Experimental results evaluated with yeast genetic interactions, TF knockouts and a synthetic dataset shows that our algorithm is significantly better than the existing ones.

3.2 Methods

mTRIM is developed to efficiently infer regulatory interactions for all possible collaborative TFs in a TRN. The feasibility is achieved in two steps. First, an EM-based Bayesian inference approach is developed to identify all the significant individual TF regulatory interactions, meaning that individual TFs that can regulate the target genes independent to the existence of other TFs. For the TFs which require collaborations with other TFs to drive the target genes, or are actually non-deterministic (meaning lack of clear evidence of regulation), their p-values are insignificant. They are considered as the inputs of the second step.

Second, in order to identify the collaboration of k TFs ($k \geq 2$), *i.e.,* k-TF regulatory interaction, a bottom-up association rule mining approach is developed. While the significant TF groups are reported to the users, the insignificant ones are joined with each other to mine $(k+1)$-TF regulatory interactions. It should be noted that unlike the conventional association rule mining which seeks the longest possible patterns, mTRIM outputs the shortest significant results, in that the goal of mTRIM is to discover the smallest group of TFs that can regulate the target genes, so that biological experiments with high-order genetic variations can be subsequently carried out for the understanding of the behavior of TRNs.

3.2.1 Concepts

A TRN can be represented as a directed graph in which each node is a TF or a gene, and each edge pointing from a TF to a gene represents a regulation relationship between them. In many organisms, in-depth transcriptome analysis has revealed the modular architecture of gene expression [33]. A regulatory module is a self-consistent regulatory unit $R(TF, G, I)$ representing a set of co-expressed genes $G = \{g_1, g_2, \ldots, g_n\}$ regulated in concert by a group of TFs in $TF = \{tf_1, tf_2, \ldots, tf_m\}$ that govern the target genes' behaviors via regulatory interaction I [61]. An example of the regulatory module is shown in Figure 3.1.

A regulatory interaction $I =< h_{tf_1}, \ldots, h_{tf_i}, \ldots, h_{tf_m} >\Rightarrow h_g$ (which is the final output of mTRIM) is defined as a set of TFs $\{tf_1, \ldots, tf_m\}$ co-regulating a set of genes $\{g_1, \ldots, g_n\}$, where h_{tf_i} is the behavior of TF i; h_g is the behavior of all the target genes in R, and $h_x \in \{\uparrow, \downarrow, -\}$, meaning up-express, down-express and no change respectively. For example, if $tf_1 \uparrow$ and $tf_2 \downarrow$ always cause the target genes g_1 and g_2 to be up-regulated, the regulatory interaction is $< tf_1 \uparrow, tf_2 \downarrow > \Rightarrow g \uparrow$. For individual regulatory interactions, $I \in \{AN, AS, RN, RS, ANS, RNS\}$. In this work, we assume that

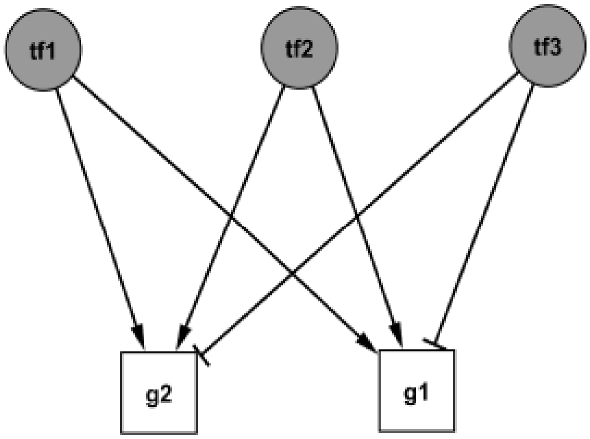


Figure 3.1: an illustrative example of regulatory interaction in a TRN, in which three TFs collaboratively co-regulate two target genes

a regulatory interaction is consistent in the context of transcriptional control as long as the experimental conditions are unchanged. Note that binaries gene expression values are used in mTRIM, since TF activity is not always proportional to its mRNA abundance [25].

3.2.2 mTRIM Step 1: Inferring Individual Regulatory Interactions

To solve a relatively easier problem of inferring the regulatory interactions for each individual TF and to prepare input for multi-TF regulatory interaction inference, an EM-based Bayesian inference algorithm has been developed [18, 71]. To define the probabilities, we followed the definitions in [18]. Eq 3.2 represents the prior probability of the interaction model I_m. Eq 3.3 represents the probability of gene expression correlation between TFs and targets given the interaction model I_m. In the Bayesian model, the training dataset is a matrix that contains gene expression levels of TFs and their targets, from which $\Gamma(I_m)$ is estimated using Eq 3.4. And then, the likelihood is calculated using Eq 3.3. The prior probabilities are randomly assigned initially, which are estimated using the frequency of I_m. In each iteration, the posterior probabilities and the frequency of I_m are updated. The iteration will continue till the posterior probabilities converge.

To define the probabilities, we followed the definitions in [18]. Eq 3.2 represents the prior probability of the interaction model I_m. Eq 3.3 represents the probability of gene expression correlation between TFs and targets given the interaction model I_m. In the Bayesian model, the training dataset is a matrix that contains gene expression levels of TFs and their targets, from which $\Gamma(I_m)$ is estimated using Eq 3.4. And then, the likelihood is calculated using Eq 3.3. The prior probabilities are randomly assigned initially, which are estimated using the frequency of I_m. In each iteration, the posterior probabilities and the frequency of I_m are updated. The iteration will continue till the posterior probabilities converge. Let *Pos* be the posterior probability of a TF tf_m to have a specific regulatory interaction I_m in regulatory module R_k, where $I_m \in \{AN, AS, RN, RS\}$ (ANS and RNS will be discussed later). To infer *Pos*, both the prior probabilities *Pri* and the likelihood *Lk* of the same TF need to be computed, given that:

$$Pos(tf_m, R_k, I_m) = Pri(I_m) \times Lk(tf_m, R_k, I_m) \tag{3.1}$$

where $Pri(I_m)$ is the prior probability of regulatory interaction I_m (defined in Eq 3.2) and the likelihood $Lk(tf_m, R_k, I_m)$ is defined in Eq 3.3.

The prior probability $Pri(I_m)$ captures how likely a given interaction I_m exists given the background of all of the other TFs:

$$Pri(I_m) = \frac{fre(I_m)}{|R| \times |TF|} \tag{3.2}$$

where $fre(I_m)$ is the frequency of regulatory interaction I_m in all of the regulatory modules, $|R|$ is the number of the regulatory modules, $|TF|$ is the number of TFs, and $I_m \in \{AS, RS, AN, RN\}$.

Given the definition of a regulatory interaction, the likelihood $Lk(tf_m, R_k, I_m)$ indicates how likely tf_m in R_k has regulatory interaction I_m, which is defined by the expression level changes of the TF and its targets:

$$Lk(tf_m, R_k, I_m) = \frac{\sum_{t=1}^{T-1} \sum_{n=1}^{|G|} \Gamma(I_m)}{\sum_{r=1}^{|R|} \sum_{m=1}^{|TF|} \sum_{t=1}^{T-1} \sum_{n=1}^{|G|} \Gamma(I_m)} \tag{3.3}$$

where T is the number of time-points in the training data, $|G|$ is the number of genes in regulatory module R_k, and $\Gamma(I_m)$ is defined as:

$$\Gamma(I_m) = \begin{cases} 1 & \text{if } I_m = \text{AS and } (tf_m \uparrow \text{ and } g \uparrow), or \\ & \text{if } I_m = \text{RS and } (tf_m \uparrow \text{ and } g \downarrow), or \\ & \text{if } I_m = \text{AN and } (tf_m \downarrow \text{ and } g \downarrow), or \\ & \text{if } I_m = \text{RN and } (tf_m \downarrow \text{ and } g \uparrow) \\ 0 & \text{otherwise} \end{cases} \tag{3.4}$$

An expectation-maximization (EM) algorithm is adopted to maximize the posterior probabilities $Pos(tf_m, R_k, I_m)$. The EM model is initialized with each TF assigned a random regulatory interaction. In the expectation step, we compute the likelihood of each TF to be a specific interaction using Eq 3.3. Consequently, the posterior probabilities of interactions for every TF is updated with Eq 3.1. As a result, each TF is assigned with the regulatory interaction with the highest posterior probability. In the maximization step, we maximize the scoring function $S(R_k) = \sum_{m=1}^{|TF|} \sum_{n=1}^{|G|} \Gamma(I_m)$ for each regulatory module R_k, which measures how the interaction of each TF in R_k matches the target gene expression changes. Note that in the iteration the priors are updated but the likelihoods are constant.

Finally, in order to determine whether I_m is "necessary and sufficient" (ANS and RNS) or "no decision", the following strategy is adopted: if none of the posterior probabilities are significant, the output is "no decision"; if the probabilities of both N and S states are significant, and there is no significant difference between them, the output is *ANS* or *RNS* depending on the target gene expression direction; otherwise the output is the regulatory interaction with the highest posterior probability.

An illustrative example is shown in Figure 3.1, in which tf_1, tf_2 and tf_3 regulate target genes g_1

Table 3.1: Illustrative example of time-series gene expression data for the genes in Figure 3.1.

	$\mathbf{t_0}$	$\mathbf{t_1}$	$\mathbf{t_2}$	$\mathbf{t_3}$	$\mathbf{t_4}$	$\mathbf{t_5}$	$\mathbf{t_6}$	$\mathbf{t_7}$	$\mathbf{t_8}$	$\mathbf{t_9}$	$\mathbf{t_{10}}$	$\mathbf{t_{11}}$
$\mathbf{tf_1}$	↑	↑	↑	↑	↑	↑	↓	↓	↓	↓	↓	↓
$\mathbf{tf_2}$	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑
$\mathbf{tf_3}$	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓
$\mathbf{g_1}$	↑	↑	↑	↑	↑	↓	↓	↓	↓	↓	↓	↓
$\mathbf{g_2}$	↑	↑	↑	↑	↑	↓	↓	↓	↓	↓	↓	↓

and g_2, and they all belong to the same regulatory module R_k. With the gene expression changes in Table 3.1, we start with equal prior probabilities, *i.e.,* $Pri(AS) = Pri(RS) = Pri(AN) = Pri(RN) = 0.25$, so $Lk(tf_1, R_k, AN) = 12/26 = 0.461$, (Eq 3.3). After 10 iterations, in the expectation step, $Pri(AN)$ is updated to 0.70 (Eq 3.2), hence $Pos(tf_1, R_k, RS) = 0.70 \times 0.461 = 0.323$ (Eq 3.1). In the maximization step, we have $< tf_1 \downarrow > \Rightarrow g \downarrow$, because the maximum posterior probability is assigned to AN with p-value 0.05 (see Table 3.2 row 1).

3.2.3 mTRIM Step 2: Mining Multiple-TF Regulatory Interactions

Besides the individual TF regulatory interactions, a significant portion of TFs collaboratively work together to regulate the same target genes. In order to identify these multiple-TF regulatory interactions, a new association rule mining approach has been developed. Instead of using the concepts of support and confidence that are commonly used in a conventional association rule mining application [1], we define an affinity scoring function (called $AfnScore$) according to the gene expression agreement between the TF groups and their target genes, to meet the biological meaning of a multiple-TF regulatory interaction.

Mathematically, $AfnScore$ of each candidate regulatory interaction $I = < h_{tf_1}, h_{tf_2}, \ldots, h_{tf_m} > \Rightarrow$

Table 3.2: Illustrative example of regulatory interaction identification on the TRN in Figure 3.1.

	Regulatory Interaction	$AfnScore$	**p-value**
$\mathbf{I_0}$	$< tf_1 \downarrow > \Rightarrow g \downarrow$	-	0.05
$\mathbf{I_1}$	$< tf_1 \uparrow, tf_2 \uparrow > \Rightarrow g \uparrow$	0.347	0.06
$\mathbf{I_2}$	$< tf_2 \uparrow, tf_3 \downarrow > \Rightarrow g \uparrow$	0.173	0.09
$\mathbf{I_3}$	$< tf_1 \uparrow, tf_3 \downarrow > \Rightarrow g \uparrow$	0.347	0.06
$\mathbf{I_4}$	$< tf_1 \uparrow, tf_2 \uparrow, tf_3 \downarrow > \Rightarrow g \uparrow$	0.347	0.04

h_g is calculated with:

$$AfnScore(I) = \frac{P(h_{tf_1}, h_{tf_2}, \ldots, h_{tf_m}, h_g) * P(h_g)}{P(h_{tf_1}, h_{tf_2}, \ldots, h_{tf_m})} \tag{3.5}$$

where $P(x)$ is the number of times that x appears in the given time series gene expression dataset divided by the product of the total number of time points and the total number of target genes. The p-value of each candidate regulatory interaction is computed by considering the distribution of $AfnScore$ for the regulatory interactions with the same number of TFs. Only the candidate interactions with p-values smaller than 0.05 are reported to the user. Specifically, if all the TFs in I are up-regulated, the TFs are “sufficient”; if they are all down-regulated, the TFs are “necessary”; otherwise, each TF acts differently to drive the target genes to the same direction.

To identify all the significant k-TF regulatory interactions, the new association rule mining algorithm starts with an empty set Q_k and all the insignificant $(k-1)$-TF interactions saved in P_{k-1} (see pseudocode in Algorithm 1 line 1). For interactions $I_1 =< h_{tf_1}, \ldots, h_{tf_{k-1}} >\Rightarrow h_g$ and $I_2 =< h'_{tf_1}, \ldots, h'_{tf_{k-1}} >\Rightarrow h'_g$ in P_{k-1}, we combine them and compose a new interaction I_{12} (line 3), if I_1 and I_2 are combinable. We define that I_1 and I_2 are combinable if and only if they satisfy the conditions that $h_g = h'_g$, $h_{tf_i} = h'_{tf_i}$ (for $i = 1, 2, .., k-2$) and $h_{tf_{k-1}} \neq h'_{tf_{k-1}}$. If none of the $(k-1)$-TF subsets of I_{12} is significant (line 2-8), I_{12} is added to candidate set C and its $AfnScore$ is computed. Finally, we compute p-values for all of the k-TF candidates in C using t-test, report all of the significant regulatory interactions to the user, and save all the insignificant ones to P_k for the identification of the $(k+1)$-TF regulatory interactions (line 9-17).

Algorithm 1 Procedure Generating Significant Patterns

```
Input:  $Q_{k-1}$: Set of significant $(k-1)$-TF regulatory interactions
        $P_{k-1}$: Set of insignificant $(k-1)$-TF regulatory interactions
        $\theta$: p-value threshold
Output: $Q_k$: Set of significant $k$-TF regulatory interactions
        $P_k$: Set of insignificant $k$-TF regulatory interactions
 1: $Q_k \leftarrow \emptyset$; $P_k \leftarrow \emptyset$; candidate set $C \leftarrow \emptyset$;
 2: for all $I_1$, $I_2 \in P_{k-1}$ with $I_1 =<h_{tf_1},..,h_{tf_{k-2}},h_{tf_{k-1}}>\Rightarrow h_{g_1}$, $I_2 =<h_{tf_1},..,h_{tf_{k-2}},h_{tf'_{k-1}}>\Rightarrow h_{g_2}$,
    and $h_{g_1} = h_{g_2}$ do
 3:    $I_{12} \leftarrow <h_{tf_1},...,h_{tf_{k-2}},h_{tf_{k-1}},h_{tf'_{k-1}}> \Rightarrow h_{g_1}$;
 4:    if none of the $(k-1)$-subset of $I_{12}$ is in $Q_{k-1}$ then
 5:       Compute $AfnScore(I_{12})$;
 6:       $C \leftarrow C \cup \{I_{12}\}$;
 7:    end if
 8: end for
 9: for all $I \in C$ do
10:    Compute p-values $pvalue(I)$;
11:    if $pvalue(I) < \theta$ then
12:       $Q_k \leftarrow Q_k \cup \{I\}$;
13:    else
14:       $P_k \leftarrow P_k \cup \{I\}$;
15:    end if
16: end for
17: return $Q_k$ and $P_k$
```

For an illustrative example, there are 40 possible multiple-TF regulatory interactions in the regulatory module shown in Figure 3.1. Using the time-series gene expression data in Table 3.1, all the 2-TF regulatory interaction candidates are screened and their p-values are computed (see Table 3.2 row 2-4). Since none of the 2-TF regulatory interaction candidates is significant, a 3-TF interaction $I_4 =< tf_1 \uparrow, tf_2 \uparrow, tf_3 \downarrow > \Rightarrow g \uparrow$ is generated by merging I_2 and I_3. The $AfnScore$ of I_4 is $((10/24)*(10/24))/(12/24)) = 0.347$ and its p-value is 0.04 (see Table 3.2 row 5). Based on I_0 and I_4, we conclude that the target genes g_1 and g_2 are induced by the up-expression of tf_1 and tf_2 and the down-expression of tf_3, and the same target genes are repressed by the down-expression of tf_1.

In terms of time complexity, consider a candidate k-TF regulatory interaction:

$I =< h_{tf_1}, \ldots, h_{tf_k} >\Rightarrow h_g$.

The algorithm computes $AfnScore$ and p-values of all of the subsets, $I - \{tf_j\}$ ($\forall j = 1, 2, ..., k$). If one of them is significant, I is immediately pruned. Hence the time complexity is $O(k)$ for each candidate k-TFs regulatory interaction. Every merging operation requires at most $k-2$ equality comparisons. In the best-case scenario, it produces a viable candidate k-TF interaction. In the worst case, the algorithm merges every pair of infrequent $(k-1)$-TF candidates. Therefore, the overall cost of merging candidates is between $\sum_{k=2}^{|TF|}(k-2)|P_k|$ and $\sum_{k=2}^{|TF|}(k-2)|P_{k-1}|^2$, where P_k is the candidate set of k-TF regulatory interactions. To improve the algorithm efficiency, a hash tree is constructed for the storage and quick access to all of the candidates. Because the maximum depth of the hash tree is k, the cost for populating the hash tree of candidates is $O(\sum_{k=2}^{|TF|} k|P_k|)$. During candidate pruning, it is required to verify whether the $k-1$ subsets of every candidate k-TF regulatory interactions are significant. Since the cost for looking up an item in a hash tree is $O(k)$, the time complex of candidate pruning step is $O(\sum_{k=2}^{w} k(k-2)|P_k|)$.

3.3 Experimental Results

mTRIM was applied on two independently-constructed yeast transcriptional regulatory networks (the Harbison dataset [27] and the Reimand dataset [57]) to identify regulatory interactions. For performance comparison, DREM v3.0 [6] and TRIM [3] were both applied on the same datasets. We did not compare mTRIM with Yeang's method [79] because the latter's objective is to build a reliable TRN instead of predicting regulatory interactions. We evaluated these methods systematically with three independent sources: single TF knockouts [30] for individual regulatory interactions, genetic interactions (GI) [14] for 2-TF regulatory interactions and synthetic data for high-order regulatory interactions.

Using the EM-Based Bayesian inference approach, 658 significant individual regulatory interactions were mined in the Harbison dataset and 164 significant ones were mined in the Reimand dataset (Table 3.4). The results show that while many individual TFs drive target genes' behaviors, it is clear that most of them (4,414 in the Harbison dataset and 1,539 in the Reimand dataset) are "no decision". It indicates that a large proportion of TFs need to work collaboratively with other TFs.

Multiple-TF regulatory interactions were inferred with a new association mining algorithm. In total, 670 regulatory interactions with multiple TFs were discovered (Table 3.5). The results show that at most 6 TFs collaboratively regulate the same target genes. All the TF combinations with more than 6 TFs are either insignificant or have a significant subset. The whole experiments finished in 30 minutes on a high performance computer cluster.

3.3.1 Data preparation

Yeast ChIP-chip binding data [27][1] was downloaded, and a p-value cutoff of 0.001 was applied (the same threshold used in [22]) to obtain the Harbison dataset. It contains 169 TFs, 2,864 target genes and 6,253 TF-DNA bindings. Next we applied the same statistical approach as in [57] to filter the union of the yeast ChIP-chip binding data [42][2] and the binding-site predictions [21, 46] [3] to generate the Reimand dataset with 2,230 TF-DNA binding relationships between 268 TFs and 1,509 target genes. To obtain the regulatory modules in the TRNs, all the target genes were

[1] http://younglab.wi.mit.edu/regulatory_code

[2] http://jura.wi.mit.edu/cgi-bin/young_public/navframe.cgi?s=17&f=downloaddata

[3] http://rulai.cshl.edu/SCPD

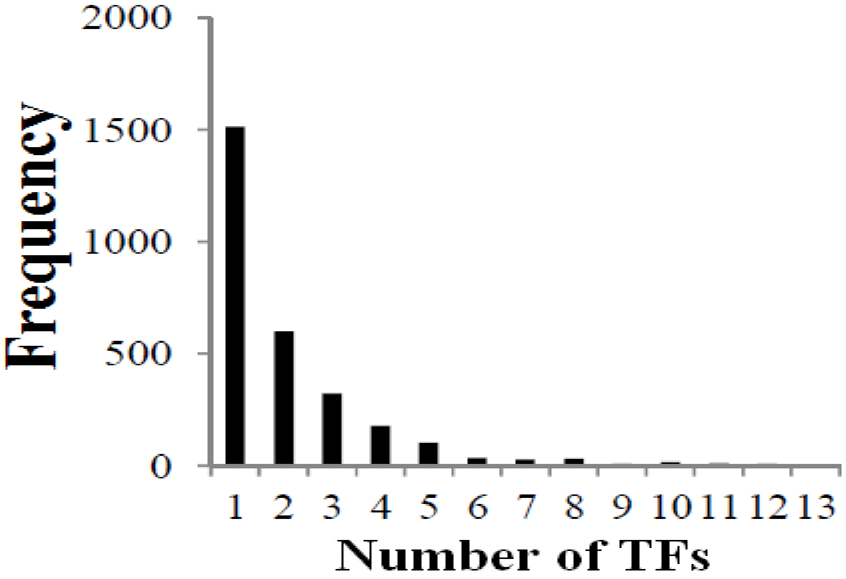


Figure 3.2: In-degree distribution of the yeast TRN for Harbison dataset.

clustered based on their gene expression values with Cluster 3.0 (specifically, k-means), which uses Pearson correlation coefficient for gene similarity metric [20], resulting in 50 clusters. The clusters are then evaluated with Gene Ontology enrichment analysis using Bingo [47], and unenriched clusters are discarded. To construct regulatory modules from the clustering results, the target genes that are regulated by the same TFs were partitioned if they are not in the same cluster. Finally, 2,172 and 1,031 regulatory modules were obtained in the Harbison and Reimand networks respectively. The distribution of genes and regulatory modules (Figure 3.2 reveals that many genes are bound by multiple TFs). Table 3.3 shows the regulatory modules sizes before and after network partitioning for both Harbison and Reimand datasets.

To identify the individual and collaborative regulatory interactions in the above datasets, three widely used time-series microarray datasets (alpha, CDC28 and elu) from yeast cell cycle studies were collected [68] as training data. These datasets contain 49 time points in total. In these experiments, yeast cells were first synchronized to the same cell cycle stage, released from synchronization, and then the total RNA samples were taken at even intervals for a period of time

Table 3.3: Regulatory modules size for Harbison and Reimand datasets

Size	**Before Network Partition**	**After Network Partition**
Harbison Dataset		
1	1512	988
2	599	504
3	323	296
4	176	153
5	102	94
6	34	34
7	26	23
8	31	29
9	10	8
10	17	15
11	11	8
12	9	2
13	3	3
14	4	3
15	0	0
16	3	2
17	1	1
18	1	1
19	2	2
Total	**2864**	**2172**
Reimand Dataset		
1	1076	630
2	275	253
3	87	82
4	39	36
5	17	16
6	7	6
7	6	6
8	1	1
9	0	0
10	1	1
Total	**1509**	**1031**

(Table 2.4). In order to decide whether a gene is significantly up or down regulated, a gene expression change cutoff of 0.35 was applied.

Table 3.4: The number and type of the regulatory interactions for individual TFs predicted by mTRIM.

(a) Habrison Dataset		
	Activator	**Repressor**
Necessary	194	184
Sufficient	118	162
Necessary & Sufficient	29	69
No Decision	4414	
(b) Reimand Dataset		
	Activator	**Repressor**
Necessary	22	43
Sufficient	42	32
Necessary & Sufficient	7	18
No Decision	1543	

To evaluate the individual regulatory relations, single-TF knockout microarray data were collected [30], and a p-value cut-off of 0.05 (as used in [30]) was applied to determine whether a gene is significantly affected by a TF knockout. To evaluate the 2-TF regulatory interactions, we downloaded the SGA genetic interaction dataset [14], which is composed of 1,711 queries crossed to 3,885 array strains. Of the 1,711 queries, 1,377 are deletion mutants of non-essential genes and 334 are essential gene alleles. The SGA dataset contains 762,146 genetic interactions. Two genes are genetically interacted if mutations in both of them produce a phenotype that is significantly different to each mutation's individual effects. In a 2-TF regulatory interaction, if TFs collaboratively regulate the same target genes, the down-regulation of both TFs should have a significantly different phenotype as the down regulation of each individual TF. Therefore, such TF pairs should have a significant p-value in the GI dataset. To evaluate the high-order multiple-TF regulatory interactions, a synthetic binding network were built, which contains 11 TFs, 17 target genes and 58 regulation/binding relationships. The network also contains two feed forward loops. Corresponding time-series gene expression data containing 500 time-points were randomly generated with 10% or 40% noise rate.

Table 3.5: Number of the multiple-TF regulatory interactions identified by mTRIM.

Dataset	2-TF	3-TF	4-TF	5-TF	6-TF
Harbison	350	61	82	43	10
Reimand	95	15	7	7	0

3.3.2 Evaluation 1: Single TF Knock-outs

We used the single TF knockout microarray data to evaluate the performance of mTRIM on individual TF regulatory interaction predictions in terms of the identification of "necessary" TFs (*i.e.,* if the expression values of the target genes are significantly changed when the TF is knocked out). For the Harbison dataset, the prediction precision of mTRIM is 94.44%, higher than the results of TRIM (82.50%). Using the Reimand dataset, mTRIM has a precision of 91.94%, significantly higher than the results of TRIM (61.54%). DREM is not compared since it does not predict "necessary" TFs.

3.3.3 Evaluation 2: Genetic Interaction

In a regulatory module with two TFs, if both TFs collaborate to regulate the same target genes, the down-regulation of both TFs should have significantly different phenotypes from the down-regulation of each individual TF. Therefore, such TF pairs should have a significant p-value in the GI dataset. To this end, for the pairs of TFs that are predicted by mTRIM to work collaboratively, we adopted the GI dataset [14] for evaluation. Figure 3.3 and Figure 3.4 show the Receiver Operating Characteristic curve (ROC) of mTRIM, TRIM and DREM on Harbison dataset and Reimand dataset respectively. For Harbison dataset, the area under curve (AUC) of mTRIM is 0.81, much higher than the AUC of DREM (0.51) and TRIM (0.75). For Reimand dataset, the AUC of mTRIM is 0.80, higher than DREM (0.52) and TRIM (0.64). In addition, to explore whether the perfor-

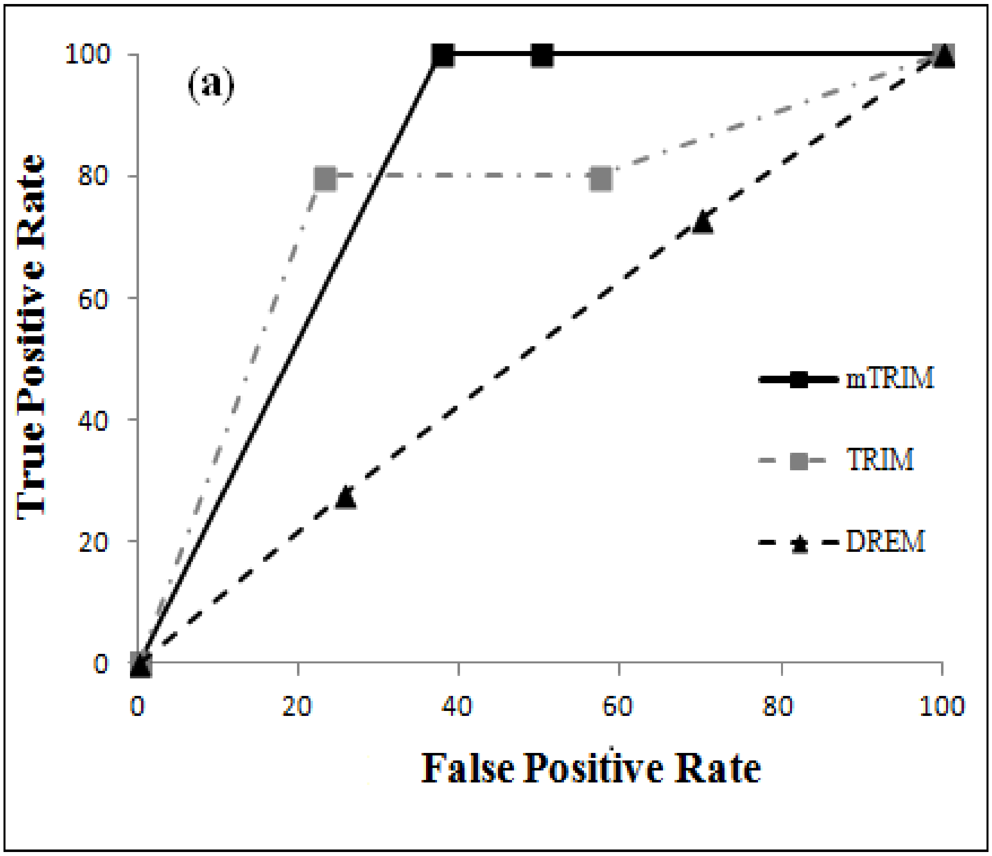


Figure 3.3: Evaluation of the 2-TF regulatory interactions using genetic interactions on Harbison dataset.

mance of mTRIM is sensitive to parameter settings, we altered its parameters systematically. For the Harbison dataset, Figure 3.5 shows the AUC values with different gene expression cutoffs, GI cutoffs, and p-value cutoffs of *AfnScore* respectively. Similarly, for Reimand dataset, Figure 3.5 shows the varying of the AUC values using different thresholds. These show that our method is robust with the GI cutoff and p-value cutoff of *AfnScore*, although its performance gradually decreases with the increase of gene expression cutoffs.

3.3.4 Evaluation 3: Synthetic Transcriptional Regulatory Networks

A synthetic transcriptional regulatory network was generated to evaluate the performance of mTRIM in detecting high-order multiple-TF regulatory interactions (see Figure 3.6). The synthetic network

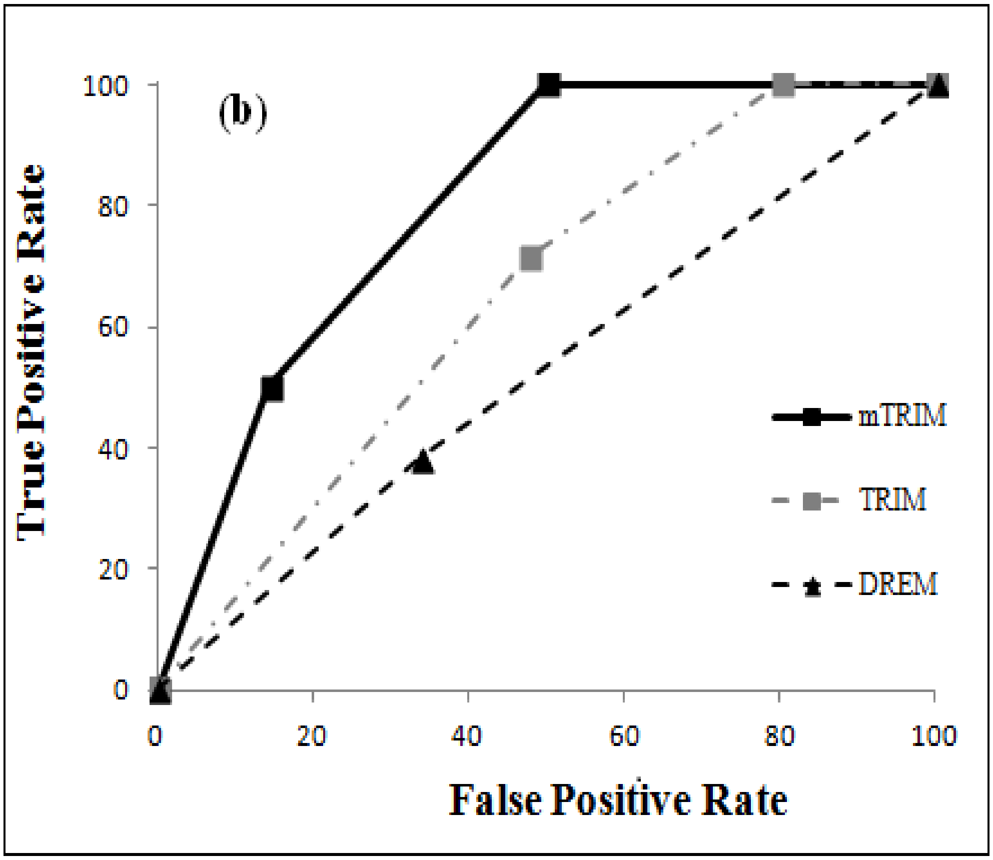


Figure 3.4: Evaluation of the 2-TF regulatory interactions using genetic interactions on Reimand dataset.

has 28 nodes (11 TFs and 17 target genes) and 58 edges, in which the solid line represents a real transcriptional regulation and 12 (20.69%) dotted lines represent TF-DNA bindings but no regulation. The dotted lines were added to the network in order to test the precision of mTRIM. For the synthetic network, two time series gene expression datasets with 500 time-points were generated. In order to test the robustness of mTRIM, we repeated the simulation test twice with different rates of noises added to the simulated gene expression data sets.

A comparison between all the three algorithms (see Figure 3.7, Figure 3.8, and Figure 3.9) indicates that the performance of mTRIM is constantly the best on precision, specificity and sensitivity (equivalent to recall). Precisely, the precision of mTRIM is 87.5%, while the precisions of DREM and TRIM are 62.5% and 66.67% respectively (Figure 3.8). The recall of mTRIM is

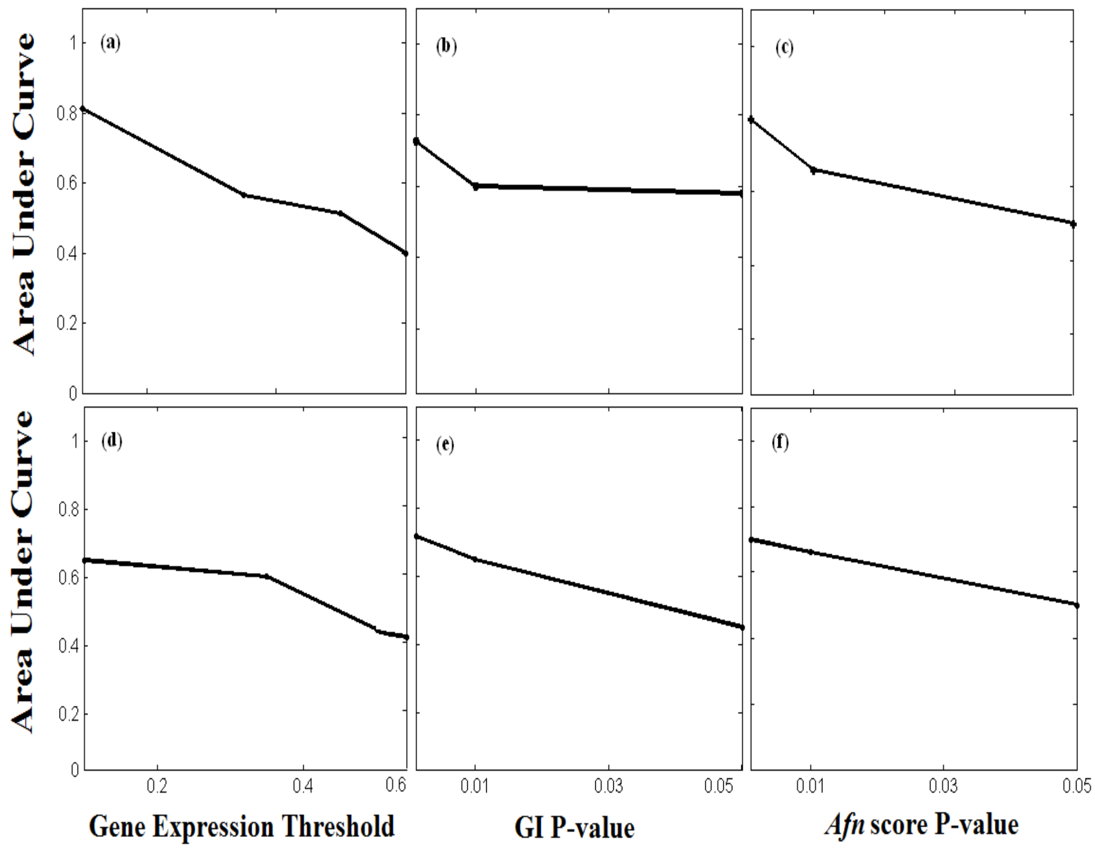


Figure 3.5: Parameter adjustments on the evaluation of the high-order regulatory interactions using genetic interactions. (a,d) Different gene expression cutoffs, (b,e) Different GI significance cutoffs, (c,f) Different mTRIM cutoffs on Harbison dataset and Reimand dataset respectively.

significantly higher than TRIM because it identified 4 out of 5 regulatory interactions with more than two TFs, while TRIM, because of the scalability issue, cannot find any regulatory interactions with more than two TFs (see Figure 3.9). It also shows that mTRIM is less sensitive to the change of the noise rates from 10% to 40% in the gene expression data than the other two algorithms.

3.4 Conclusion

Revealing the mechanisms of the transcriptional regulatory programs in TRNs is essential for understanding the complex control by which genes are expressed in living cells. The inference of collaborative protein-DNA functions helps paving the critical path for new drug development. In

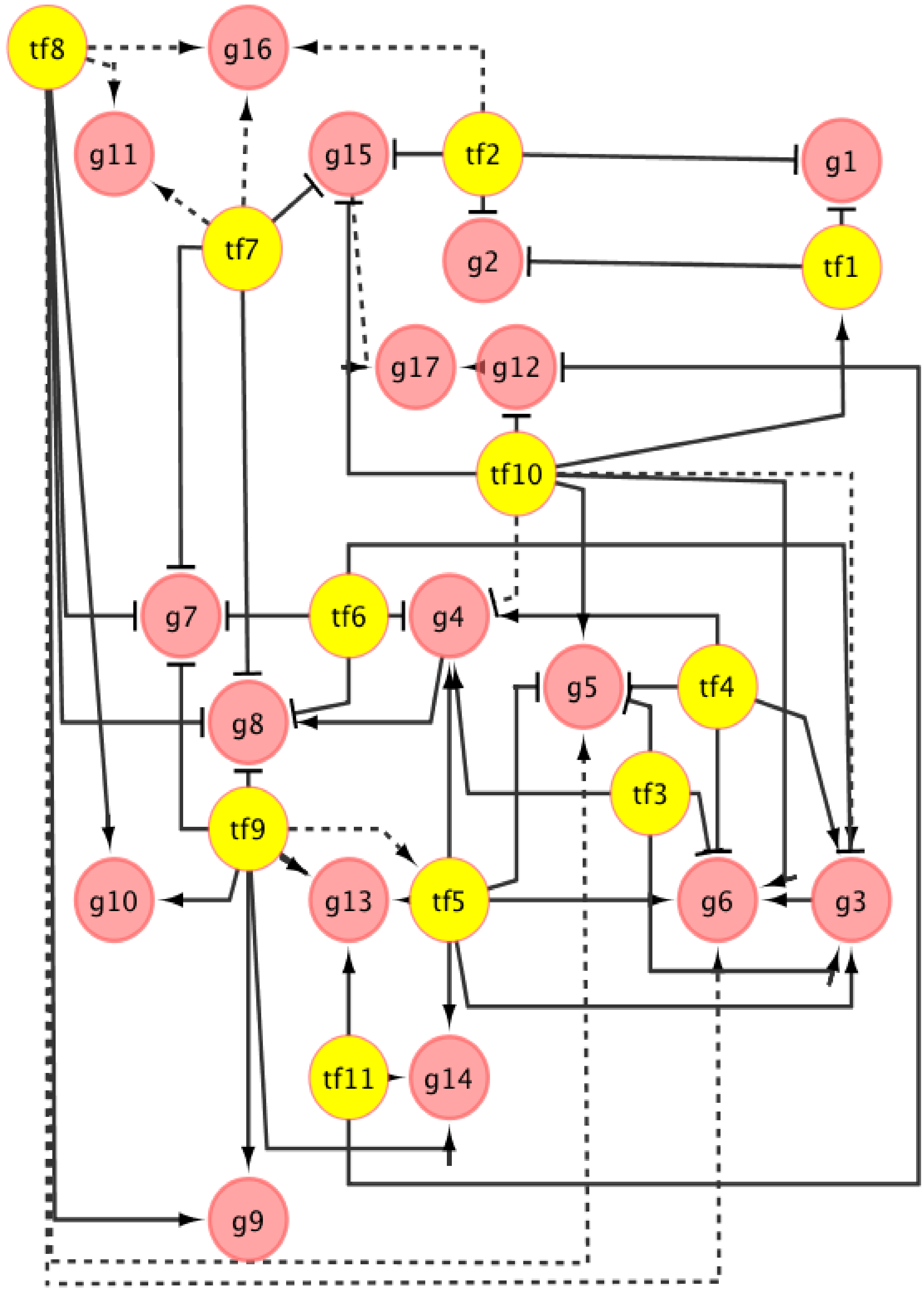


Figure 3.6: A synthetic transcriptional regulatory network, in which the solid lines represent transcriptional regulations and the dotted lines represent TF-DNA bindings only (meaning binding but not regulation).

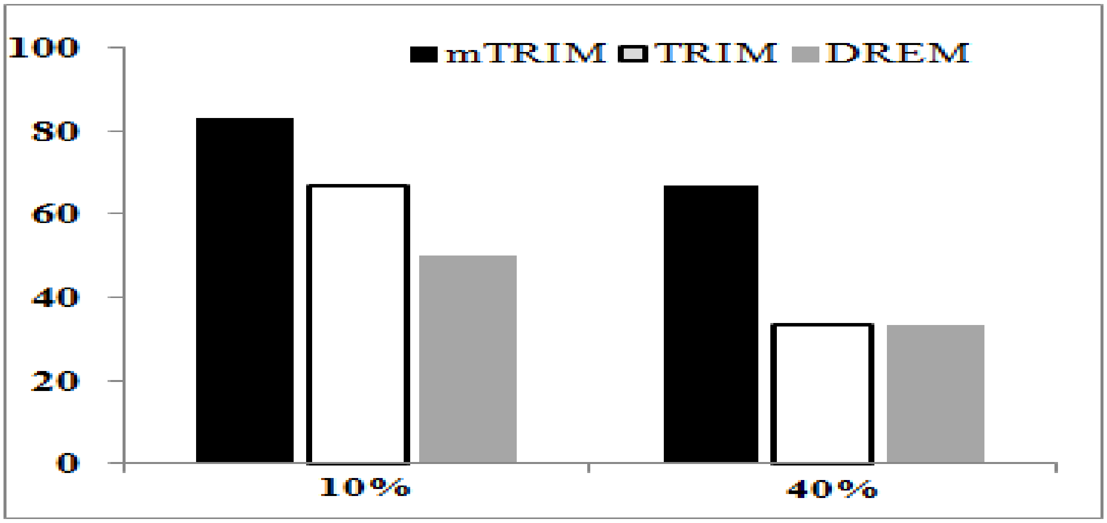


Figure 3.7: Comparing mTRIM with DREM and TRIM on two sets of synthetic data with different noise rates: Specificity

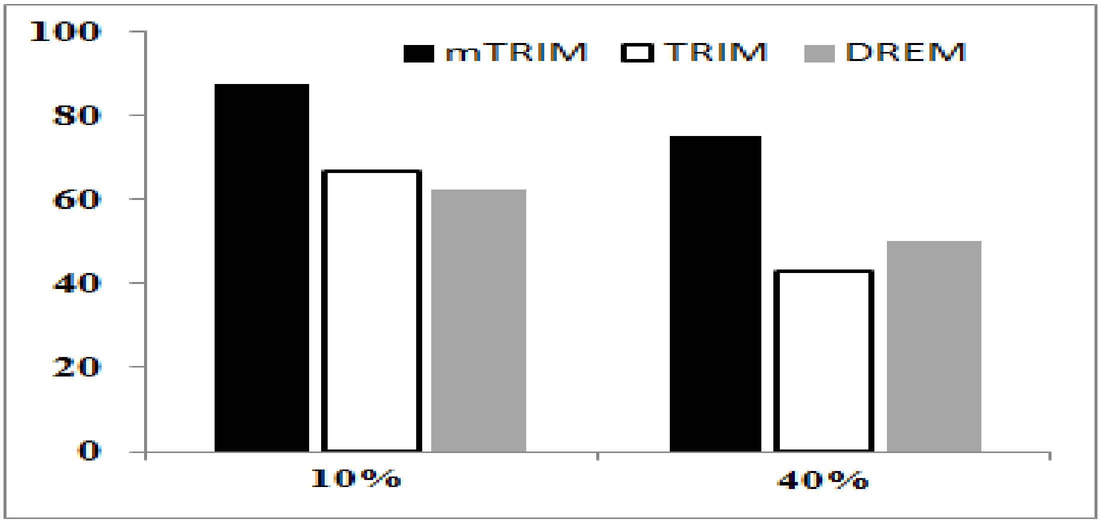


Figure 3.8: Comparing mTRIM with DREM and TRIM on two sets of synthetic data with different noise rates: Precision

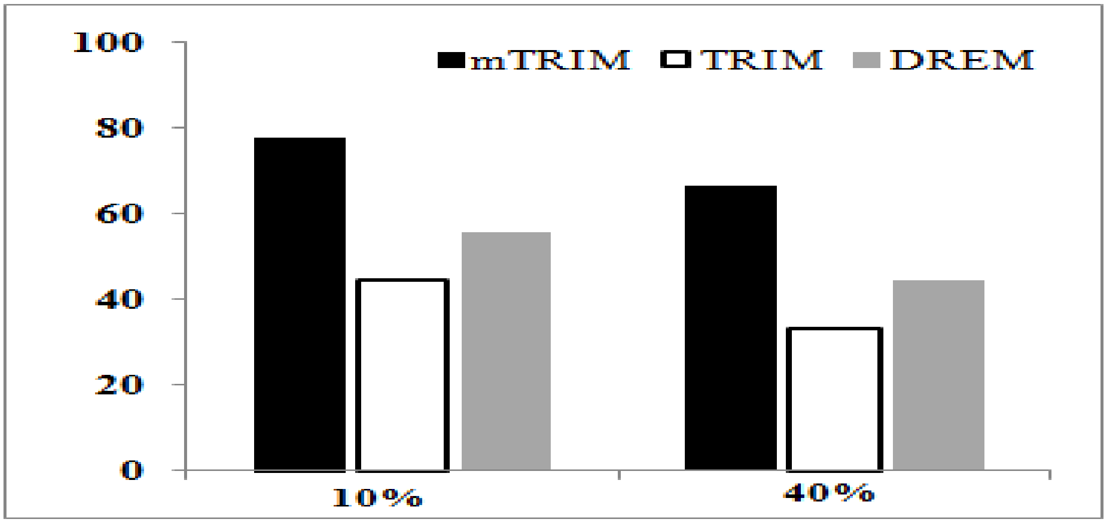


Figure 3.9: Comparing mTRIM with DREM and TRIM on two sets of synthetic data with different noise rates: Sensitivity.

this chapter, we identify the *regulatory interactions* between TFs and target genes with mTRIM, an integration of an EM-based Bayesian inference and a new association rule mining approach built on a set of basic constraints that relate gene expression patterns to regulatory interactions. mTRIM is not limited by the number of TFs. The experimental results show that mTRIM is clearly better than the existing algorithms. We compared in this chapter mTRIM, TRIM and DREM on three independent datasets (*i.e.,* single TF-knockouts, genetic interactions, and synthetic data). Clearly mTRIM has higher scalability than TRIM, since the latter can only handle up to 2-TF regulatory modules, missing 31.3% regulatory modules in Harbison dataset and missing 14.4% regulatory modules in Reimand dataset. Since it is difficult to obtain the ground truth for algorithm performance evaluation, we generated two sets of synthetic data and used them to validate the experimental results.

In the next chapter, we would like to extend this work by including extra data. For example, since miRNA can degrade the genes induced by certain TFs [36], we will consider miRNA-target bindings aiming to enhance the performance and understand how different regulators collaborate to regulate target gene expressions.

Chapter 4

Human Cancer Transcription Regulation

4.1 Introduction

As explained in previous chapters, understanding the regulation of gene expression is critical to decipher the control of biological processes in cellular organisms. At the transcriptional level, the main regulators are the transcription factors, that can regulate the global gene expressions behavior by activating or repressing their target genes. However, recent studies have revealed another sort of regulation, that is microRNA (miRNA) regulation at post-transcriptional level [65, 72, 84]. This regulation takes place via mRNA cleavage or translational repression by binding to the 3 untranslated region of target mRNA. In general, a single miRNA can concurrently down-regulate hundreds of target mRNAs [65, 72, 84]. Hence, understanding the regulatory mechanisms of the two main regulators (TFs and miRNAs) are necessary to decipher the gene regulation mechanism. It is necessary to understand the regulatory relationships between TFs, miRNAs and target genes. However, the combined regulations of miRNAs and TFs are complicated due to several reasons. The regulation mechanism does not only include the interactions between each regulator and the target genes, but also involves the interactions between the regulators themselves, and one TF might cancel the effect of another. In addition, the miRNA might block the effect of the TFs on their targets. Besides, miRNAs often have overlapping targets and act combinatorially which creates a complex regulatory networks [60].

Although biological experiments can help discover these complicated relationships, but the process is extremely costly and time consuming. Hence, computational approaches may help understanding such complex relationships. Reverse engineering has been widely used in inferring TF-controlled transcriptional regulation networks [6, 55, 63]. However, successful applications of reverse engineering in inferring combinatorial networks involving TFs and miRNAs were rarely seen except when small-scaled combinatory networks of miRNAs and TFs were mapped around some selected genes [80]. The major obstacle is the lack of simultaneously measured miRNA expression data and mRNA expression data. These parallel miRNA expression and miRNA expression expression datasets are continuously released to public [12] but for a set of samples for example, various tumor samples [45, 80].

The co-regulation of TFs and miRNAs has been studied previously in [65, 84]. The authors in [65, 84] provided a method to find out the shared downstream targets between TFs and miRNAs. The method then used a statistical tests to measure the significance of the shared targets between the regulators, and to prune out the insignificant co-regulating interactions.

A rule based approach is proposed in [74], that discover the regulatory interactions that include miRNAs, TFs, and their targets based on the predicted target bindings. Le Bechec et al. in [41] utilized target prediction databases to construct a regulatory network that involves miRNAs, TFs, and mRNAs.

Recently, Huang et al. in [31] developed a web tool (mirConnX) for constructing the regulatory networks that include miRNA, TFs, and mRNAs. Their method used both predicted targets and expression data to build the network. However, an edge in this network represents association, which may not necessarily indicate regulatory interaction. Zacher et al. in [81] proposed a Bayesian inference approach that used expression data to infer the activity of miRNAs and TFs individually, however, this approach does not consider the interactions between miRNAs and TFs.

Le et al. in [72] used Bayesian network learning to learn the network structure to construct a regulatory network from multiple sources of data: gene expression profiles of miRNAs, TFs and target genes. Then, the learned network is further learned to identify the interactions between miRNAs and TFs. The method also applies a network motif finding algorithm to further infer the network. Z. Liang et al. in [44] proposed mirAct, a web tool that uses the negative regulatory relationship between miRNAs and their target genes to infer the miRNA activity based on gene-expression data. mirAct evaluates the activity change of a miRNA by determining the miRNA activity in a sample, then it analyze the collective behavior of miRNA activity in different classes of samples.

In this chapter, we extend the definition of a regulatory module R, to include a set of regulators L that regulates a set of target genes via an interaction type I. The regulators includes a set of TFs and a set of miRNAs. The interaction type I represents the way the regulators L collaborates with the set of their target genes. We describe TmiRNA (Transcription Regulations and miRNA), a new model to infer the interactions types between different regulators L and their set of target genes.

4.2 Methods

A regulatory network can be represented as a directed graph in which each node is a TF, a miRNA, or gene, and each edge pointing from a TF or a miRNA to a gene represents a regulation relationship between them. A regulatory interactions is a self-consistent regulatory unit $R(L,G,I)$ representing a set of co-expressed genes $G = \{g_1, g_2, \ldots, g_n\}$ regulated in concert by a group of regulators L which includes a set of TFs and miRNAs $L = \{tf_1, tf_2, \ldots, tf_m, mir_1, mir_2, \ldots, mir_n\}$ that govern the target genes' behaviors via regulatory interaction I [72].

4.2.1 Concepts

A regulatory interaction sequence S is a non-empty set of regulators L. A k-sequence is a set of k regulators and their target gene, the target gene is considered a dummy item in the regulatory sequence and does not contribute to the value of k. In addition, any sequence that includes only miRNA regulators and a target gene, is also excluded, as miRNA-targets regulatory relations are not the focus of our work. Specifically, a regulatory interaction sequence at tissues t $S_t =< h_{tf_1}, \dots, h_{tf_i}, \dots, h_{tf_m}, a_{mir_1}, \dots, a_{mir_n} >\Rightarrow h_g$ is defined as a set of TFs $\{tf_1, \dots, tf_m\}$ and a set of miRNAs $\{mir_1, \dots, mir_n\}$ co-regulating a set of genes $\{g_1, \dots, g_n\}$, where h_{tf_i} is the behavior of TF i in tissues t; a_{mir_i} is a miRNA that is bound to g and functioning in tissue t. S_t is expressed as $S_t = h_{tf_i}$, if there is no miRNAs bound to g, or the miRNAs bound to g but not functioning in tissue t. h_g is the behavior of the target genes in t; $h_x \in \{\uparrow, \downarrow, -\}$, meaning up-express, down-express and no change respectively.

For example, given three tissues, t_1, t_2, and t_3. tf_1, tf_2, mir_1 are bound to gene g_1. mir_1 functions in t_2, t_3 only. tf_1 is up-expressed in t_1, and t_2 and down-expressed in t_3, and tf_2 is up-expressed in all the tissues. Then we have three regulatory interaction patterns, $P_{t_1} =< tf_1, \uparrow, tf_2, \uparrow, g_1 >$, $P_{t_2} =< tf_1, \uparrow, tf_2, \uparrow, mir_1, g_1 >$, and $P_{t_3} =< tf_1, \downarrow, tf_2, \uparrow, mir_1, g_1 >$, and the three regulatory sequences have g_1 as a dummy item, i.e, $S_{t_1} = \{P_{t_1}, g_1\}$,$S_{t_2} = \{P_{t_2}, g_1\}$, $S_{t_3} = \{P_{t_3}, g_1\}$.

4.2.2 Inferring Regulatory Sequences

In order to identify multiple-TF and miRNAs regulatory interactions, a modified sequential pattern mining approach has been developed. A sequential pattern mining approach is defined as follows: Given a set of regulatory sequences S_i, and given a user-specified threshold, sequential pattern mining is to find all frequent sub-regulatory sequences, i.e., the sub-regulatory sequence whose occurrence frequency in the set of tissues is no less than a given threshold. Instead of using the concepts of support that is commonly used in a conventional frequent mining application [1, 69], we define a scoring function (called *Rscore*) according to the gene expression agreement between the regulators groups and their target genes.

$$Rscore(s) = 2 * \oplus(s) + \ominus(s) \tag{4.1}$$

where $\oplus(s)$ and $\ominus(s)$ are calculated using in Equation 4.2 and Equation 4.3 respectively.

$$\oplus(s) = F_{s_+} * \log \frac{F_{s_+}}{e_{(+)}} \tag{4.2}$$

$$\ominus(s) = F_{s_-} * \log \frac{F_{s_-}}{e_{(-)}} \tag{4.3}$$

where e_+ and e_- are computed using Equation 4.4 and Equation 4.5 respectively and F_{s_+} and F_{s_-} are the frequency of s in the positive set and the negative set respectively. If the target gene direction in s is up, the positive set is the set of all sequences that up-regulates their targets, and the negative set is the set of all sequences that down-regulate or does not change their targets. If the target gene direction in s is down, the positive set is the set of all sequences that down-regulates their targets, and the negative set is the set of all sequences that up-regulate or does not change their targets.

$$e(+) = N_s * \log \frac{F_+}{B} \tag{4.4}$$

$$e(-) = N_s * \log \frac{F_-}{B} \tag{4.5}$$

where $F(+)$ and $F(-)$ are the number of positive set and negative set respectively. B is the background, the number of up-regulated, down-regulated and no-change genes in every tissue.

The key steps of our approach is candidate generations and is explained in details in the following sub-section.

4.2.2.1 Candidate Generations

In TmiRNA, each sequence contains only one element composed of a set of items. An item could be a TF or a miRNA. A frequent *k*-sequence is a sequence with a p-value of the *Rscore* less than the user-specified threshold. Let P_k denote the set of all frequent *k*-sequences,and C_k the set of candidate *k*-sequences. Given a sequence *S* and a sub-sequence *c*, *c* is a *c*ontiguous subsequence of *s* if any of the following conditions hold:

1. c is derived from s by dropping an item (not the target gene);

2. c is a contiguous subsequence of c', and c' is a contiguous sub-sequence of *s*.

Canidate generation is generated in two steps:

1. **Join Phase:** We generate candidate sequences by joining C_{k-1} with C_{k-1}. A sequence s_1 is joined with s_2 if the subsequence obtained by dropping the last item of s_1 is obtained by dropping the last item of s_2, and the dummy item (target gene) of s_1 is equal to the dummy item of s_2. The candidate sequence generated by joining s_1 and s_2 is extended with the last item in s_2.

2. **Prune Phase:** Compute the *Rscore* for every k-sequence using Equation 4.1. A k-sequence s is significant if the p-value of its *Rscore* within all k-sequences distribution is less than a user-given threshold. For any significant sequence s that has a $k-1$ contiguous subsequence in P, delete s, otherwise add s to P. For any insignificant sequence, it is added to candidate set C.

The algorithm starts with all C_1-sequences. A c_i in C_1 possibly includes each TF_i and its target gene. The *Rscore* of each $c_i \in C_1$ sequence is computed using equation 4.1. Then, the prune phase is applied. Specifically, $\forall c_i \in C_1$, if c_i is significant, it is added to P and is output to the user. Otherwise, c_i is added to C. Then, the join phase is applied to join C_{k-1} with C_{k-1} sequences. Any significant sequence c that does not have a $k-1$ contiguous subsequence in P is reported to the user and added to P, otherwise, it is deleted.

For counting the candidates, we followed the same approach used in [69]. Given a set of candidates sequences C and a data-sequence S, the goal is to find all sequences in C that are found in S. A hash-tree data structure is used to reduce the number of candidates in C.

4.2.3 Constructing Regulatory Modules

Starting with the binding network as an initial regulatory network, each gene constructs a separate regulatory network with its regulators (TFs and miRNAs). Then TmiRNA is used to find significant regulatory sequences. A set of genes that have the same directions and regulated by the same set of regulators are grouped into the same regulatory network. Each significant regulatory interaction that drive the same set of target genes G to the same direction is considered a replacement for the target genes set G.

4.3 Experimental Results

We applied TmiRNA on human NCI-60 cancer cell lines to infer regulatory interactions. We downloaded series GSE5846 Gene Expression Omnibus. The series represents NCI-60 cancer cell lines, which were profiled with their genome-wide expression patterns using Affymetrix HG-U133A chips. The Total RNA sample of each of the NCI-60 cell lines was obtained before the treatment of any anticancer compound. It contains 60 samples. In total, we got 12385 gene ensembles.

We used miRNA-targets binding dataset from a collection of seven algorithm/databases i.e., TargetScan[43] (release 5.2), PITA [37], Starbase [78], HitSensor[83], miR2Disease[35], miRecords [77]and TarBASE [64]. The sequences of mature miRNAs were collected from the miRBase (release 19) and 3' UTR sequences of mRNAs were collected from the UCSC Genome Browser (http://genome.ucsc.edu). We used only the binding relation that has been predicted by at least two prediction methods. As a result, we have 564934 relations. We also used binding data obtained from [80]. The data contains 4289 relations including 2850 TF to gene relations, 1439 miRNA to gene relations. To combine the data-set, and save balance between both sets, we used only genes that have at least one TF connection. As a result, we have 77684 miRNAs to gene relations and 2850 TF to gene relations. See Figure 4.2 for TFs distributions in human TRN. We can see that 71.75% of genes are regulated by individual TFs, 21.79% of genes are regulated by two TFs, and 6.45% of the genes are regulated by three or four TFs. We used SAM in [75] to identify genes with statistically significant changes in expression in specific tissues. As the genes expressions fits Normal distribution (See Figure 4.3 for fitting the gene expression levels distribution to Normal distribution).

We used miTEA in [70] to determine whether a given miRNA functions in a specific tissue. miTEA [70] is a framework for miRNA target enrichment analysis. miTEA uses a ranked list of

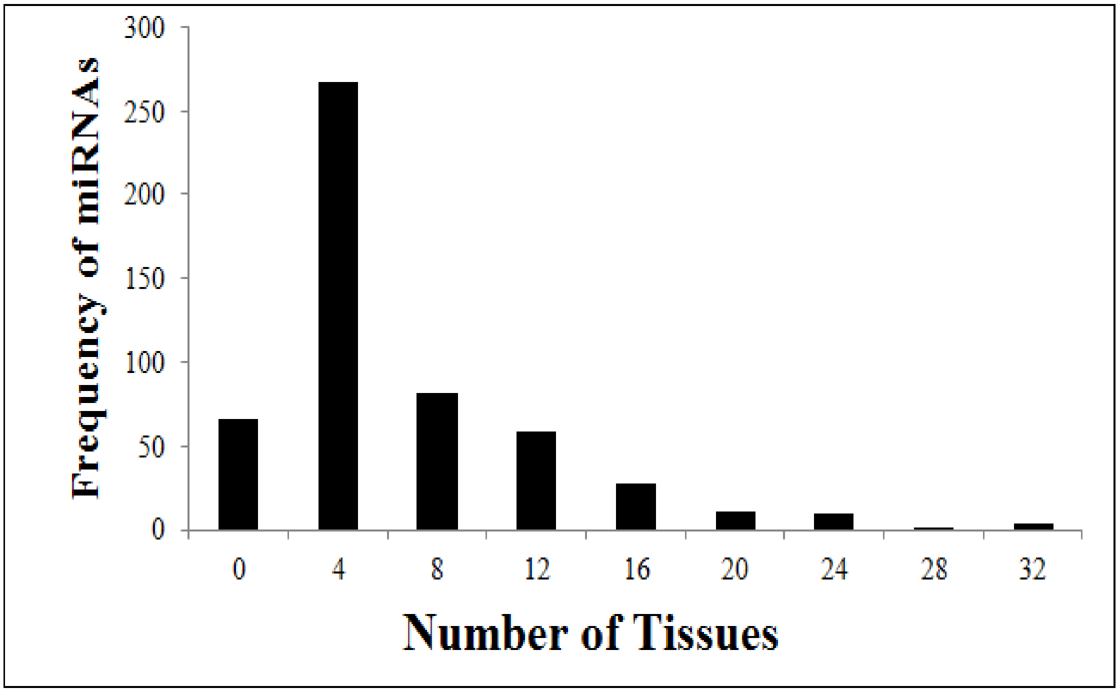


Figure 4.1: A distribution of miRNAs that are active in different tissues.

genes to find miRNAs of which their targets are enriched in the top of the the ranked list. We provided miTEA with a list of ranked genes based on their down-regulation, as identified by SAM, miTEA reports the list of miRNAs that functions in tissue t. See Figure 4.1 for the distributions of miRNAs that function in different tissues. We can see that 50.95% of miRNAs are active in four tissues, 12.59% of miRNAs are not active in any of the given tissues, and 36.54 % of miRNAs are active in 8 or more tissues.

Since there is no ground truth for evaluating our model. We used gene Ontology functional enrichment and synthetic data to give an insight on the performance of our model.

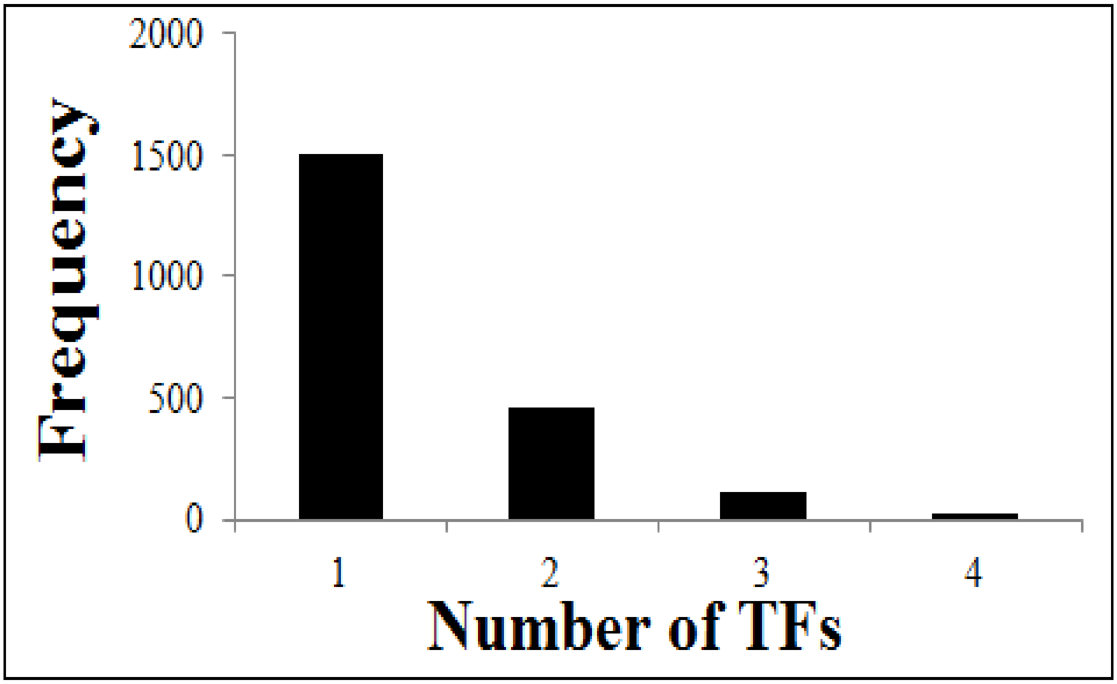


Figure 4.2: In-degree distribution of the Human TRN

4.3.1 Evaluating TmiRNA Regulatory Modules using Gene Ontology Functional Enrichment

To evaluate the coherence of the regulatory modules constructed by TmiRNA, a functional analysis consisted on Fishers exact (two-tailed) test implemented in DAVID v6.7 [1] [66] was used to identify the functional categories enriched among all target genes associated (FDR-adjusted P-value 0.05). Categories that at least 75% of their genes are significant at FDR-adjusted P-value 0.05 are considered enriched. We compare the biological meaningfulness of the regulatory modules constructed by TmiRNA to those constructed by network partitioning as describe in Chapter two and three. Specifically, in network partitioning, we cluster all the target genes in a TRN based on their gene expression values with Cluster 3.0 (specifically, k-means), which uses Pearson correlation coefficient for gene similarity metric [20]. Genes in the same cluster are considered to be co-expressed.

[1] `http://david.abccncifcrf.gov/`

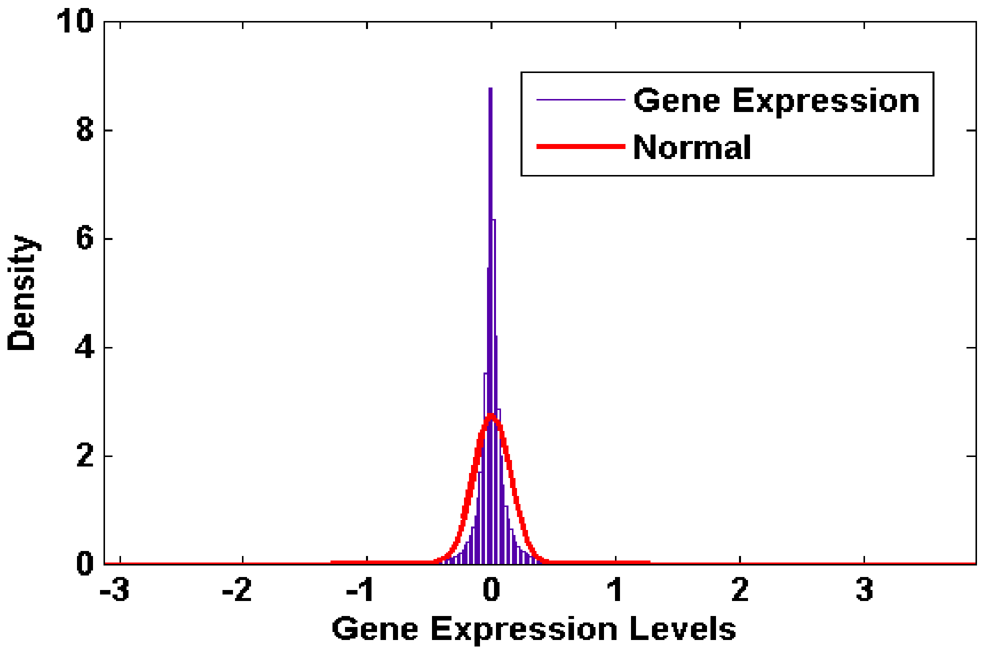


Figure 4.3: Fitting gene expression Levels to normal distribution

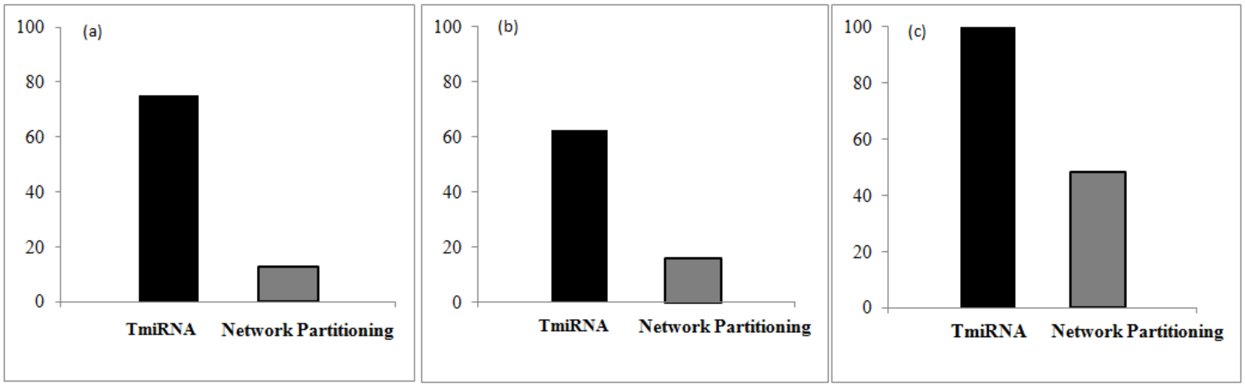


Figure 4.4: Functional enrichment evaluation of TmiRNA under different categories (a) Biological Process, (b) Molecular Functions, and (c) Cellular Components.

And the co-expressed and co-regulated genes are usually weighed to be regulated by the same TF(s) with a similar interaction model. For the target genes that are regulated by the same set TFs and miRNAs, we partition them based on whether they are in the same cluster to construct their regulatory modules.

Figure 4.4 shows a comparison between functional enrichment of the regulatory networks generated by TmiRNA, and regulatory networks generated by network partitioning, under 3 categories, (a) biological process, (b) molecular functions, and (c) cellular components. Under biological process category, 75% of regulatory modules of TmiRNA are enriched, higher than network portioning regulatory modules (12.90%). Under the molecular functions category, 62.50% of TmiRNAs regulatory modules are enriched, significantly higher than network partitioning regulatory modules (16.12%). Under cellular components category, 100% the regulatory modules formed by TmiRNA are enriched while only 48.38% of the network partitioning modules are enriched. Clearly, TmiRNA better clusters genes into biologically meaningful groups.

4.3.2 Evaluating TmiRNA using Synthetic miRNA-TF Regulatory Networks

A synthetic transcriptional regulatory network was generated to evaluate the performance of TmiRNA detecting high-order multiple-TF regulatory interactions (see Figure 4.5). The synthetic network has 61 nodes (21 TFs, 14 miRNAs, and 26 target genes) and 104 edges, in which the solid line represents a real transcriptional regulation dotted lines represent TF-DNA bindings but no regulation. For the synthetic network, two time series gene expression data-sets with 500 time-points were generated with a different (10% and 40%) noise rate. The networks contains 6 loops.

The synthetic network simulates the case where a set of k-TFs collaboratively regulates a target gene, we design the gene expressions such that, if $(h_g \neq 0) \Rightarrow abs(\sum_{i=1}^{k} h_{tf_i} = k)$ and if $(h_{tf_i} = 0$ for $i \in \{1,2,\ldots,k\}) \Rightarrow h_g = 0$. A comparison between all the TmiRNA and mTRIM algorithms (see Figure 4.6 and Figure 4.7) indicates that the performance of TmiRNA is slightly better in terms of precision and recall. This can be explained by the synthetic network design that considers the miRNA-target binding as regulation as we cannot use miTEA on synthetic miRNAs to infer active miRNAs in various tissues.

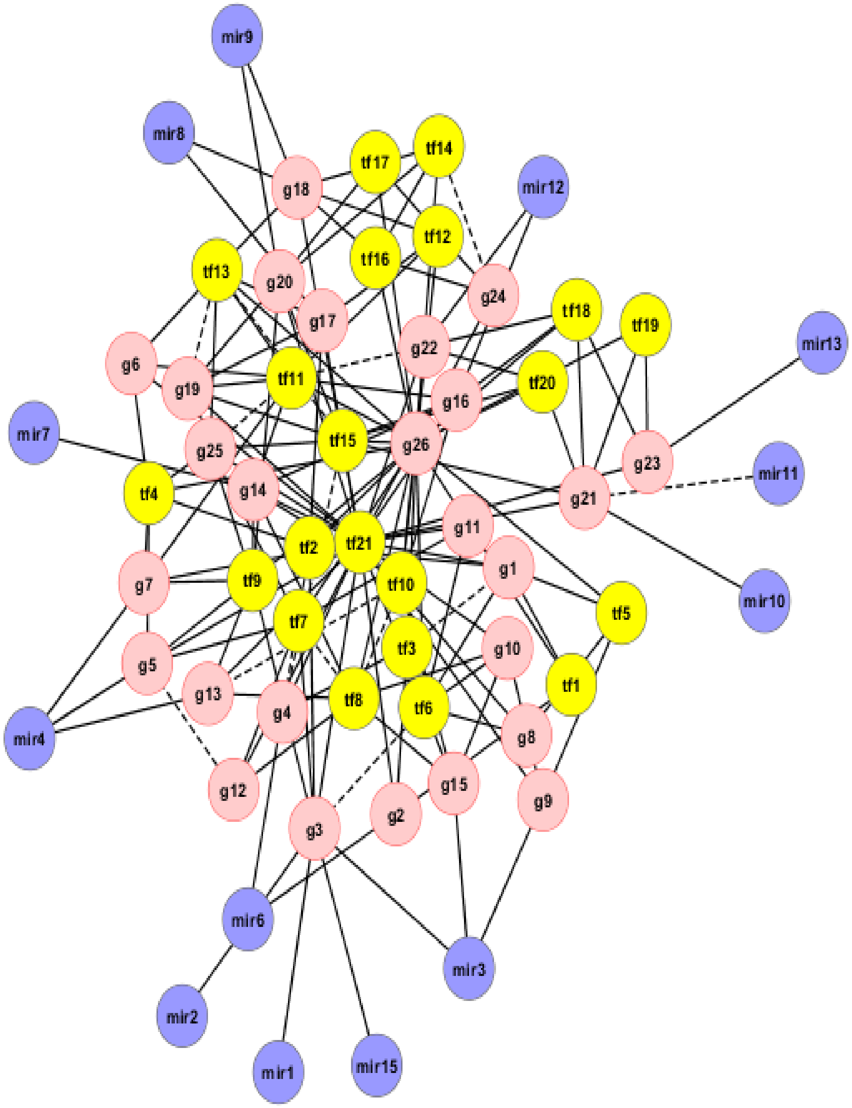


Figure 4.5: Synthetic network used for evaluating TmiRNA.

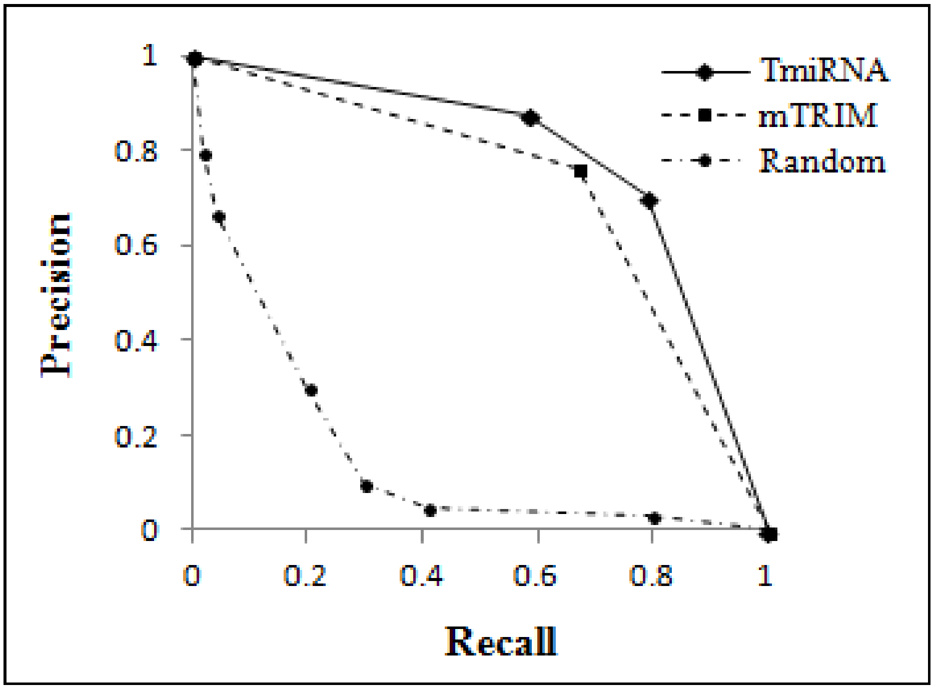


Figure 4.6: Performance of TmiRNA on synthetic data with 10% noise rate.

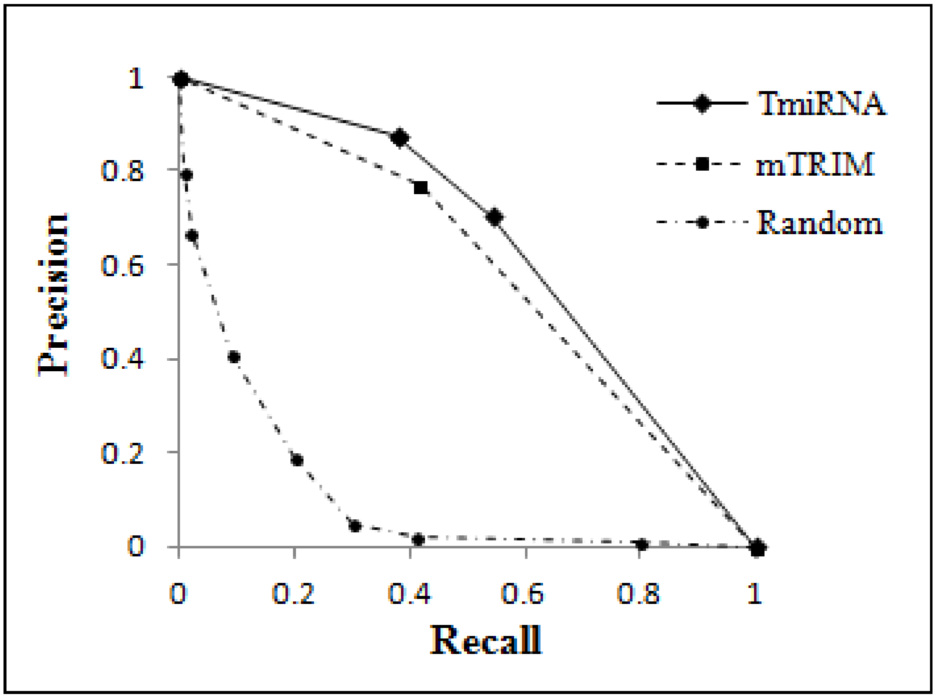


Figure 4.7: Performance of TmiRNA on synthetic data with 40% noise rate.

4.4 Conclusion

Considerable efforts have been considered to explore the transcriptional regulatory networks such that the TFs are the main regulator. Other studies investigated the post-transcriptional regulatory networks where miRNAs only are the main regulators. In the previous Chapter, we proposed mTRIM, an approach to infer collaborations of TFs in transcriptional regulatory networks. However, mTRIM ignores the post-transcriptional regulation of miRNAs. In this work, we proposed TmiRNA, a new approach to infer the regulatory interactions between multiple regulators, that involves both TFs and miRNAs. TimRNA shows better results than mTRIM. In addition, gene Ontology functional enrichment shows that the regulatory modules inferred by TmiRNA have better coherence.

Chapter 5

Conclusions and Future Work

In this chapter we conclude our work and outline directions for future work.

5.1 Conclusions

The overall goal of the approaches presented in this dissertation, is to infer the regulatory interactions between multiple TFs and the set of their target genes.

In chapter one, we provided a brief background about transcriptional regulations. We described in short both the experimental and computational approaches used to study transcription regulation. We provided a short introductory to the computational models used in the rest of the dissertation.

In chapter two, we defined the regulatory interaction as a set of one or two TFs' interactions with their set of target genes. We described TRIM, an HMM based approach that infer the regulatory interactions of at most 2-TFs and the set of their target genes. TRIM showed promising results on two organisms, yeast and Arapidopsis. Thus TRIM is a useful tool to help predict the phenotype of TF double-knockouts to reduce the number of double knock-out or over-expression experiments needed. However, it is common that a set of TFs (one or more) collaboratively regulate a set of target genes. In order to infer the interactions between multiple TFs and their set of target genes, we need to a new model that can address the scalability issue of combining multiple TFs.

In chapter three, we designed a new model mTRIM, a Bayesian inference and an association rules based approach that infer the regulatory interactions between multiple TFs and their set of target genes. mTRIM captures the least number of TFs that can derive their set of target genes to a specific direction. Unlike TRIM, mTRIM has no limitations on the number of TFs it handles. However, mTRIM does not take the effect of miRNAs on their targets into considerations.

In chapter four, we extended the definition of regulatory interactions to include not only the TFs but the miRNAs as well. We presented TmiRNA, a sequential pattern mining approach that infers the regulatory interactions of multiple TFs, miRNAs, and their set of target genes. TmiRNA is based on an affinity score that takes the negative hits of the regulators into account, which helps capturing the interactions that are significantly different from the negative set. However, TmiRNA doesn't utilize miRNA expressions which could potentially enhance the performance of the inference.

Although the approaches detailed in this dissertation shows successful results and better performance when compared with competitor models, however these approached don't address TF-miRNA regulation, the cases when a TF regulates a miRNA. These approaches also don't address miRNA-TF regulation, the cases when a miRNA down-regulates a TF.

5.2 Future Work

Although the approaches detailed in this dissertation infer multiple regulators interactions with their set of targets, there is still plenty of room for improvements. The intended future work falls into the following directions:

- The approaches detailed in this dissertation infer collaborations between transcription factors. However, the proposed approaches can't infer the competence between TFs. TFs compete with each others. This competition is achieved via the binding of the TFs to overlapping binding sites. The winner of this competition depends on some factors that include the TF concentrations, the number of TFs, and the quantitative nature of the protein–protein and protein–DNA interactions. We need to design new models that can take these factors into considerations.

- In the presented approaches, we used static binding of transcription factors to their targets. As a result, the temporal order of binding of transcriptional factors to their target has not been examined. In our future work, we plan to integrate and analyze dynamic binding information of transcription factors to their targets which helps revealing the hierarchical circuit in which a combinatorial set of transcription factors target distinct sets of genes at different times.

- We also plan to study metabolic behaviours that can be controlled by gene networks. Microorganisms engage in continuous adaptations of their physiology as they are exposed to changing environments. Gene networks that control metabolism should restore metabolic functions upon environmental changes. Although metabolic networks are better understood than their corresponding gene networks, the identity of the metabolites that regulate the activity of transcription factors of metabolic genes and the kinetics of reactions in the gene network are harder to be determined. As a result, it is not yet understood which metabolic behaviours can be controlled by gene networks as well as the functional limits of gene networks. Whether gene networks can optimise metabolic functions is an open question we need to address in our future work.

- Finally, we need to use indirect biological evidences including multiple TF knockouts, metabolic pathways, protein-protein interactions, etc., for biological validation.

BIBLIOGRAPHY

BIBLIOGRAPHY

[1] R. Agrawal and R. Srikant. Fast algorithms for mining association rules. Proc. of 20th Intl. Conf. on VLDB, pages 487–499, 1994.

[2] S. Awad and J. Chen. Inferring transcription factor collaborations in gene regulatory networks. BMC Systems Biology, In Press, 2013.

[3] S. Awad, N. Panchy, S. Ng, and J. Chen. Inferring the regulatory interaction types of transcription factors in transcriptional regulatory networks. Journal of Bioinformatics and Computational Biology, 10(5):1250012, 2012.

[4] O. Babur, E. Demir, M. Gnen, C. Sander, and U. Dogrusoz. Discovering modulators of gene expression. Nucleic Acids Research, 38:5648–5656, 2010.

[5] S. Balaji, M. Babu, M. Iyer, M. Luscombe, and L. Aravind. Comprehensive analysis of combinatorial regulation using the transcriptional regulatory network of yeast. Molecular Biology, 360:213–227, 2001.

[6] Z. Bar-Joseph, G. Gerber, T. Lee, N. Rinaldi, J. Yoo, F. Robert, B. Gordon, E. Fraenkel, T. Jaakkola, R. Young, and K. Gifford. Computational discovery of gene modules and regulatory networks. Nature Biotechnology, 21:1337–1342, 2003.

[7] Z. Bar-Joseph, A. Gitter, and I. Simon. Studying and modelling dynamic biological processes using time-series gene expression data. Nature Genetics Reviews, 13, 2012.

[8] R. Bonneau, D. Reiss, P. Shannon, M. Facciotti, L. Hood, N. Baliga, and V. Thorsson. The inferelator: an algorithm for learning parsimonious regulatory networks from systems-biology data sets de novo. Genome Biology, 7:R36, 2006.

[9] M. Cascante, L. Boros, B. Comin-Anduix, P. De Atauri, J. Centelles, and P. Le. Metabolic control analysis in drug discovery and disease. Nature Biotechnology, 20:243 – 249, 2002.

[10] J. Chen, Z. Hu, M. Phatak, J. Reichard, J. Freudenberg, S. Sivaganesan, and M. Medvedovic. Genome-wide signatures of transcription factor activity: Connecting transcription factors, disease, and small molecules. PLoS Comput Biol, 9, 2013.

[11] G. Chena, P. Sullivanb, and M. Kosorok. Biclustering with heterogeneous variance. PNAS, 10:12253–12258, 2013.

[12] S. Cho, Y. Jun, S. Lee, H. Choi, S. Jung, Y. Jang, C. Park, S. Kim, S. Lee, and W. Kim. mirgator v2.0: an integrated system for functional investigation of micrornas. Nucleic Acids Research, 39:D158D162, 2011.

[13] W. Cookson, L. Liang, G. Abecasis, M. Moffatt, and M. Lathrop. Mapping complex disease traits with global gene expression. Nature Genetics Reviews, 10:184–194, 2009.

[14] M. Costanzo, A. Baryshnikova, J. Bellay, et al. The genetic landscape of a cell. Science, 327:425–431, 2010.

[15] B. Deplancke, D. Dupuy, M. Vidal, and J. Walhout. A gateway-compatible yeast one-hybrid system. Genome Research, 14(10b):2093–2101, 2004.

[16] B. Deplancke, A. Mukhopadhyay, W. Ao, M. Elewa, A. Grove, J. Martinez, R. Sequerra, L. Doucette-Stamm, S. Reece-Hoyes, A. Hope, A. Tissenbaum, E. Mango, and M. Walhout. A gene-centered c.elegans protein-dna interaction network. Cell, 125:1193–1205, 2006.

[17] D. Deutscher, I. Meilijson, S. Schuster, and E Ruppin. Can single knockouts accurately single out gene function? BMC Systems Biology, 2, 2008.

[18] R. Duda, P. Hart, and D. Stork. Pattern Classification. 2nd edition, 2001.

[19] R. Durbin, S. Eddy, A. Krogh, and G. Mitchison. Biological sequence analysis: probabilistic models of proteins nucleic acids.

[20] M. Eisen, P. Spellman, P. Brown, and D. Botstein. Cluster analysis and display of genome-wide expression patterns. PNAS, 95:14863–14868, 1998.

[21] I. Erb and E. Nimwegen. Statistical features of yeast's transcriptional regulatory code. Proceeding of International Conference on Computational and System Biology, 1:111–118, 2006.

[22] J. Ernst, O. Vainas, C. Harbison, I. Simon, and Z. Bar-Joseph. Reconstructing dynamic regulatory maps. Molecular Systems Biology, 3(74):1–13, 2007.

[23] N. Friedman. Probabilistic models for identifying regulation networks. Bioinformatics, 19:II57, 2003.

[24] B. Futcher. Transcriptional regulatory networks and the yeast cell cycle. Current Opinion in Cell Biology, 14:676–683, 2002.

[25] S. Gygi, Y. Rochon, R. Franza, and R. Aebersold. Correlation between protein and mrna abundance in yeast. Molecular and Cellular biology, 19:1720–1730, 1999.

[26] D. Hagen, G. McCaffrey, and G. Sprague. Pheromone response elements are necessary and sufficient for basal and pheromone-induced transcription of the fus1 gene of saccharomyces cerevisiae. Molecular and Cellular biology, 11(6):2952–2961, 1991.

[27] K. Harbison, B. Gordon, T. Lee, N. Rinaldi, K. Macisaac, T. Danford, N. Hannett, J. Tagne, D. Reynolds, J. Yoo, E. Jennings, J. Zeitlinger, D. Pokholok, M. Kellis, A. Rolfe, K. Takusagawa, E. Lander, D. Gifford, E. Fraenkel, and R. Young. Transcriptional regulatory code of a eukaryotic genome. Nature, 431:99–104, 2004.

[28] A. Honkela, C. Girardot, E. Gustafson, Y. Liu, E. Furlongb, N. Lawrence, and M. Rattray. Model-based method for transcription factor target identification with limited data. PNAS, 107:7793–7798, 2010.

[29] S. Hoth, M. Morgante, J. Sanchez, M. Hanafey, S. Tingey, and N. Chua. Genome-wide gene expression profiling in arabidopsis thaliana reveals new targets of abscisic acid and largely impaired gene regulation in the abi1-1 mutant. Journal of Cell Science, 115:4891–4900, 2006.

[30] Killion P. Hu, Z. and V. Iyer. Genetic reconstruction of a functional transcriptional regulatory network. Nature Genetics, 39:683–687, 2007.

[31] G. Huang, C. Athanassiou, and P. Benos. mirconnx: condition-specific mrna-microrna network integrator. Nucleic Acids Research, 39:W416W423, 2011.

[32] T. Ideker and V. Thorsson. Discovery of regulatory interactions through perturbation: Inference and experimental design. Pacific Symposium of Biocomputing, 5, 2000.

[33] J. Ihmels, G. Friedlander, S. Bergmann, O. Sarig, Y. Ziv, and N. Barkai. Revealing modular organization in the yeast transcriptional network. Nature Genetics, 31:370–377, 2002.

[34] R. Jansen. Studying complex biological systems using multifactorial perturbation. Nature Genetics, 4:145–151, 2003.

[35] Q. Jiang, Y. Wang, Y. Hao, L. Juan, M. Teng, X. Zhang, M. Li, G. Wang, and Y. Liu. miR2Disease: a manually curated database for microRNA deregulation in human disease. Nucleic acids research, 37(suppl 1):D98–D104, 2009.

[36] J. Joung, K. Hwang, J. Nam, S. Kim, and B. Zhang. Discovery of microrna-mrna modules via population-based probabilistic learning. Bioinformatics, 23:1141–1147, 2007.

[37] M. Kertesz, N. Iovino, U. Unnerstall, U. Gaul, and E. Segal. The role of site accessibility in microRNA target recognition. Nature Genetics, 39(10):1278–1284, October 2007.

[38] J. Kilian, D. Whitehead, J. Horak, D. Wanke, S. Weinl, O. Batistic, C. D'Angelo, E. Bornberg-Bauer, J. Kudla, and K. Harter. The atgenexpress global stress expression data set: protocols, evaluation and model data analysis of uv-b light, drought and cold stress responses. Plant Journal, 50:347–363, 2007.

[39] J. Kreps, Y. Wu, H. Chang, T. Zhu, X. Wang, and J. Harper. Transcriptome changes for arabidopsis in response to salt, osmotic, and cold stress. Plant Physiol, 130:2129–2141, 2002.

[40] H. Le and B. Ziv. Inferring transcription factor collaborations in gene regulatory networks. Bioinformatics, 29:i89i97, 2013.

[41] A. Le Bchec, E. Portales-Casamar, G. Vetter, M. Moes, P. Zindy, A. Saumet, D. Arenillas, C. Theillet, W. Wasserman, C. Lecellier, and E. Friederich. Mir@nt@n: a framework integrating transcription factors, micrornas and their targets to identify sub-network motifs in a meta-regulation network model. BMC Bioinformatics, 12, 2011.

[42] T. Lee, N. Rinaldi, F. Robert, D. Odom, Z. Bar-Joseph, G. Gerber, N. Hannett, C. Harbison, C. Thompson, I. Simon, J. Zeitlinger, E. Jennings, H. Murray, B. Gordon, B. Ren, J. Wyrick, J. Tagne, T. Volkert, E. Fraenkel, D. Gifford, and R. Young. Transcriptional regulatory networks in saccharomyces cerevisiae. Science, 298:799–804, 2002.

[43] B. Lewis, C. Burge, and D. Bartel. Conserved seed pairing, often flanked by adenosines, indicates that thousands of human genes are microRNA targets. Cell, 120:15–20, 2005.

[44] Z. Liang, H. Zhou, Z. He, H. Zheng, and J. Wu. miract: a web tool for evaluating microrna activity based on gene expression data. Nucleic Acids Research, 39:W139–W144, 2011.

[45] J. Lu, G. Getz, E. Miska, E. Alvarez-Saavedra, J. Lamb, D. Peck, A. Sweet-Cordero, B. Ebert, R. Mak, A. Ferrando, J. Downing, T. Jacks, R. Horvitz, and T. Golub. Microrna expression profiles classify human cancers. Nature, 435:834–838, 2005.

[46] K. MacIsaac, T. Wang, B. Gordon, D. Gifford, G. Stormo, and E. Fraenkel. An improved map of conserved regulatory sites for saccharomyces cerevisiae. BMC Bioinformatics, 7:113, 2006.

[47] S. Maere, K. Heymans, and M. Kuiper. Bingo: a cytoscape plugin to assess overrepresentation of gene ontology categories in biological networks. Bioninformatics, 21:3448–3449, 2005.

[48] M. Maienschein-Cline, J. Zhou, K. White, R. Sciammas, and A. Dinner. Discovering transcription factor regulatory targets using gene expression and binding data. Bioinformatics, 28:206–213, 2011.

[49] F. Markowetz. How to understand the cell by breaking it: Network analysis of gene perturbation screens. PLoS Comput Biol, 6, 2010.

[50] C. Miller, B. Schwalb, K. Maier, D. Schulz, S. Dumcke, B. Zacher, A. Mayer, J. Sydow, L. Marcinowski, L. Dolken, D. Martin, A. Tresch, and P. Cramer. Dynamic transcriptome analysis measures rates of mrna synthesis and decay in yeast. Molecular Systems Biology, (458):1–13, 2010.

[51] S. Mukherjee and S. Mitra. Hidden markov models, grammars, and biology: a tutorial. Journal of Bioinformatics and Computational Biology, 3:491–526, 2005.

[52] I. Ong, J. Glasner, and D. Page. Modelling regulatory pathways in e. coli from time-series expression profiles. Journal of Bioinformatics, 18:S241–S248, 2002.

[53] S. Palaniswamy, S. James, H. Sun, R. Lamb, R. Davuluri, and E. Grotewold. Agris and atregnet: A platform to link cis-regulatory elements and transcription factors into regulatory networks. Plant Physiology, 140:818–829, 2006.

[54] P. Park. Chipseq: advantages and challenges of a maturing technology. Nature Reviews Genetics, 10(10):669–680, 2009.

[55] J. Qian, J. Lin, N. Luscombe, H. Yu, and M. Gerstein. Prediction of regulatory networks: genome-wide identification of transcription factor targets from gene expression data. Bioinformatics, 19:1917–1926, 2003.

[56] P. Qiu. Recent advances in computational promoter analysis in understanding the transcriptional regulatory network. Biochemical. and Biophysical Research Communications, 309:495–501, 2003.

[57] J. Reimand, J. Vaquerizas, A. Todd, J. Vilo, and N. Luscombe. Comprehensive reanalysis of transcription factor knockout expression data in saccharomyces cerevisiae reveals many new targets. Nucleic Acids Research, 38:4768–4777, 2010.

[58] B. Ren, F. Robert, J. Wyrick, O. Aparicio, E. Jennings, I. Simon, J. Zeitlinger, J. Schreiber, N. Hannett, E. Kanin, T. Volkert, C. Wilson, S. Bell, and R. Young. Genome-wide location and function of dna binding proteins. Science, 290:2306–2309, 2000.

[59] W. Schmitt, R. Raab, and G. Stephanopoulos. Elucidation of gene interaction networks through time-lagged correlation analysis of transcriptional data. Genome Res, 14(8):1654–1663, 2004.

[60] M. Schulza, K. Panditb, C. Cardenasb, N. Ambalavananc, N. Kaminskib, and Z. Bar-Joseph. Reconstructing dynamic microrna-regulated interaction networks. PNAS, 2013.

[61] E. Segal, M. Shapira, A. Regev, D. Pe'er, D. Botstein, D. Koller, and N. Friedman. Module networks: identifying regulatory modules and their condition-specific regulators from gene expression data. Nature Genetics, 34:166–167, 2003.

[62] E. Segal, B. Taskar, A. Gasch, N. Friedman, and D. Koller. Rich probabilistic models for gene expression. Bioinformatics, 17:S243–52, 2001.

[63] E. Segal, R. Yelensky, and D. Koller. Genome-wide discovery of transcriptional modules from dna sequence and gene expression. Bioinformatics, 19:i273–i282, 2003.

[64] P. Sethupathy, B. Corda, and A. Hatzigeorgiou. TarBase: A comprehensive database of experimentally supported animal microRNA targets. RNA, 12(2):192–197, 2006.

[65] R. Shalgi, D. Lieber, M. Oren, and Y. Pilpel. Global and local architecture of the mammalian micrornatranscription factor regulatory network. PLoS Comput Biol, 3:e131, 2007.

[66] B. Sherman, D. Huang, Q. Tan, Y. Guo, S. Bour, D. Liu, R. Stephens, M. Baseler, C. Lane, and R. Lempicki. A gene-centered database integrating heterogeneous gene annotation resources to facilitate high-throughput gene functional analysis. BMC Bioinformatics, 8, 2007.

[67] I. Shmulevich, E. Dougherty, S. Kim, and W. Zhang. Probabilistic boolean networks: a rule-based uncertainty model for gene regulatory networks. Bioinformatics, 18:261–274, 2002.

[68] P. Spellman, G. Sherlock, M. Zhang, V. Iyer, K. Anders, M. Eisen, P. Brown, D. Botstein, and B. Futcher. Comprehensive identification of cell cycle-regulated genes of the yeast saccharomyces cerevisiae by microarray hybridization. Molecular Biology of the Cell, 9:3273–3297, 1998.

[69] R. Srikant and R. Agrawal. Mining sequential patterns: Generalizations and performance improvements. EDBT 96 Proceedings of the 5th International Conference on Extending Database Technology: Advances in Database Technology, 1057:1–17, 1996.

[70] I. Steinfeld, R. Navon, R. Ach, and Z. Yakhini. mirna target enrichment analysis reveals directly active mirnas in health and disease. Nucleic Acids Research, 41:e45, 2013.

[71] T. Thorne and M. Stumpf. Inference of temporally varying bayesian networks. Bioinformatics, 24:3298–3305, 2012.

[72] Le. Thuc, L. Liu, B. Liu, A. Tsykin, G. Goodall, K. Satouand, and J. Li. Inferring microrna and transcription factor regulatory networks in heterogeneous. BMC Bioinformatics, 14, 2013.

[73] A. Tong and C. Boone. Synthetic genetic array analysis in saccharomyces cerevisiae. Methods in Molecular Biology, 313:171–191, 2006.

[74] D. Tran, K. Satou, T. Ho, and T. Pham. Computational discovery of mir-tf regulatory modules in human genome. Bioinformation, 8:371377, 2010.

[75] V. Tusher, R. Tibshirani, and G. Chu. Significance analysis of microarrays applied to the ionizing radiation response. PNAS, 98:51165121, 2001.

[76] A. Valouev, D. Johnson, A. Sundquist, C. Medina, E. Anton, S. Batzoglou, R. Myers, and A. Sidow. Genome-wide analysis of transcription factor binding sites based on chip-seq data. Nature Methods, 5:829–834, 2008.

[77] F. Xiao, Z. Zuo, G. Cai, S. Kang, X. Gao, and T. Li. miRecords: an integrated resource for microRNA–target interactions. Nucleic acids research, 37(suppl 1):D105–D110, 2009.

[78] J. Yang, J. Li, P. Shao, H. Zhou, Y. Chen, and L. Qu. starBase: a database for exploring microRNA-RNA interaction maps from Argonaute CLIP-Seq and Degradome-Seq data. Nucleic Acids Research, 39(suppl 1):D202–D209, 2011.

[79] H. Yeang and T. Jaakkola. Modeling the combinatorial functions of multiple transcription factors. Journal of Computational Biology, 13:463–480, 2006.

[80] H. Yu, K. Tu, Y. Wang, J. Mao, L. Xie, Y. Li, and Y. Li. Combinatorial network of transcriptional regulation and microrna regulation in human cancer. BMC Systems Biology, 6, 2012.

[81] B. Zacher, K. Abnaof, S. Gade, E. Younesi, A. Treschand, and H. Frhlich. Joint bayesian inference of condition specific mirna and transcription factor activities from combined gene and microrna expression data. Bioinformatics, 28:17141720, 2012.

[82] Q. Zhang and M. andersen. Dose response relationship in anti-stress gene regulatory networks. PLoS Comput Biol, 3:e24, 2007.

[83] Y. Zheng and W. Zhang. Animal microRNA target prediction using diverse sequence-specific determinants. Journal of Bioinformatics and Computational Biology, 8(4):763–788, 2010.

[84] Y. Zhou, J. Ferguson, J. Chang, and Y. Kluger. Inter and intra combinatorial regulation by transcription factors and micrornas. BMC Genomics, 8, 2007.

[85] C. Zou and J. Feng. Granger causality vs. dynamic bayesian network inference: a comparative study. BMC Bioinformatics, 10, 2009.