2017

DEVELOPMENT OF BIOINFORMATICS TOOLS AND ALGORITHMS FOR IDENTIFYING PATHWAY REGULATORS, INFERRING GENE REGULATORY RELATIONSHIPS AND VISUALIZING GENE EXPRESSION DATA

Chathura J. Gunasekara
Michigan Technological University, cjgunase@mtu.edu

DEVELOPMENT OF BIOINFORMATICS TOOLS AND ALGORITHMS FOR IDENTIFYING PATHWAY REGULATORS, INFERRING GENE REGULATORY RELATIONSHIPS AND VISUALIZING GENE EXPRESSION DATA

By

Chathura J. Gunasekara

A DISSERTATION

Submitted in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

In Computational Science and Engineering

MICHIGAN TECHNOLOGICAL UNIVERSITY

2017

This dissertation has been approved in partial fulfillment of the requirements for the Degree of DOCTOR OF PHILOSOPHY in Computational Science and Engineering.

School of Forest Resources and Environmental Science

Dissertation Advisor: *Dr. Hairong Wei*

Committee Member: *Dr. Qiuying Sha*

Committee Member: *Dr. Victor Busov*

Committee Member: *Dr. Laura Brown*

School Dean: *Dr. Terry Sharik.*

Table of Contents

List of figures

List of tables

Preface

The research conducted for this dissertation is presented in four chapters. Chapter 2 presents, "An Efficient Algorithm for Identifying Pathway Regulators through Evaluation of Triple-Gene Mutual Interaction (TGMI)." The author conceptualized this algorithm, with the help of author's advisor, Dr. Hairong Wei. Dr. Wei proposed to evaluate different triple gene interaction measures using information theory, provided data sets and contributed to interpreting the results using existing biological knowledge base. The author developed a computer program in R language, applied the program to multiple datasets, and compared the efficiency of the TGMI algorithm with those of several existing methods.

Chapter 3 presents a web-based application, titled "TF-mining Pipelines for Identifying Regulatory Genes Controlling a Biological Pathway, Process, or Complex Trait from High-Throughput Gene Expression Data." We submitted a manuscript to 'BMC Genomics' journal in 2016 with the same title and received reviews and comments to improve functionalities of the web application. We are planning to resubmit the manuscript to the same journal with novel features. Dr. Wei proposed the idea for this web application, and he was a corresponding author of the two original algorithms; TF-Cluster and TF-Finder, which were published in 'BMC Systems Biology' and 'BMC Bioinformatics' journals in 2011 and 2010 respectively. Dr. Sapna Kumari, a previous member of Dr. Wei's laboratory, published a manuscript under the title of "Evaluation of Gene Association methods for Co-Expression Network Construction and Biological Knowledge Discovery" in 'PLoS ONE' journal in 2012. Xiaohui Ji, a visiting student in Dr. Wei's Laboratory, published two decomposition algorithms, SSGA and MSGA, in 'Scientific Reports' journal in 2017. The author combined the work of these multiple researchers as well as additional functionalities into both web-based data analysis pipelines. Jialin Lei and Avinash aided with the implementation of a data management portal for the web application.

Chapter 4, "ExactSearch: A fast plant motif search tool" was published in BMC Plant Methods journal in 2016. The author developed the software with the aid from Dr. Bin Li, Avinash Subramanian, and Ram Kumar Avari. The author's advisor Dr. Hairong Wei proposed the original idea.

Chapter 5 describes the implementation details of a web-based visualization tool, called STTM JBrowse. The author developed the tool utilizing JBrowse open source platform and produced visualizations shown in this dissertation. Dr. Guiliang Tang and Dr. Hairong Wei proposed the original idea for this web-based system.

Acknowledgements

First, I would like to thank those who helped and motivated me to keep working toward the goal of my doctoral studies and made this dissertation possible. I owe my heartfelt appreciation to my advisor, Dr. Hairong Wei, who opened the door for a career in bioinformatics and computational biology and spent tremendous time guiding me to learn and understand the field of this exciting interdisciplinary area. He spent endless hours illuminating my thoughts and proofreading my manuscripts and dissertation. I am earnestly thankful to my committee members, Dr. Qiuying Sha, Dr. Laura Brown and Dr. Victor Busov for their support and willingness to serve on my committee. I would like to thank Dr. Jennifer Sanders for revising and proofreading this dissertation.

I am indebted to all my teachers for providing me the knowledge to build a foundation for my research and future career. I would also like to acknowledge the financial and technical support provided by the School of Forest Resources and Environmental Science of the Michigan Technological University during my entire doctoral studies. I also grateful to the Graduate School of the Michigan Technological University for providing the finishing fellowship to complete my dissertation during my final semester.

I would like to thank my parents for their patience, encouragement, and without their support, I would not be where I am today. Last, but not least, I thank my wife, without her continuous assistance this dissertation would not be a reality.

Abstract

In the era of genetics and genomics, the advent of big data is transforming the field of biology into a data-intensive discipline. Novel computational algorithms and software tools are in demand to address the data analysis challenges in this growing field. This dissertation comprises the development of a novel algorithm, web-based data analysis tools, and a data visualization platform. Triple Gene Mutual Interaction (TGMI) algorithm, presented in Chapter 2 is an innovative approach to identify key regulatory transcription factors (TFs) that govern a particular biological pathway or a process through interaction among three genes in a triple gene block, which consists of a pair of pathway genes and a TF. The identification of key TFs controlling a biological pathway or a process allows biologists to understand the complex regulatory mechanisms in living organisms. TF-Miner, presented in Chapter 3, is a high-throughput gene expression data analysis web application that was developed by integrating two highly efficient algorithms; TF-cluster and TF-Finder. TF-Cluster can be used to obtain collaborative TFs that coordinately control a biological pathway or a process using genome-wide expression data. On the other hand, TF-Finder can identify regulatory TFs involved in or associated with a specific biological pathway or a process using Adaptive Sparse Canonical Correlation Analysis (ASCCA). Chapter 4 presents ExactSearch; a suffix tree based motif search algorithm, implemented in a web-based tool. This tool can identify the locations of a set of motif sequences in a set of target promoter sequences. ExactSearch also provides the functionality to search for a set of motif sequences in flanking regions from 50 plant genomes, which we have incorporated into the web tool. Chapter 5 presents STTM JBrowse; a web-based RNA-Seq data visualization system built using the JBrowse open source platform. STTM JBrowse is a unified repository to share/produce visualizations created from large RNA-Seq datasets generated from a variety of model and crop plants in which miRNAs were destroyed using Short Tandem Target Mimic (STTM) Technology.

Chapter 1
Introduction

In the post-genomics era, high-throughput technologies generate terabytes of sequencing and expression datasets, which demand highly efficient computational tools for discovering novel biological knowledge. This dissertation includes several computational approaches, which address theoretical and practical challenges in identifying transcription factors (TFs) which regulate biological pathways or processes, reverse engineering context specific gene regulatory networks and visualizing large-scale RNA-Seq gene expression data. The algorithms and the data analysis pipelines developed for this dissertation are now available as efficient and user-friendly web-based software tools, each relating to some aspect of transcriptional regulation, a very important process during which transcription factors (TFs) bind to promoter regions of the target genes and interact with basal transcriptional machinery, including RNA polymerase. These algorithms and the software tools will be instrumental for the elucidation of complex gene regulatory networks, which is a central component of modern biological research.

Analyzing large-scale gene expression datasets to identify key TFs that control a biological pathway or a process is a challenging research problem. In Chapter 2, we present a novel algorithm, TGMI to address this challenge; the TGMI algorithm uses two types of input files: the gene expression profiles of genes in a biological pathway or a process of interest and the gene expression profiles of all TFs or a set of differentially expressed TFs. The algorithm produces a ranked list of regulatory TFs as well as a putative regulatory network that controls the biological pathway or the process. Regulatory TFs are identified by evaluating all combinations of triple gene blocks based on mutual interaction among the three genes; each block consists of two pathway genes and a TF. The advantage of evaluating a triple gene block is that causal patterns can be detected in a tri-variate setting rather than in a bivariate context (Schäfer & Strimmer, 2005). Also, the importance of evaluating a triple gene block was evident in several of our previous publications (Kumari et al., 2016; Lin et al., 2013; Lu et al., 2013; H. Wei, Yordanov, Kumari, Georgieva, & Busov, 2013; H. Wei, Yordanov, Georgieva, Li, & Busov, 2013). The TFs were ranked by the frequencies of interactions in significant triple gene blocks; these frequencies reflected the importance of regulatory TFs in governing a given pathway. Regulatory networks were constructed using the significantly interacting triple gene blocks. Additionally, we developed an algorithm to identify combinatorial TFs, which have significant interactions with pathway genes. Finally, the accuracy of the new algorithm was compared with those of three other existing algorithms.

Chapter 3 presents TF-Miner, a web-based data analysis application that can analyze large-scale gene expression datasets for biological knowledge discovery. TF-Miner is comprised of two data analysis pipelines: TF-Cluster and TF-Finder. TF-Cluster is a data analysis pipeline that includes a collaborative network construction phase and a network decomposition phase, which can be used for building collaborative clusters of TFs. In this web-based implementation, the collaborative network construction phase was supplemented with four additional pair-wise association methods to facilitate the identification of a range of linear and non-linear associations, thereby increasing the accuracy of identifying collaborative clusters of TF (Kumari et al., 2012). The decomposition phase of the original TF-Cluster algorithm utilized only Triple-link Algorithm (Nie et al., 2011); for the web-based TF-Cluster pipeline, two additional algorithms, Single-Seed Growing Algorithm (SSGA) and Multi-Seed Growing Algorithm (MSGA) (Ji et al., 2017) were incorporated. This inclusion facilitated the decomposition of large collaborative networks into a multitude of collaborative clusters of TFs. In contrast to TF-Cluster, TF-Finder can be used for identifying regulatory TFs involved in a particular biological pathway or a process; this is accomplished using Adaptive Sparse Canonical Correlation Analysis (ASCCA), in combination with a user-supplied regulatory TF knowledge base (Cui, Wang, Chen, Busov, & Wei, 2010). In the web-based TF-Finder pipeline, the knowledge base requirement can be avoided using the Sparse Partial Least Squares (SPLS) algorithm. SPLS was used to recognize candidate regulatory TFs, which are used in place of the existing knowledge base.
TF-Miner is available at: http://sys.bio.mtu.edu/cluster/

Biologists frequently need to determine if a set of motif sequences bound by specific transcription, translation factors are present in the proximal promoters or 3′ untranslated regions (3′ UTRs) of a set of plant genes of interest. In Chapter 4, we developed a web portal, ExactSearch that enables users to search for motif sequences either in a set of custom sequences or the proximal flanking regions of all genes from 50 plant species available in public repositories such as Phytozome.org. The ExactSearch was implemented using a suffix tree-based search algorithm, which can execute an exhaustive search of 100 motifs against 35,000 target sequences (2 kb in length) in 4.2 minutes. This web tool will facilitate the work of plant biologists to identify and elucidate the roles of novel gene regulatory elements. ExactSearch was recently published in *BMC Plant Methods* (Gunasekara et al., 2016).
ExactSearch is available at: http://sys.bio.mtu.edu/motif/

In Chapter 5, a web-based RNA-Seq data visualization platform called STTM JBrowse is presented. STTM JBrowse is a web-based system for sharing/generating visualizations to compare alterations in gene expression by Short Tandem Target Mimic (STTM)

technology, which can target specific microRNAs (miRNAs) for degradation in transgenic plants (Guiliang Tang, 2016; G. Tang et al., 2012). This platform currently includes RNA-Seq data extracted from four plant species, including *Arabidopsis thaliana*, rice, soybean, and maize. Utilizing the STTM JBrowse, differentially expressed genes in several biological pathways were compared between wild-type (WT) and STTM transgenic lines. We adopted an open source genome browser, JBrowse, to implement the STTM JBrowse visualization system (Rat Genome Database, 2015; Skinner, Uzilov, Stein, Mungall, & Holmes, 2009; Westesson, Skinner, & Holmes, 2013).
STTM JBrowse is available at: https://blossom.ffr.mtu.edu/designindex2.php

1.1 Reference List

Cui, X., Wang, T., Chen, H. S., Busov, V., & Wei, H. (2010). TF-finder: a software package for identifying transcription factors involved in biological processes using microarray data and existing knowledge base. *BMC Bioinformatics, 11*, 425. doi:10.1186/1471-2105-11-425

Gunasekara, C., Subramanian, A., Avvari, J. V., Li, B., Chen, S., & Wei, H. (2016). ExactSearch: a web-based plant motif search tool. *Plant Methods, 12*, 26. doi:10.1186/s13007-016-0126-6

Ji, X., Chen, S., Li, J. C., Deng, W., Wei, Z., & Wei, H. (2017). SSGA and MSGA: two seed-growing algorithms for constructing collaborative subnetworks. *Sci Rep, 7*(1), 1446. doi:10.1038/s41598-017-01556-z

Kumari, S., Deng, W., Gunasekara, C., Chiang, V., Chen, H. S., Ma, H., . . . Wei, H. (2016). Bottom-up GGM algorithm for constructing multilayered hierarchical gene regulatory networks that govern biological pathways or processes. *BMC Bioinformatics, 17*(1), 132. doi:10.1186/s12859-016-0981-1

Kumari, S., Nie, J., Chen, H. S., Ma, H., Stewart, R., Li, X., . . . Wei, H. (2012). Evaluation of gene association methods for coexpression network construction and biological knowledge discovery. *PLoS One, 7*(11), e50411. doi:10.1371/journal.pone.0050411

Lin, Y. C., Li, W., Sun, Y. H., Kumari, S., Wei, H., Li, Q., . . . Chiang, V. L. (2013). SND1 transcription factor-directed quantitative functional hierarchical genetic regulatory network in wood formation in Populus trichocarpa. *Plant Cell, 25*(11), 4324-4341. doi:10.1105/tpc.113.117697

Lu, S., Li, Q., Wei, H., Chang, M. J., Tunlaya-Anukit, S., Kim, H., . . . Chiang, V. L. (2013). Ptr-miR397a is a negative regulator of laccase genes affecting lignin content in Populus trichocarpa. *Proc Natl Acad Sci U S A, 110*(26), 10848-10853. doi:10.1073/pnas.1308936110

Nie, J., Stewart, R., Zhang, H., Thomson, J. A., Ruan, F., Cui, X., & Wei, H. (2011). TF-Cluster: a pipeline for identifying functionally coordinated transcription factors

via network decomposition of the shared coexpression connectivity matrix (SCCM). *BMC Syst Biol, 5*, 53. doi:10.1186/1752-0509-5-53

Rat Genome Database. (2015). GBrowse-to-JBrowse Comparison.

Schäfer, J., & Strimmer, K. (2005). *Learning large-scale graphical Gaussian models from genomic data*. Aveiro, PT, August 2004.: The American Institute of Physics.

Skinner, M. E., Uzilov, A. V., Stein, L. D., Mungall, C. J., & Holmes, I. H. (2009). JBrowse: a next-generation genome browser. *Genome Res, 19*(9), 1630-1638. doi:10.1101/gr.094607.109

Tang, G. (2016). Targeting Small RNAs for Destruction in Crops by Short Tandem Target Mimic (STTM)

Tang, G., Yan, J., Gu, Y., Qiao, M., Fan, R., Mao, Y., & Tang, X. (2012). Construction of short tandem target mimic (STTM) to block the functions of plant and animal microRNAs. *Methods, 58*(2), 118-125. doi:10.1016/j.ymeth.2012.10.006

Wei, H., Yordanov, Y., Kumari, S., Georgieva, T., & Busov, V. (2013). Genetic networks involved in poplar root response to low nitrogen. *Plant Signal Behav, 8*(11), e27211. doi:10.4161/psb.27211

Wei, H., Yordanov, Y. S., Georgieva, T., Li, X., & Busov, V. (2013). Nitrogen deprivation promotes Populus root growth through global transcriptome reprogramming and activation of hierarchical genetic networks. *New Phytol, 200*(2), 483-497. doi:10.1111/nph.12375

Westesson, O., Skinner, M., & Holmes, I. (2013). Visualizing next-generation sequencing data with JBrowse. *Briefings in Bioinformatics, 14*(2), 172-177. doi:10.1093/bib/bbr078

Chapter 2

An Efficient Algorithm for Identifying Pathway Regulators through Evaluation of Triple-Gene Mutual Interaction (TGMI) [1]

2.1 Abstract

This chapter introduces a novel algorithm, Triple-Gene Mutual Interaction (TGMI), which can efficiently identify true regulators of biological pathways or processes. The algorithm recognizes significant triple gene blocks, each consisting of a pair of pathway genes (PWGs) and a transcription factor (TF), through using information theory. TGMI evaluates all combinations of triple-gene blocks and identifies those that had potentially regulatory interactions through a measure, which reflects the interaction among three genes. The statistical significance of this measure is determined for each triple gene block by randomized permutation p-value. After that, the frequency of presence of each TF in all those significant triple gene blocks, named the interaction frequency, are calculated. The interaction frequency acts as a likelihood measure for the TF to control the biological pathway or the process of interest. We demonstrated that the TFs with higher interaction frequencies are usually true pathway regulators as validated against the existing biological literature. The comparison of the accuracy of TGMI with those of existing algorithms, including Backward Elimination Random Forest (BWERF) algorithm, Bottom-up Gaussian Graphical Model (Bottom-up GGM), and Algorithm for Reconstruction of Accurate Cellular Networks (ARACNE), was conducted using Receiver Operating Characteristic (ROC) curves. TGMI algorithm was more accurate than other three algorithms and will be instrumental in identifying true pathway regulators, generating regulatory networks and identifying combinatorial TFs.

---

[1] The material contained in this chapter is being prepared for submission to a journal.

2.2 Introduction

Present knowledge indicates that there are several hundred metabolic pathways and a multitude of biological processes in an organism. Our understanding of how these biological pathways are regulated is still limited. For example, *Arabidopsis thaliana* has 549 metabolic pathways and currently, regulators of the majority of these pathways are unknown (Huala et al., 2001; Lv, Cheng, & Shi, 2014; Sweetlove, Last, & Fernie, 2003). This chapter presents a novel efficient algorithm, Triple Gene Mutual Interaction (TGMI), to identify regulatory transcription factors (TFs) that govern a particular pathway by analyzing genome-wide gene expression profiles. TGMI is based on an information theoretic approach to determine which triple gene blocks have significant interactions among the three genes; each consists of a pair of pathway genes and a TF.

Currently available methods for identifying pathway regulators can be broadly categorized into phenotype-driven approaches, such as quantitative trait loci (QTL) mapping and microsatellite analysis, and genotype-driven approaches, which includes gene silencing technology, proteomics, and gene trapping (Marbach et al., 2012; Mitchell-Olds, 2010). Some of these methods focus on individual genes and thus are lab-intensive and time-consuming. With the advent of the whole-genome approach, there has been a heightened demand for efficient statistical and computational methods, which can be used to predict regulatory TFs and relationships between TFs and pathway genes from genome-wide high dimensional gene expression datasets. Techniques such as principal component analyses (PCA) and sliced inverse regression (SIR) have been used to perform dimension reduction for clustering in gene expression microarray datasets (Dai, Lieu, & Rocke, 2006). LASSO regression-based methods (Friedman, Hastie, & Tibshirani, 2010) have also been applied to high dimensional microarray datasets, but biologically interpretable results have not materialized as expected using these methods (Li, Liang, & Zhang, 2014), due to challenges in validating a large number of genes and interactions involved. During the last decade, many systems biological approaches have been developed to identify regulatory relationships between genes. Still, developing methods that can accurately identify true causal relationships have been challenging. Methods that are specialized in identifying regulatory relationships from time-series gene expression data include; Dynamic Bayesian networks (X. H. Chen, Chen, & Ning, 2006; MURPHY, 1999; M. Zou & Conzen, 2005), differential equations (T. Chen, He, & Church, 1999), control logic (Becskei, Seraphin, & Serrano, 2001), Boolean networks (Kauffman, 1969), stochastic networks (B. S. Chen, Chang, Wang, Wu, & Lee, 2011) and finite state linear models (Ruklisa, Brazma, & Viksna, 2005). However, in recent years, gene expression datasets that are available in the public domain have increased dramatically; the majority are static non-time series gene expression datasets, which

include both treatments versus controls data or those with very large time intervals varying from a few hours to several days (Yang & Wei, 2015). To infer regulatory relationships from these kinds of data, only a few highly efficient methods have been developed in the past; for example, Algorithm for the Reconstruction of Accurate Cellular Networks (ARACNE) (Margolin et al., 2006), Backward Elimination Random Forest (BWERF) algorithm (Deng, Zhang, Busov, & Wei, 2017), Bottom-up Graphical Gaussian Model (Bottom-up GGM) algorithm (Kumari et al., 2016). BWERF and Bottom-up GGM are developed and tailored for building multi-layered hierarchical gene regulatory networks (ML-hGRNs) operating above a pathway. BWERF is based on random forest algorithm with a recursive evaluation process that reduces the number of TFs before relationships among genes are established. The Bottom-up GGM algorithm constructs a ML-hGRN using a set of pathway genes as the bottom layer and all TFs as inputs for upper layers. This approach evaluates the significance of interference of an upper-layered candidate TF on two combined genes at the current bottom layer. The interference can be determined by examining if the difference between two correlation coefficients: the correlation coefficient of two bottom-layered genes and the partial correlation coefficient of two bottom-layered genes after the effect of the upper-layered TF is removed. The ARACNE algorithm uses mutual information to identify the dependency relationships between pairwise genes and then implements data processing inequality to remove weakest links. In this study, the accuracy of the TGMI algorithm is evaluated by comparing with those of BWERF, Bottom-up GGM and ARACNE methods.

The proposed triple-gene block in the TGMI algorithm is based on the biological knowledge that genes with similar expression patterns are regulated by the same mechanism (Allocco, Kohane, & Butte, 2004; Clements, van Someren, Knijnenburg, & Reinders, 2007; Yeung, Medvedovic, & Bumgarner, 2004). Existing literature suggests that identification of statistically significant triple gene blocks give important clues about the regulatory TFs, which govern a biological pathway or a process (Kumari et al., 2016; Watkinson, Liang, Wang, Zheng, & Anastassiou, 2009). We hypothesized that by representing this biologically acceptable triple gene block using information theory based mathematical representation, we could create an efficient algorithm to identify positive pathway regulators better than existing algorithms. We utilized mutual information and conditional mutual information to develop a novel measure to represent the interaction among the genes in a triple gene block. Using real world datasets we demonstrated that by evaluating this measure, regulatory relationships among the TF and two pathway genes in a triple gene block could be determined. The TGMI algorithm takes two input data files; 1) A set of expression profiles of genes involved in a known biological pathway (e.g. canonical or non-canonical) or a biological process, and 2) A set of

expression profiles of all TFs under the same experimental condition. The use of differentially expressed pathway genes and TFs may lead to the identification of pathway regulators with a higher accuracy. The output results include, 1) A list of TFs that are sorted in descending order by frequencies of interactions in triple gene blocks; we hypothesized that the top ranked TFs with higher frequencies are more likely to be true pathway regulators, 2) A regulatory network diagram, and 3) a list of combinatorial TFs which regulate each pathway gene. Given the fact that many TFs may control a pathway, it is possible that several TFs may act in combination to govern a single pathway gene (K. B. Singh, 1998). Finally, the algorithm was evaluated by applying it to several biological pathways using microarray gene expression data from *Arabidopsis thaliana* stem tissues and mouse embryonic stem cells. The ranked TFs were compared to those obtained with three other algorithms; BWERF, Bottom-up GGM and ARACNE with the same input data. The results indicated that the TGMI algorithm is more efficient and accurate than all three algorithms we tested. Thus, this novel algorithm will be instrumental to the analysis of gene expression data for the biological research community.

2.3 Materials and Methods

2.3.1 Data

2.3.1.1 *Arabidopsis thaliana* Microarray Gene Expression Data

A compendium dataset (128 microarray samples) was pooled from several microarray datasets generated from hypocotyledonous stem tissues under short-day condition known to induce wood formation (Chaffey, Cholewa, Regan, & Sundberg, 2002). These datasets were obtained from the NCBI GEO repository (http://www.ncbi.nlm.nih.gov/geo/) with the following accession numbers: GSE 607, GSE 6153, GSE 18985, GSE 2000, GSE 24781, and GSE 5633. The platform for these datasets are Affymetrix 25k ATH1 microarrays. The original CEL files for all 128 chips were downloaded and then processed with the Robust Multi-array Analysis (RMA) algorithm available at https://www.bioconductor.org (Irizarry et al., 2003). A previously published method was used to perform quality control of the datasets (Persson, Wei, Milne, Page, & Somerville, 2005).

2.3.1.2 Mouse Microarray Gene Expression Data

A mouse microarray gene expression dataset related to the pluripotency maintenance pathway was downloaded from the Embryonic Stem Cells Atlas of Pluripotency Evidence (ESCAPE) repository. This time-course dataset contains data from following time points: 0 h, 6 h, 12 h, 18 h, 24 h, 36 h, 48 h, 4 days, 9 days and 14 days generated using the Affymetrix MOE439A arrays from embryonic stem cells (ESCs) under undirected differentiation. Validated regulatory relationships were obtained from ChIP-X studies available in the ESCAPE repository. The 24 pluripotency maintenance pathway genes and 35 known regulatory TFs were used to construct three datasets, with 100, 200, or 300 random noise variables (genes) being added to evaluate the performance of TGMI algorithm.

2.3.2 Triple-Gene Mutual Interaction Algorithm

The interaction of a regulatory TF with a pair of pathway genes is captured as a significant interaction among three genes in a triple gene block. A novel measure was developed to quantify the interaction employing information theory (Thomas M. Cover & Joy A. Thomas, 2006). In the triple gene block, the TF is represented in variable X and the pair of pathway genes are represented in variable Y1 and variable Y2 (see Figure 2.1A). First, the gene expression data for each variable was discretized utilizing the equal frequency discretization algorithm (Boulle, 2005). The entropy (H) of each variable was calculated as,

$$H(X) = -\sum_x p(x) log\, p(x) = -E[log(p(x))],$$

Where x represents each discretized value in X and $p(x)$ is the probability mass function.

Similarly, $\mathrm{H(Y1)}$ and $\mathrm{H(Y2)}$ can be calculated.

The mutual information between each pair of variables, including$(Y1, Y2)$, $(Y1, X)$, and $(Y2, X)$ are calculated based on the following formulas. For the $(Y1, Y2)$ pair;

Conditional entropy, $H(Y1|Y2)$:

$$H(Y1|Y2) = -\sum_{y_1, y_2} p(y_1, y_2) log p(y_1|y_2)$$

Joint entropy, $H(Y1, Y2)$:

$$H(Y1, Y2) = H(Y1|Y2) + H(Y2)$$

Mutual information, $(Y1;\ Y2)$:

$$I(Y1; Y2) = H(Y1) + H(Y2) - H(Y1, Y2)$$

Similarly, mutual information for the other pairs of variables, $(X, Y1)$ and $(X, Y2)$ can be calculated.

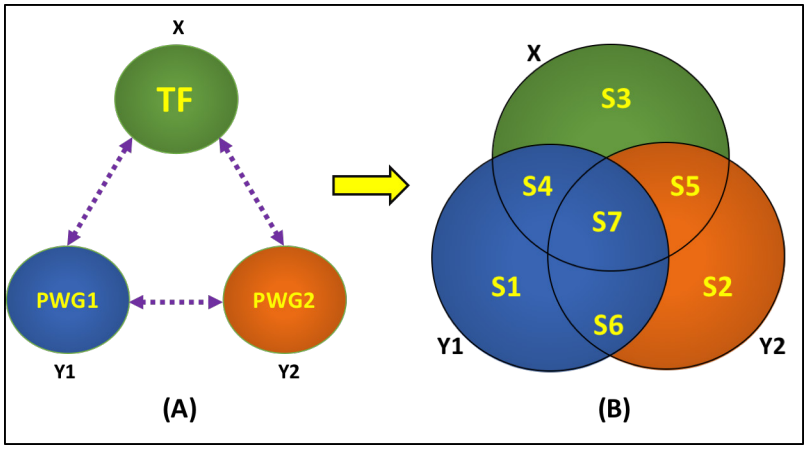


Figure 2.1 Representation of information theoretic quantities among three genes in the triple gene block. Pathway gene 1 (PWG1) and pathway gene 2 (PWG2), and the TF are represented by Y1, Y2, and X, respectively. (A)The interactions among genes in the triple genes block. (B) The Venn diagram segments S1 to S7. S1 = H(Y1|Y2,X) = conditional entropy of Y1 given Y2 and X. S2 = H(Y2|Y1,X) = conditional entropy of Y2 given Y1 and X. S3 = conditional entropy of X given Y1 and Y2. S4 = conditional mutual information of Y1,X given Y2. S5 = conditional mutual information of Y2,X given Y1. S6 = conditional mutual information of Y1,Y2 given X. S7 = difference of mutual information between Y1,Y2 and conditional mutual information Y1,Y2 given X.

The conditional entropies of each variable given the other two variables, $H(Y1|X, Y2), H(Y2|X, Y1), H(X|Y1, Y2)$, are represented in S1, S2, and S3 segments of Figure 2.1B. These quantities, were calculated using the definition of multivariate conditional entropy (Thomas M. Cover & Joy A. Thomas, 2006) as follows.

Joint entropy, $H(Y1, Y2, X)$:

$$H(Y1, Y2, X) = -\sum_{y_1, y_2, x} p(y_1, y_2, x) log p(y_1, y_2, x)$$

Conditional entropy (S1 in Figure 2.1B), $H(Y1|X, Y2)$:

$$H(Y1|X, Y2) = H(Y1, Y2, X) - H(Y2, X) - H(X)$$

Similarly, S2 and S3 can be calculated.

Conditional mutual information (S6 in Figure 2.1B), $(Y1; Y2|X)$:

$$I(Y1; Y2|X) = H(Y1|X) - H(Y1|X, Y2)$$

Then, multivariate mutual information, (S7 in Figure 2.1B), $I(Y1; Y2; X)$ is calculated as follows.

$$I(Y1; Y2; X) = I(Y1; Y2) - I(Y1; Y2|X)$$

If $I(Y1; Y2; X)$ is a positive quantity, this can be represented by S7 in Figure 2.1B. If $I(Y1; Y2; X)$ is a negative quantity, that triple gene block is discarded.

The mutual interaction measure for a triple gene block $\frac{S7}{S1+S2+S3}$ is calculated as follows.

$$\frac{S7}{S1 + S2 + S3} = \frac{I(Y1; Y2; X)}{H(Y1|X, Y2) + H(Y2|X, Y1) + H(X|Y1, Y2)}$$

S1, S2, S3, and S7 are shown in Figure 2.1B.

P-value for each triple gene block was calculated using randomized permutation method (Sham & Purcell, 2014). First, by randomly permuting the data vector of TF in the triple-gene block, 1000 permuted datasets are created. The randomization of the data vector of TF in each triple gene block causes the relationship among the TF and the pairs of genes to be broken. Then, $\frac{S7}{S1+S2+S3}$ measures are calculated for all 1000 randomized triple gene blocks. A p-value for the non-permuted original triple gene block is calculated as the probability of obtaining higher $\frac{S7}{S1+S2+S3}$ measures for permuted triple gene blocks than non-permuted original triple gene block.

Figure 2.2 illustrates the workflow of the TGMI algorithm. Suppose we have q number of TFs, p number of pathway genes and n samples, then the inputs are a TF matrix of n×q dimension, and a pathway gene matrix of n×p dimension. Triple gene mutual interaction measure for each block was calculated as shown in Figure 2.2A. After obtaining p-values for all triple gene combinations, the Benjamini-Hochberg, method (Benjamini & Hochberg, 1995) was used for the correction of multiple testing. Triple gene blocks with corrected p-values is less than the significance level (0.05) are considered significant and kept for the further steps of the algorithm. Figure 2.2B shows three output results, which were generated using the significant triple gene blocks. The first output was a TF list sorted in descending order by the frequencies of interactions. The second output was a network diagram, which shows the regulatory relationships from TFs to pathway genes. The TF nodes in the network were arranged in clock-wise circular direction from most frequently interacting TFs to least frequently interacting TFs. The pathway genes were placed in the middle of the circular network. The third output of the TGMI algorithm was multiple sets of combinatorial TFs; TFs within each set regulate the same pathway gene. To obtain such output, the significant triple gene blocks (PWG1-TF-PWG2) were first merged to form a layered network as shown in output 3 of Figure 2.2B. Then TFs connected to each pathway gene were extracted from the layered network. From this extracted TFs, all combinations of triple gene blocks, which each contains two TFs and one pathway gene (TF1-PWG-TF2), were evaluated using the triple gene mutual interaction measure. The significant triple-gene blocks (TF1-PWG-TF2) were merged to obtain combinatorial TFs for each pathway gene. The frequencies of interactions for each TF in the significant triple gene blocks (TF1-PWG-TF2) for each pathway gene were given within parentheses as shown in final output 3 of Figure 2.2B. The algorithm is summarized in the pseudo code shown in Procedure 2.1.

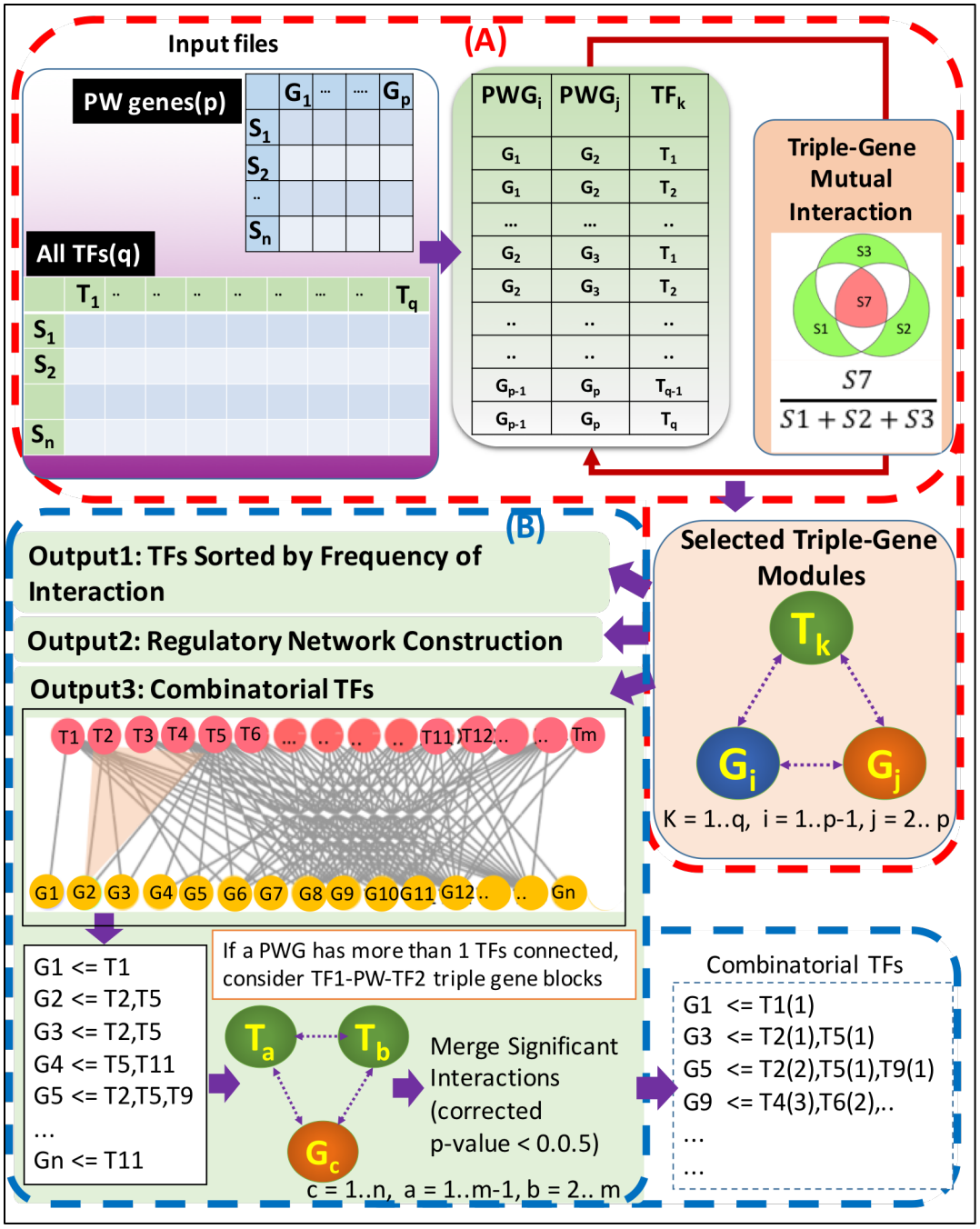


Figure 2.2 Workflow of Triple-gene Mutual Interaction (TGMI) algorithm. (A)The triple gene combinations are evaluated in parallel to accelerate the process shown in the red dashed region, and (B) Three types of results from the algorithm are shown in the blue dashed region. T letters represent TFs, and G letters represent pathway genes. The frequencies of interactions of TFs involved in significant two TF one-pathway gene blocks (TF1-PWG-TF2) are given within parentheses.

Algorithm 1 Triple-gene algorithm

procedure
  Input: (PWG, TFs) with m number of samples
  for each $(P_i, P_j), [(P_i, P_j) \in PWG]$ **do**
    for each $TF_k \leftarrow TFs$ **do**
      a: obtain the number of bin $n_b = \sqrt[3]{m}$
      b: Partition the interval $[a, b]$ in to n_b bins
      c: Put samples in to bins as each bin get $\frac{m}{n_b}$ points
      d: Change the data points to the bin number
      $\frac{S7}{S1+S2+S3}$ = Triple_Gene_Interaction (P_i, P_j, TF_k)
      if $\frac{S7}{S1+S2+S3} > 0$ **then**
        Calculate permutation P-value
        Save the three gene combination
      else
        Discard Combination
      end if
    end for
  end for
  Output 1: Selected Triple gene combinations (Adj.P-Value < 0.05)
  Input: Significant Triple Gene combinations from output1
  Set Significant Level for cutoff (default: $\alpha = 0.05$)
  Merge PW-TF-PW Triple-gene modules to obtain the TF-PW connections
    a: Obtain TF- PW edge strength using $\frac{S7}{S1+S2+S3}$
    b: Build the network with TF-PW edges
  Sort by TF-PW edge strength
  Calculate TF-PW-TF interaction for 3 genes
  Merge TF-PW-TF combinations which have significant levels < 0.05 to extract the combinatorial TF set
  Output 2: TF combinations associated with each pathway gene
end procedure

Procedure 2.1 TGMI pseudo code

2.4 Results

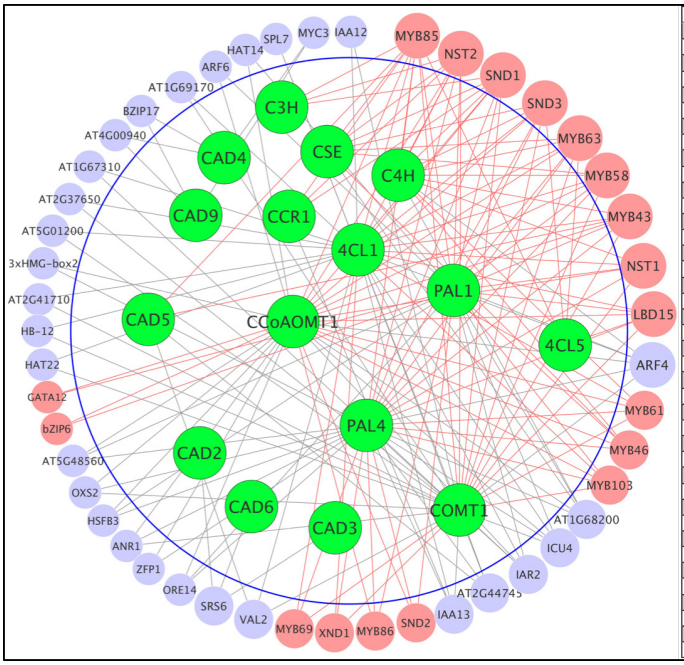
The TGMI algorithm was tested for identification of pathway regulators and building regulatory networks surrounding several biological pathways in *Arabidopsis thaliana* and mouse embryonic stem cells. The pathway genes are non-regulatory genes that can be found in public repositories (*e.g*. https://www.arabidopsis.org/). Alternatively, genes involved in a biological process, for example, those defined by a gene ontology term of biological process, can also be used as non-canonical pathway genes in the algorithm.

2.4.1 Lignin biosynthesis pathway (*Arabidopsis thaliana*)

Lignin is the second most abundant plant biopolymer present in the secondary cell walls and fibers in wood (Dixon & Paiva, 1995; Vanholme, Demedts, Morreel, Ralph, & Boerjan, 2010). Understanding how lignin is synthesized has long been a research focus of plant biologists and the wood industry due to its importance in plant structural integrity and stem stiffness (Chabannes et al., 2001; Donaldson, 2001). To identify pathway regulators that govern lignin biosynthesis, we used a compendium dataset comprises 128 microarray samples. The data in this compendium were generated from *Arabidopsis thaliana* hypocotyledonous stem tissues under short-day conditions, which can induce secondary wood formation in hypocotyls. The expression data of lignin pathway genes and all TFs were extracted from this compendium data for analysis with TGMI algorithm.

The TGMI algorithm extracted triple-gene blocks based on the user-defined significance level of 0.05 (i.e. the cut-off). The table on the right in Figure 2.3 shows top TFs ranked by frequencies of interactions with the lignin pathway genes in descending order. The TFs highlighted in red are known positive TFs that are evidenced by literature to regulate lignin biosynthesis pathway. SND1 is a higher-level regulator which has been evidenced to control SND2, SND3, MYB103, MYB85, MYB52, MYB54, MYB69, MYB42, MYB43, MYB86, MYB61, MYB46, MYB20, and KNAT7 (Lin et al., 2013; R. Zhong, Richardson, & Ye, 2007; R. Zhong & Ye, 2015). Out of these 15 TFs, TGMI algorithm identified 9 TFs (SND1, SND2, SND3, MYB103, MYB85, MYB43, MYB46, MYB86, MYB61) were identified by the TGMI algorithm. NST1, NST2, VND6, and VND7 are functional homologs of SND1 that regulate the same downstream targets in different cell types (R. Zhong, Lee, Zhou, McCarthy, & Ye, 2008). Our algorithm was able to recognize NST1 and NST2. Furthermore, MYB58 and MYB63, which are transcriptional activators of lignin biosynthesis in the SND1-mediated transcriptional regulatory network (J. Zhou, Lee, Zhong, & Ye, 2009), were identified by TGMI algorithm. In addition,

LBD15 (Shuai, Reynaga-Pena, & Springer, 2002a), XND1 (Zhao, Avci, Grant, Haigler, & Beers, 2008), bZIP6 (R. Zhong & Ye, 2012) and GATA12 (Nishitani & Demura, 2015) that are involved in regulating different aspects of secondary cell wall synthesis were also identified by TGMI algorithm. The triple gene blocks with significant regulatory interactions were combined to generate a circular network that is shown in Figure 2.3. The TFs were arranged in the clock-wise direction from the most to the least frequent TFs. The directed edges from TFs to pathway genes represent regulatory relationships. The known positive lignin pathway regulatory TFs highlighted in light coral are highly connected to lignin biosynthesis pathway genes.



Rank	TF	Frequency	References
1	MYB85	11	(Zhou, Lee, Zhong, & Ye, 2009)
2	NST2	9	(Zhong & Ye, 2015)
3	SND1	9	(Zhong & Ye, 2015)
4	MYB63	8	(Zhou et al., 2009)
5	SND3	8	(Zhong, Lee, Zhou, McCarthy, & Ye, 2008)
6	MYB43	7	(Yoon, Choi, & An, 2015)
7	MYB58	7	(Zhou et al., 2009)
8	ARF4	6	-
9	LBD15	6	(Shuai, Reynaga-Pena, & Springer, 2002)
10	NST1	6	(Mitsuda et al., 2007)
11	AT1G68200	5	-
12	IAR2	5	-
13	ICU4	5	-
14	MYB103	5	(Ohman et al., 2013)
15	MYB46	5	(Kim, Kim, Ko, Kang, & Han, 2014)
16	MYB61	5	(Zhong & Ye, 2009)
17	AT2G44745	4	-
18	IAA13	4	-
19	ANR1	3	-
20	AT5G48560	3	-
21	HSFB3	3	-
22	MYB69	3	(Zhong et al., 2008)
23	MYB86	3	(Zhong et al., 2008)
24	ORE14	3	-
25	OXS2	3	-
26	SND2	3	(Hussey et al., 2011)
27	SRS6	3	-
28	VAL2	3	-
29	XND1	3	(Zhao, Avci, Grant, Haigler, & Beers, 2008)
30	3xHMG-box2	2	-
31	ARF6	2	-
32	AT1G67310	2	-
33	AT1G69170	2	-
34	AT2G37650	2	-
35	AT2G41710	2	-
36	AT4G00940	2	-
37	AT5G01200	2	-
38	BZIP17	2	-
39	bZIP6	2	(Zhong & Ye, 2012)
40	GATA12	2	(Nishitani & Demura, 2015)

Figure 2.3 Regulatory network generated by TGMI algorithm for the *Arabidopsis thaliana* lignin biosynthesis pathway from hypocotyledonous stem tissues. Green nodes represent pathway genes. All other nodes are TFs regardless of their colors. Light coral nodes represent positive TFs. References are provided for those TFs that are evidenced to regulate lignin pathway.

ROC curves were used compare the accuracy of the TGMI algorithm with the other three algorithms, which include BWERF, Bottom-up GGM, and ARACNE. First, true positive rate (TPR) values and false positive rate (FPR) values were calculated for all possible cut points and ROC curves were plotted as shown in Figure 2.4. If a ROC curve, first closely follows the TPR axis, and then closely follows top FPR axis, the identification of positive

TFs by the algorithm are more accurate, but if the curve more closer to the 45-degree diagonal line, the identification of positive TFs by the algorithm are less accurate. As shown in Figure 2.4, TGMI algorithm has a much higher accuracy compared to other three algorithms. To show the differences quantitatively, area under the ROC curves (AUROCs) were calculated (*see* Table 2.1). AUROC values can vary from lowest value of 0.5 to highest value of 1. AUROC of 1 indicates the method has a identified of all positive TFs and AUROC of 0.5 indicate the method failed to identify any positive TFs. The significantly larger AUROC of TGMI algorithm (0.92) supports that it has a better performance than other three algorithms in identifying lignin pathway regulators. Note that the performance of some algorithms like BWERF and Bottom-up GGM is based on only one layer of TFs, but these algorithms were designed and tailored to build ML-hGRN.

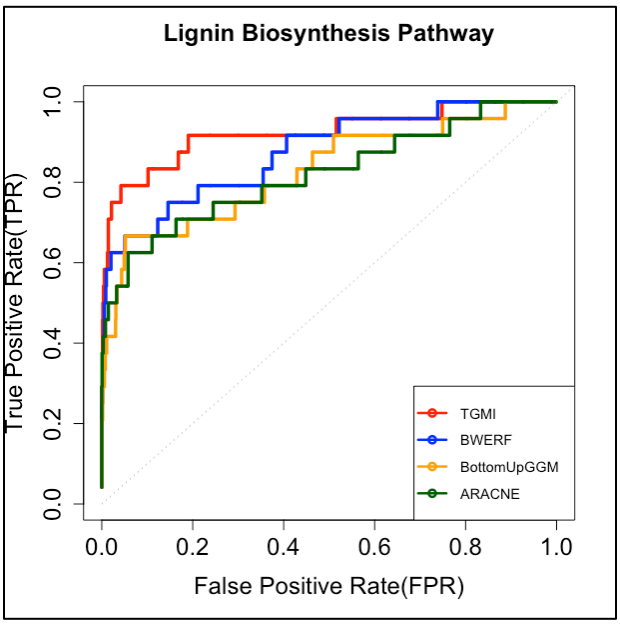


Figure 2.4 The comparison of TGMI with the other three algorithms in recognition of lignin pathway regulators using receiver operating characteristic (ROC) curves. The ROC curves, which first closely follow the TPR axis and then, closely follow the top FPR axis, reflect higher accuracies in identifying positive regulatory TFs. The ROC curves, which are closer to the 45-degree diagonal line, reflect lesser accuracy in identifying positive regulatory TFs.

Table 2.1 The areas under the ROC curves (AUROCs) of TGMI and other algorithms in recognition of lignin pathway regulators (*Arabidopsis thaliana*). An AUROC of 1 indicates the method perfectly identified of all positive TFs and AUROC of 0.5 indicate the method failed to identify positive TFs in when ranked by the frequencies of interactions.

Method	**TGMI**	**BWERF**	**BottomUpGGM**	**ARACNE**
AUC	0.9223461	0.874843	0.7847574	0.7976863

Figure 2.5 shows the combinatorial TFs extracted for each pathway gene. The frequencies of each TF in triple gene blocks (TF1-PWG-TF2) with significant interactions are given in parentheses next to each TF. The results of combinatorial regulations are in agreement with existing literature. For example, the algorithm identified MYB85, MYB53, MYB63 as combinatorial TFs for, PAL1/4, COMT and CCoAMT1 lignin pathway genes, which are supported by existing literature(Hussey, Mizrachi, Creux, & Myburg, 2013). Additionally, MYB103 and MYB46 combinatorial TFs are direct regulators of PAL1/4 pathway genes, which is consistent with the existing biological literature (Hussey et al., 2013). SND1 is a higher level regulator which appears in many combinatorial TF, as shown in Figure 2.5. Existing literature suggests SND1 regulate many lignin pathway genes including PAL1, CCoAOMT, and 4CL1(Ohashi-Ito, Oda, & Fukuda, 2010). The results show NST2 and MYB58 combinatorial TFs are directly related to C3H and C4H genes. This is consistant with the earlier conclusion that C3H and C4H are directly regulated by higher level TFs which include NST2 and MYB58 (Poovaiah, Nageswara-Rao, Soneji, Baxter, & Stewart, 2014). The algorithm identified 4CL1/5 pathway genes that are directly regulated by several MYB domain TFs which includes combinatorial TFs, MYB85, MYB58, and MYB63 as shown in Figure 2.5. Literature evidence shows these three TFs regulate many 4CL family of genes including 4CL1/5 (Y. Liu et al., 2017). CAD and CCR genes are coordinately regulated by many MYB binding sites (Rahantamalala et al., 2010). However, MYB43 and VAL1 were the only combinatorial TFs identified by the algorithm. Literature evidence show MYB43 as one of the regulatory TFs which directly regulates CAD family of genes (Thevenin et al., 2011). Existing literature suggests that CCR family of pathway genes are directly regulated by secondary wall thickening TFs, which include SND1, NST2 (Mitsuda & Ohme-Takagi, 2008). Our algorithm also identified this combinatorial TFs are regulating CAD genes as well as many lignin pathway genes (*see* Figure 2.5). Based on literature evidence that in later steps of the lignin biosynthesis, CCR and CAD family of genes have a milder influence on lignin deposition (Yoon, Choi, & An, 2015). This

fact is confirmed by the relatively low frequency combinatorial of TFs (SND1, NST2) regulating CAD family of genes identified by the algorithm as shown in Figure 2.5.

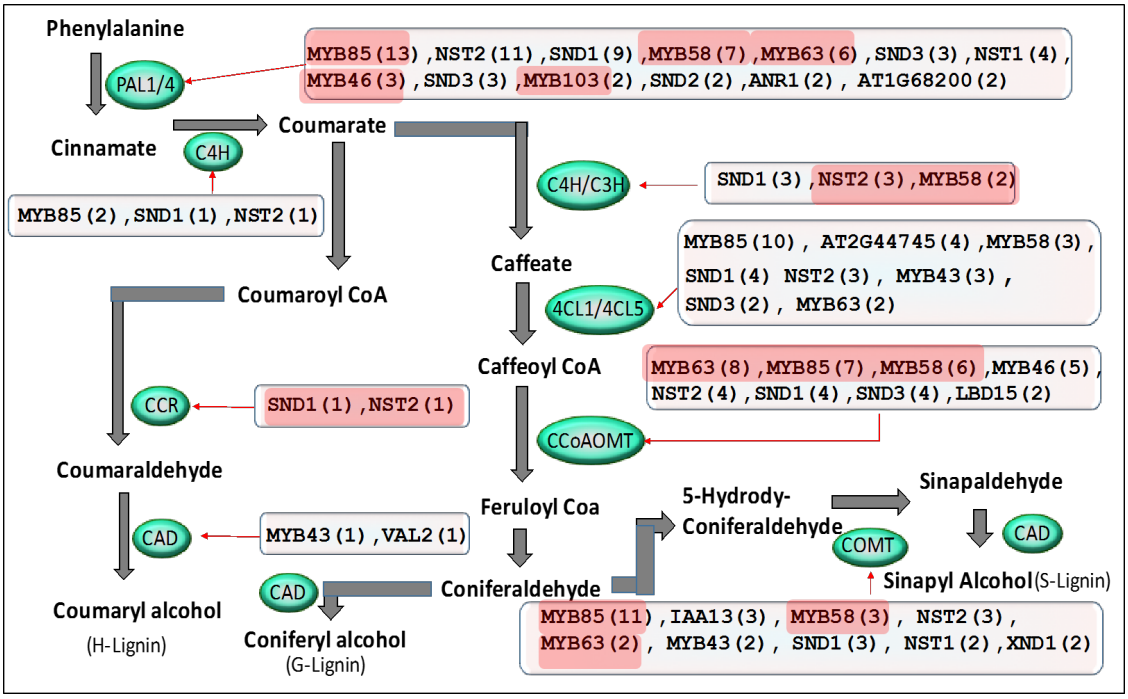


Figure 2.5 Display of combinatorial TFs within the pathway diagram of lignin biosynthesis. The green oval shapes show pathway genes involved in the lignin biosynthesis. The combinatorial TFs are shown in squares. The frequencies of interactions for each TF in significant triple gene blocks (TF1-PWG-TF2) are given in parentheses next to each TF. The combinatorial TFs, which have supporting literature evidence for regulating pathway genes, are highlighted in the squares.

2.4.2 Pigment biosynthesis pathways (*Arabidopsis thaliana*)

In plants, pigments provide a broad range of colors from red/orange to blue/violet and serve as important compounds that attract insects for pollination and act as a protectant against UV-B radiation(Tanaka, Sasaki, & Ohmiya, 2008). Literature evidence suggests coordinated activity among four biosynthesis pathways synthesizes plant pigments. For example, leucopelargonidin and leucocyanidin are colorless intermediates, which are synthesized in the course of colored anthocyanin pigmentation (Springob, Nakajima, Yamazaki, & Saito, 2003). Chemical reactions on leucopelargonidin and leucocyanidin compounds result in red/pink anthocyanin pigmentation in a variety of plants including

Arabidopsis thaliana (Burbulis & Winkel-Shirley, 1999). Flavonoids are modified by series of chemical reactions contributing to pigmentation in seeds and flowers (Forkmann & Martens, 2001). Visible patterns of anthocyanin in plants are intermediated by chemical compounds synthesized by flavonol biosynthesis (Martens, Teeri, & Forkmann, 2002). In this study, anthocyanin biosynthesis and the three related pathways, which include flavonol, flavonoid, leucopelar-gonidin and leucocyanidin biosynthesis, were combined as a unified pigment biosynthesis pathway for identifying pathway regulators. First, co-expression analysis was carried out to identify co-expressed gene pairs across these several pigment related pathways, and significantly co-expressed pathway gene pairs were used in triple gene blocks. Figure 2.6 shows the co-expression among these four pigmentation-related pathways using four different pair-wise association methods, Spearman rank correlation coefficient, Pearson product moment correlation coefficient, Kendall rank correlation coefficient (Kumari et al., 2012) and Maximum Information Coefficient(MIC) (Reshef et al., 2011). Among these methods, Spearman and Kendall can capture monotonic relationships, whereas MIC can capture varying degrees of linear and non-linear relationships between genes. For the co-expression analysis, pathway genes for the four pigment synthesis pathways (anthocyanin, flavonol, flavonoid, leucopelar-gonidin and leucocyanidin biosynthesis) were obtained from the "aracyc_pathways.20140902.txt" file available in the www.arabidopsis.org repository. All significant co-expressed pathway gene pairs (after removing duplicated pairs), identified by four pair-wise association methods were used in triple gene blocks for identifying pigment pathway gene regulators.

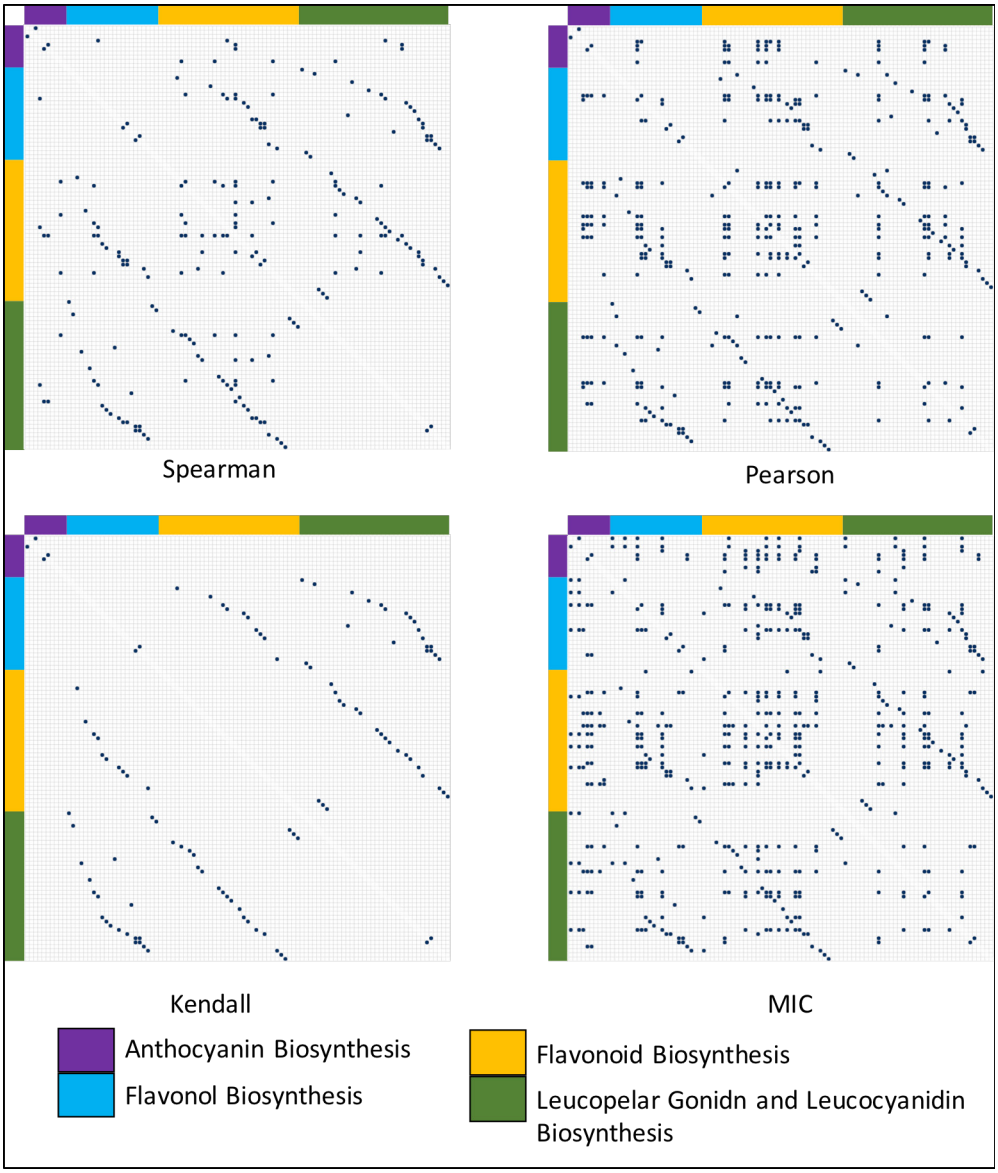
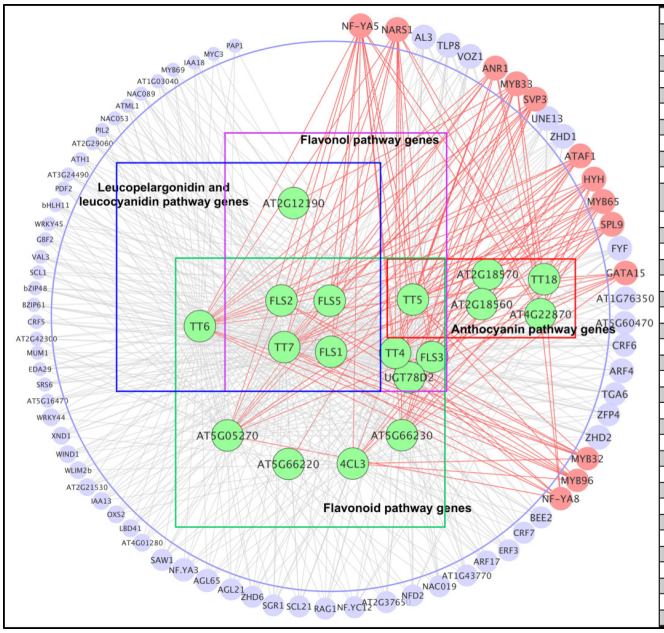


Figure 2.6 Co-expression analysis to reduce pigment synthesis pathway gene combinations using four different pair-wise association methods. Spearman rank correlation coefficient, Pearson correlation coefficient, Kendall rank correlation coefficient and Maximum Information Coefficient (MIC), to identify co-expressed pigment biosynthesis pathway genes from anthocyanin, flavonol, flavonoid, and leucopelar-gonidin and leucocyanidin biosynthesis pathways. Black dots indicated significant pair-wise associations.



Rank	Var1	Freq	Reference
1	NFYA5	14	(W. X. Li et al., 2008)
2	NARS1	12	(Kunieda et al., 2013)
3	AL3	11	-
4	TLP8	11	-
5	VOZ1	11	-
6	ANR1	10	(Kovinich et al., 2012)
7	MYB33	10	(Ambawat, Sharma, Yadav, & Yadav, 2013)
8	SVP3	10	(Wu et al., 2014)
9	UNE13	10	-
10	ZHD1	10	-
11	ATAF1	9	(Mahmood, Xu, El-Kereamy, Casaretto, & Rothstein, 2016)
12	HYH	9	(Zhang, Zheng, Liu, Wang, & Bi, 2011)
13	MYB65	9	(Ambawat et al., 2013)
14	SPL9	9	(Chao et al., 2017)
15	FYF	9	-
16	GATA15	8	(Manfield, Devlin, Jen, Westhead, & Gilmartin, 2007)
17	AT1G76350	8	-
18	AT5G60470	8	-
19	CRF6	8	-
20	ARF4	8	-
21	TGA6	8	-
22	ZFP4	8	-
23	ZHD2	8	-
24	MYB32	7	(Lotkowska et al., 2015)
25	MYB96	7	(Seo & Park, 2010)
26	NFYA8	7	(S. Li et al., 2016)
27	BEE2	7	-
28	CRF7	7	-
29	ERF3	7	-
30	ARF17	7	-
31	AT1G43770	7	-
32	NAC019	7	-
33	NFD2	7	-
34	AT2G37650	7	-
35	NF.YC12	7	-
36	RAG1	7	-
37	SCL21	7	-

Figure 2.7 Regulatory network generated by TGMI for *Arabidopsis thaliana* unified pigment biosynthesis pathway that consists of anthocyanin, flavonol, flavonoid, and leucopelar-gonidin and leucocyanidin biosynthesis pathways with a compendium microarray data set (128 chips) from *Arabidopsis thaliana*. Green nodes represent pathway genes. All other nodes are TFs regardless of their colors. Light coral nodes represent positive TFs. References are provided for those TFs that are evidenced to regulate pigment synthesis pathways.

The regulatory network constructed to manifest the relationships between genes involved in the unified pigment pathway is illustrated in Figure 2.7. The TFs that have higher frequent dependency with pigment biosynthesis pathway genes are listed at the right side of Figure 2.7. The NFYA5 identified by the TGMI algorithm is a stress-sensitive regulator related to anthocyanin synthesis that regulates purple pigmentation under drought conditions (W. X. Li et al., 2008). NARS1 is involved in anthocyanin pigmentation of the epidermal cells of the *Arabidopsis thaliana* (Kunieda et al., 2013). ANR1 and ANR2, have shown to induce over accumulation of flavonoid intermediates which suppresses anthocyanin pathway genes (Kovinich et al., 2012). MYB33 and MYB65 are involved in anthocyanin accumulation and seed color pigmentation (Ambawat, Sharma, Yadav, & Yadav, 2013). Overexpression of the SVP3 gene in kiwifruit has been shown to interfere with anthocyanin biosynthesis in petals (Wu et al., 2014). Literature evidence suggests ATAF1 is involved in anthocyanin synthesis in *Arabidopsis thaliana* in adverse growth conditions (Mahmood, Xu, El-Kereamy, Casaretto, & Rothstein, 2016). HYH is an *Arabidopsis* bZIP transcription factor directly

involved in anthocyanin and chlorophyll estimation (Zhang, Zheng, Liu, Wang, & Bi, 2011). SPL9 negatively regulates anthocyanin by directly suppressing anthocyanin biosynthesis genes (Gou, Felippes, Liu, Weigel, & Wang, 2011a). GATA 15 is involved in various activities to modify chlorophyll pigment content in response to different environmental conditions (Xu, 2016). Literature evidence shows MYB32 indirectly regulate anthocyanin biosynthesis through MYB112, which is a known regulator of anthocyanin pathway (Lotkowska et al., 2015). Also MYB96 is a drought stress response regulator, which has an effect on anthocyanin synthesis in *Arabidopsis Thaliana(Seo & Park, 2010).* NFYA8 was found to be a major regulator of tomato ripening; the literature indicates that this TF is also present in *Arabidopsis thaliana* (Seo & Park, 2010).

ROC curves were used compare the accuracy of the TGMI algorithm with the other three algorithms as shown in Figure 2.8. Figure 2.8 indicate that TGMI algorithm has a higher accuracy in comparison to the other three algorithms based on the ROC curves. To show the differences quantitatively, area under the ROC curves (AUROCs) were calculated (*see* Table 2.2). The AUROC of TGMI algorithm for the identification of pigment biosynthesis regulatory TFs (0.8), supports that it has a better performance than the other three algorithms. Note that the performance of some algorithms like BWERF and Bottom-up GGM is based on only one layer of TFs, but these algorithms were designed and tailored to build ML-hGRNs.

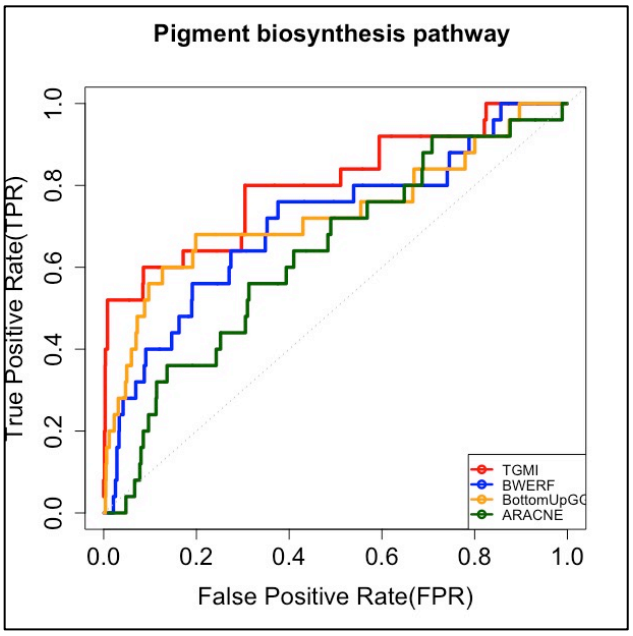


Figure 2.8 The comparison of TGMI with the other three algorithms in recognition of pigment pathway regulators using reciever operating characteristic (ROC) curves. ROC curves, which closely follow the TPR axis, and then, more closely follow the top FPR axis, reflect the higher accuracies in identifying positive regulatory TFs. The ROC curves, which are closer to the 45-degree diagonal line, reflect lesser accuracies in identifying positive regulatory TFs.

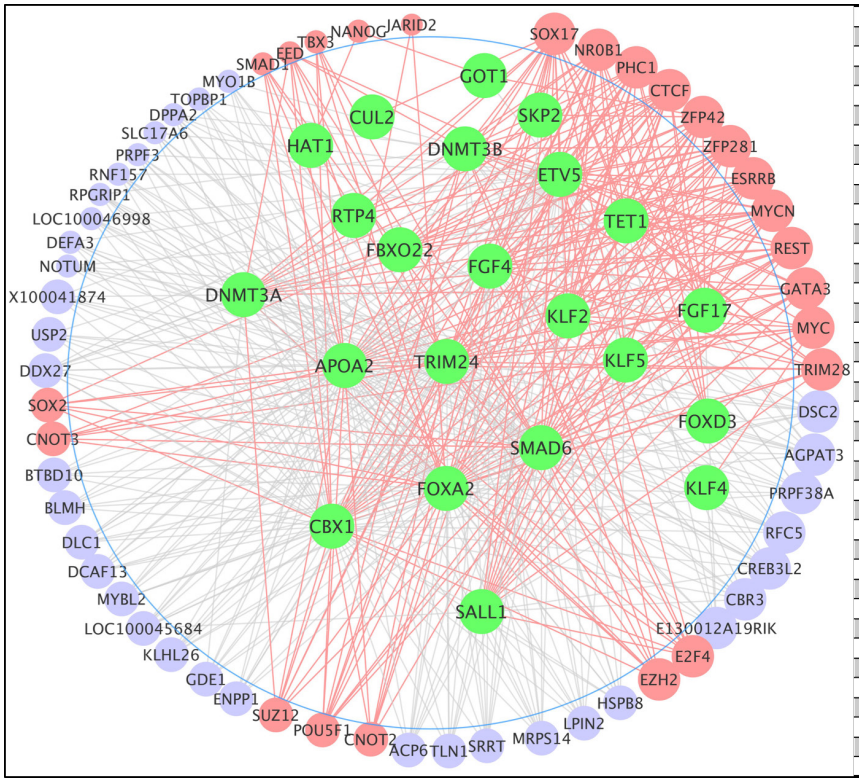
Table 2.2 The area under ROC curves (AUROCs) of TGMI and other three algorithms in recognition of the unified pigment pathway regulators(*Arabidopsis thaliana*). An AUROC of 1 indicates the method perfectly identified of all positive TFs and AUROC of 0.5 indicate the method failed to identify any positive TFs.

Method	**TGMI**	**BWERF**	**BottomUpGGM**	**ARACNE**
AUC	0.8015492	0.7080889	0.7290233	0.63199

2.4.3 Pluripotency maintenance pathway (Mouse embryonic stem cells)

A mouse time-course microarray dataset of 34 samples were downloaded from the Embryonic Stem Cells Atlas of Pluripotency Evidence (ESCAPE) repository, which was used to compare the performance of TGMI with the other three algorithms. Additionally, 24 pluripotency maintenance pathway genes and 35 known positive TFs, which regulate the pathway genes, were obtained from a Chip-Seq study available in ESCAPE web portal. In order to compare the performance of the TGMI algorithm to the other three

algorithms as mentioned above, three datasets; Dataset 1, Dataset 2, and Dataset 3, were created by adding profiles of 100, 200, and 300 randomly selected noise genes respectively, to the 35 known positive TFs. Figure 2.9 illustrates the mouse pluripotency maintenance non-canonical pathway Regulatory network that were identified using the TGMI algorithm using the Dataset 3.



Rank	TF	Freq
1	SOX17	19
2	NR0B1	16
3	PHC1	15
4	CTCF	14
5	ZFP42	14
6	ZFP281	13
7	ESRRB	12
8	MYCN	12
9	REST	12
10	GATA3	10
11	MYC	9
12	TRIM28	9
13	DSC2	9
14	AGPAT3	9
15	PRPF38A	9
16	RFC5	9
17	CREB3L2	9
18	CBR3	8
19	E130012A19RIK	8
20	E2F4	8
21	EZH2	8
22	HSPB8	8
23	LPIN2	8
24	MRPS14	8
25	SRRT	8
26	TLN1	8
27	ACP6	7
28	CNOT2	7
29	POU5F1	7
30	SUZ12	7
31	ENPP1	7
32	GDE1	7
33	KLHL26	7
34	LOC100045684	7
35	MYBL2	7
36	DCAF13	7
37	DLC1	7
38	BLMH	6

Figure 2.9 Regulatory network generated by TGMI for mouse pluripotency maintenance non-canonical pathway using Dataset 3 (335 TF). The green nodes are non-canonical pathway genes, and all others are TFs while the light coral colored nodes represent known positive TFs.

Top positive TFs shown in red color in the table on the right of Figure 2.9 which includes, SOX17(Niakan et al., 2010), NROB1(Fujii et al., 2015), PHC1(Morey, Santanach, & Di Croce, 2015), CTCF(Donohoe, Silva, Pinter, Xu, & Lee, 2009), ZFP42(Masui et al., 2008), ZFP281(Fidalgo et al., 2011), ESRRB(Papp & Plath, 2012), MYCN(Ruiz-Perez, Henley, & Arsenian-Henriksson, 2017), REST(S. K. Singh, Kagalwala, Parker-Thornburg, Adams, & Majumder, 2008), GATA3(Shu et al., 2015), MYC(Chappell & Dalton, 2013),and TRIM28(Miles et al., 2017) are positive TFs out of

the known 35 TFs which are involved with pluripotency maintenance in mouse embryonic stem cells. However, POU5F1, SOX2 and NANOG are well known master regulatory genes that govern the stem renewal in mice (Hall & Hyttel, 2014; Kellner & Kikyo, 2010; Loh et al., 2006; Rodda et al., 2005; Sharov et al., 2008; Q. Zhou, Chipperfield, Melton, & Wong, 2007). Although they were identified by the TGMI, their frequencies of interaction are low, as shown in Figure 2.9. The possible reason is that these three master regulators are located at higher hierarchical levels, and are relatively distal from the pathway genes that were used to identify them.

As shown in Figure 2.10, ROC curves were used to compare the accuracy of TGMI with the other three algorithms using the three datasets. Figure 2.10 indicates that TGMI algorithm has higher accuracies in comparison to the other three algorithms when tested using all three datasets. To show the differences quantitatively, area under the ROC curves (AUROCs) were calculated (*see* Table 2.3). AUROCs of the TGMI algorithm for the identification of pluripotency pathway regulatory TFs using three datasets; Dataset 1, Dataset 2, and Dataset 3 are 0.77, 0.79, 0.8, respectively. These results indicate that TGMI has a higher accuracy than other three algorithms for all three datasets. Note that the performance of some algorithms like BWERF and Bottom-up GGM is based on only one layer of TFs, but these algorithms were designed and tailored to build ML-hGRN.

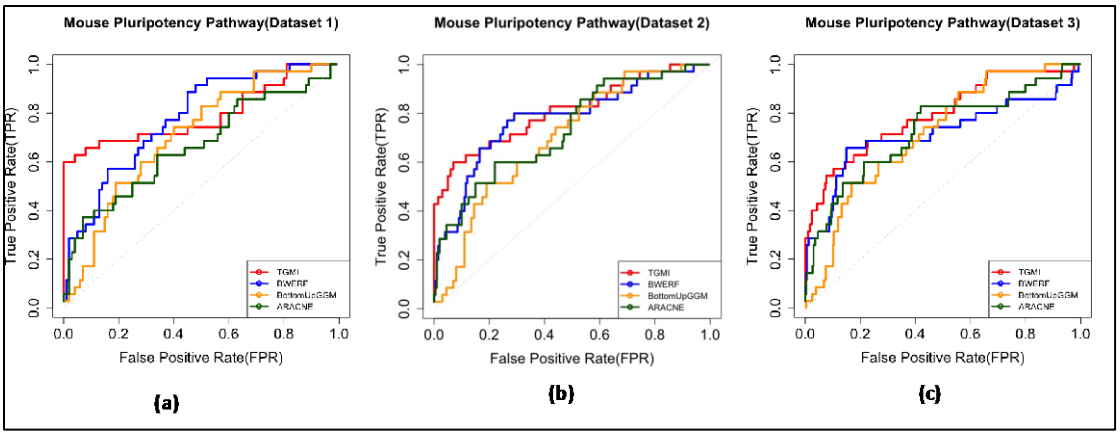


Figure 2.10 The performance of TGMI in comparison with the three algorithms in identifying mouse pluripotency pathway positive regulatory TFs. (a) ROC curves generated using 135 TF dataset (Dataset 1). (b) ROC curves generated using 235 TF dataset (Dataset 2). (c) ROC curves generated using 335 TF dataset (Dataset 3). ROC curves, which closely follow the TPR axis first, and then, closely follow the top FPR axis, reflect the higher accuracies in identifying positive regulatory TFs. The ROC curves, which are closer to the 45-degree diagonal line, reflect lesser accuracies in identifying positive regulatory TFs.

Table 2.3 The area under ROC curves of TGMI algorithm in comparison to other algorithms in recognition positive TFs which regulates the mouse pluripotency maintenance pathway. Adding 100 created 200 and 300 random noise genes to known 35 regulatory TFs, following three datasets created (Dataset1, Dataset2, and Dataset3).

	TGMI	**BWERF**	**BottomUpGGM**	**ARACNE**
Dataset 1	0.7711429	0.7442857	0.698	0.6662857
Dataset 2	0.7895714	0.7331429	0.6945714	0.7168571
Dataset 3	0.802381	0.786	0.7518816	0.7382857

2.5 Discussion

The TGMI algorithm was developed for identifying novel regulators controlling canonical or non-canonical pathways. After testing with three real transcriptome datasets, we found that the TGMI algorithm consistently performed well in identifying regulators of both canonical and non-canonical pathways, through constructing regulatory network between pathway genes and TFs. Additionally, TGMI can unearth combinatorial TFs that interact with each pathway gene. TGMI is based on mutual information and conditional mutual information. The use of mutual information has several advantages in comparison to linear pair-wise association methods because mutual information can be generalized to identify linear or non-linear relationships between two variables (Schneidman, Still, Berry, & Bialek, 2003). When expanded to three variables, the results have proven that the TGMI algorithm is better than Bottom-up GGM, which utilizes the significance of the difference in correlation and partial correlation to identify triple gene blocks. The triple gene interaction measure used in the TGMI algorithm is largely responsible for the high accuracy in determining positive regulatory TFs. The TGMI algorithm has led to;

1) The reduction of the dimensionality of gene space by fitting differentially expressed regulatory genes to a set of pathway genes.
2) The emergence of true regulatory relationships by suppressing spurious relationships.
3) The extraction of linear and non-linear associations between pathway genes and TFs.

It is well established that the detection of causal patterns is more effective in a tri-variate setting than in a bivariate context (Schäfer & Strimmer, 2005). The efficiency and accuracy of evaluating three genes for causal relationships have been demonstrated in two previous publications (Lin et al., 2013; Lu et al., 2013).

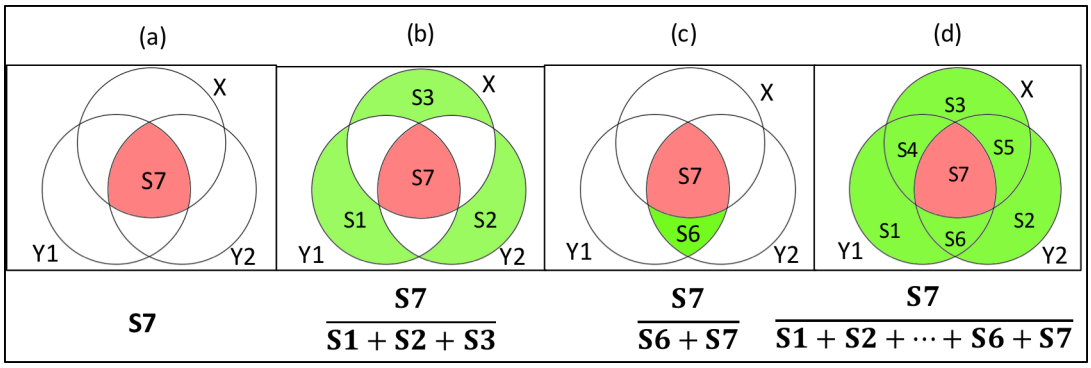


Figure 2.11 Illustration of four interaction measures for a triple-gene block. Y1, Y2 and X represent pathway gene 1, pathway gene 2 and a TF, respectively. S7 = I(Y1;Y2)-I(Y1;Y2|X), S1 = H(Y1|Y2,X) = conditional entropy of Y1 given Y2 and X. S2 and S3 can be similarly defined. Segment 4 (S4) = conditional mutual information of Y1, X given Y2. S5 and S6 can be similarly defined.

A triple gene block can be symbolized as shown in Figure 2.11, which also shows different predictors for quantifying the strength of interaction among three genes. To evaluate the best predictors of positive TFs systematically, out of the several triple gene interaction measures, illustrated in Figure 2.11(a, b, c, and d), we created a dataset using combinations of pairs of pathway genes with all TFs. Each combination in the dataset was given a label of "1" if the TF in the triple gene block was a positive TF and "0" if the TF was a non-positive TF. The dataset was sampled to create a training data partitions to build logistic regression classification models and testing data partitions to predict positive TFs. The prediction performance was measured by determining the area under the ROC curves (AUROCs) for the four different triple gene interaction measures (*see* a, b, c, and, d in Figure 2.11) and a random predictor. As shown in Figure 2.12, $\frac{S7}{S1+S2+S3}$ has a higher prediction performance compared to the other three interaction measures and, of course, the random predictor, for both the lignin and pigment biosynthesis datasets. For the mouse pluripotency maintenance pathways, the result from Dataset 1 (135 TFs) illustrate that S7 and $\frac{S7}{S1+S2+S3}$ have nearly the same AUROC values.

However, for both Dataset 2 (235 TFs) and Dataset 3 (335 TFs), it is obvious that $\frac{S7}{S1+S2+S3}$ has a better predictive performance for predicting positive TFs than the other candidate triple gene interaction measures and obviously the random predictor.

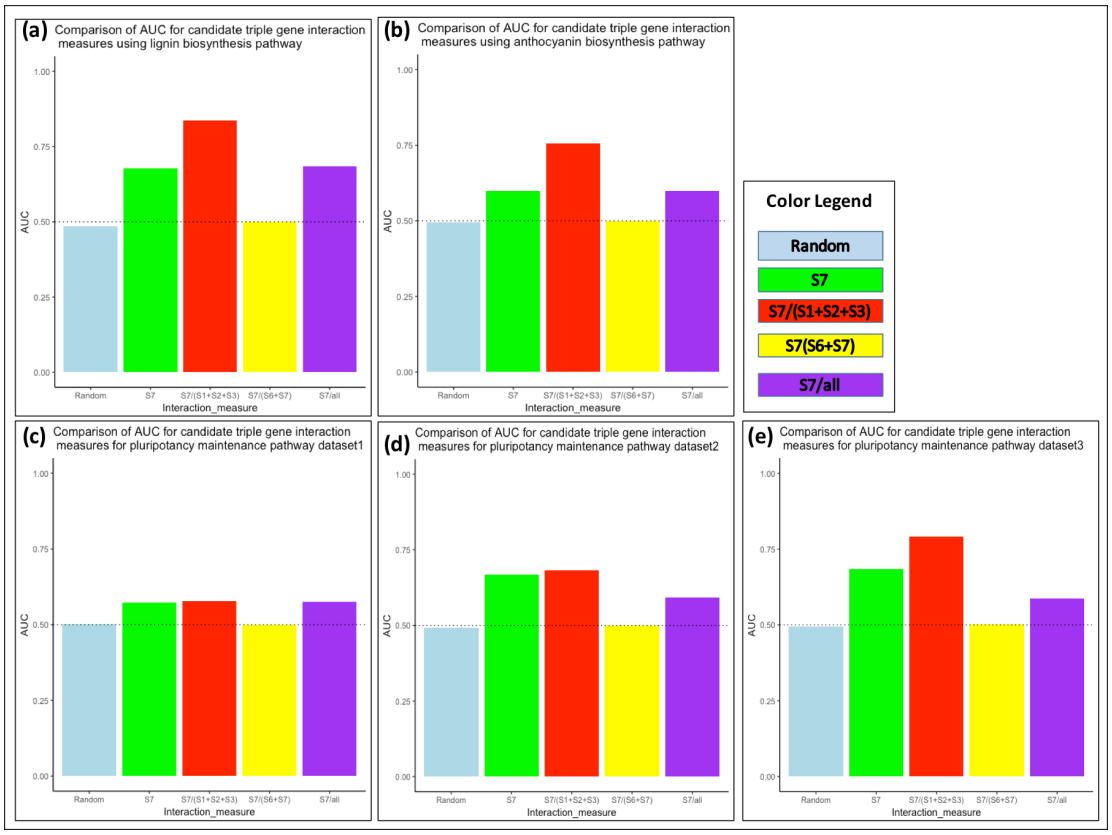


Figure 2.12 The use of areas under ROC curves (AUROCs) to compare efficiencies of four types of triple gene interaction measures using logistic regression model. (a) AUC of lignin pathway regulators identified from TGMI; (b) AUC of pigment biosynthesis pathway regulators identified from TGMI; (c) (d) and (e) are AUCs of mouse pluripotency pathway regulators identified from TGMI from Dataset 1 (135 TFs), Dataset 2 (235 TFs), and Dataset 3 (335 TFs) respectively.

The results indicate the triple gene interaction measure; $\frac{S7}{S1+S2+S3}$ can be used to identify key TFs that govern a biological pathway through evaluating triple gene blocks. Biologists, thereby reducing expensive and time-consuming explorative research, can test significant pathway regulators that emerged from frequencies of interactions on pathway

genes. Additionally, the triple gene blocks can be used to build network structures that convey information regarding cellular mechanisms, combinatorial regulatory behavior, and models of TFs. Computational validations of the TGMI algorithm using real-world microarray datasets have demonstrated that our approach is effective in identifying many positive regulatory genes that are significantly enriched in gene regulatory networks. Our algorithm will be instrumental in constructing regulatory networks and identifying key TFs that govern important biological pathways and processes.

2.6 Conclusion

The TGMI algorithm can be used for identifying pathway regulatory TFs, discovering combinatorial TFs, and constructing regulatory network operating above the pathway. The algorithm accomplishes these objectives through evaluating combined triple gene blocks; each contains two pathway genes and one TF. The gene expression data do not necessarily have to be a time series because we have used pooled compendium to test the algorithm. The algorithm was tested on several real datasets, the results aligned well with the existing literature. Furthermore, performance tests using ROC and AUROC proved that the TGMI algorithm performs better than several existing algorithms. Our method will be instrumental to biologists who are interested in identifying regulators that control biological pathways and processes using gene expression data available in public repositories.

2.7 Reference List

Allocco, D. J., Kohane, I. S., & Butte, A. J. (2004). Quantifying the relationship between co-expression, co-regulation and gene function. *BMC Bioinformatics, 5*. doi:Artn 18

Doi 10.1186/1471-2105-5-18

Ambawat, S., Sharma, P., Yadav, N. R., & Yadav, R. C. (2013). MYB transcription factor genes as regulators for plant responses: an overview. *Physiology and Molecular Biology of Plants, 19*(3), 307-321. doi:10.1007/s12298-013-0179-1

Becskei, A., Seraphin, B., & Serrano, L. (2001). Positive feedback in eukaryotic gene networks: cell differentiation by graded to binary response conversion. *Embo Journal, 20*(10), 2528-2535. doi:DOI 10.1093/emboj/20.10.2528

Boulle, M. (2005). Optimal bin number for equal frequency discretizations in supervized learning. *Intelligent Data Analysis, 9*(2), 175-188.

Burbulis, I. E., & Winkel-Shirley, B. (1999). Interactions among enzymes of the Arabidopsis flavonoid biosynthetic pathway. *Proceedings of the National Academy of Sciences of the United States of America, 96*(22), 12929-12934. doi:DOI 10.1073/pnas.96.22.12929

Chabannes, M., Ruel, K., Yoshinaga, A., Chabbert, B., Jauneau, A., Joseleau, J. P., & Boudet, A. M. (2001). In situ analysis of lignins in transgenic tobacco reveals a differential impact of individual transformations on the spatial patterns of lignin deposition at the cellular and subcellular levels. *Plant Journal, 28*(3), 271-282. doi:DOI 10.1046/j.1365-313X.2001.01159.x

Chaffey, N., Cholewa, E., Regan, S., & Sundberg, B. (2002). Secondary xylem development in Arabidopsis: a model for wood formation. *Physiol Plant, 114*(4), 594-600.

Chappell, J., & Dalton, S. (2013). Roles for MYC in the Establishment and Maintenance of Pluripotency. *Cold Spring Harbor Perspectives in Medicine, 3*(12). doi:ARTN a014381

10.1101/cshperspect.a014381

Chen, B. S., Chang, C. H., Wang, Y. C., Wu, C. H., & Lee, H. C. (2011). Robust model matching design methodology for a stochastic synthetic gene network. *Mathematical Biosciences, 230*(1), 23-36. doi:10.1016/j.mbs.2010.12.007

Chen, T., He, H. L., & Church, G. M. (1999). Modeling gene expression with differential equations. *Pac Symp Biocomput*, 29-40.

Chen, X. H., Chen, M., & Ning, K. D. (2006). BNArray: an R package for constructing gene regulatory networks from microarray data by using Bayesian network. *Bioinformatics, 22*(23), 2952-2954. doi:10.1093/bioinoformatics/btl491

Clements, M., van Someren, E. P., Knijnenburg, T. A., & Reinders, M. J. (2007). Integration of known transcription factor binding site information and gene expression data to advance from co-expression to co-regulation. *Genomics Proteomics Bioinformatics, 5*(2), 86-101. doi:10.1016/S1672-0229(07)60019-9

Cover, T. M., & Thomas, J. A. (2006). *Elements of information theory* (2nd ed.). Hoboken, N.J.: Wiley-Interscience.

Cover, T. M., & Thomas, J. A. (2006). *Elements of Information Theory (Wiley Series in Telecommunications and Signal Processing)*: Wiley-Interscience.

Deng, W., Zhang, K., Busov, V., & Wei, H. (2017). Recursive random forest algorithm for constructing multilayered hierarchical gene regulatory networks that govern biological pathways. *PLoS One, 12*(2), e0171532. doi:10.1371/journal.pone.0171532

Dixon, R. A., & Paiva, N. L. (1995). Stress-Induced Phenylpropanoid Metabolism. *Plant Cell, 7*(7), 1085-1097. doi:DOI 10.1105/tpc.7.7.1085

Donaldson, L. A. (2001). Lignification and lignin topochemistry - an ultrastructural view. *Phytochemistry, 57*(6), 859-873. doi:Doi 10.1016/S0031-9422(01)00049-8

Donohoe, M. E., Silva, S. S., Pinter, S. F., Xu, N., & Lee, J. T. (2009). The pluripotency factor Oct4 interacts with Ctcf and also controls X-chromosome pairing and counting. *Nature, 460*(7251), 128-U147. doi:10.1038/nature08098

Fidalgo, M., Shekar, P. C., Ang, Y. S., Fujiwara, Y., Orkin, S. H., & Wang, J. L. (2011). Zfp281 Functions as a Transcriptional Repressor for Pluripotency of Mouse Embryonic Stem Cells. *Stem Cells, 29*(11), 1705-1716. doi:10.1002/stem.736

Forkmann, G., & Martens, S. (2001). Metabolic engineering and applications of flavonoids. *Curr Opin Biotechnol, 12*(2), 155-160.

Fujii, S., Nishikawa-Torikai, S., Futatsugi, Y., Toyooka, Y., Yamane, M., Ohtsuka, S., & Niwa, H. (2015). Nr0b1 is a negative regulator of Zscan4c in mouse embryonic stem cells. *Scientific Reports, 5*. doi:ARTN 9146
10.1038/srep09146

Gou, J. Y., Felippes, F. F., Liu, C. J., Weigel, D., & Wang, J. W. (2011a). Negative Regulation of Anthocyanin Biosynthesis in Arabidopsis by a miR156-Targeted SPL Transcription Factor. *Plant Cell, 23*(4), 1512-1522. doi:10.1105/tpc.111.084525

Hall, V. J., & Hyttel, P. (2014). Breaking Down Pluripotency in the Porcine Embryo Reveals Both a Premature and Reticent Stem Cell State in the Inner Cell Mass and Unique Expression Profiles of the Naive and Primed Stem Cell States. *Stem Cells and Development, 23*(17), 2030-2045. doi:10.1089/scd.2013.0502

Huala, E., Dickerman, A. W., Garcia-Hernandez, M., Weems, D., Reiser, L., LaFond, F., . . . Rhee, S. Y. (2001). The Arabidopsis Information Resource (TAIR): a comprehensive database and web-based information retrieval, analysis, and visualization system for a model plant. *Nucleic Acids Res, 29*(1), 102-105.

Hussey, S. G., Mizrachi, E., Creux, N. M., & Myburg, A. A. (2013). Navigating the transcriptional roadmap regulating plant secondary cell wall deposition. *Front Plant Sci, 4*, 325. doi:10.3389/fpls.2013.00325

Irizarry, R. A., Hobbs, B., Collin, F., Beazer-Barclay, Y. D., Antonellis, K. J., Scherf, U., & Speed, T. P. (2003). Exploration, normalization, and summaries of high density oligonucleotide array probe level data. *Biostatistics, 4*(2), 249-264. doi:10.1093/biostatistics/4.2.249

Kauffman, S. (1969). Homeostasis and differentiation in random genetic control networks. *Nature, 224*(5215), 177-178.

Kellner, S., & Kikyo, N. (2010). Transcriptional regulation of the Oct4 gene, a master gene for pluripotency. *Histol Histopathol, 25*(3), 405-412. doi:10.14670/HH-25.405

Kovinich, N., Saleem, A., Rintoul, T. L., Brown, D. C., Arnason, J. T., & Miki, B. (2012). Coloring genetically modified soybean grains with anthocyanins by suppression of the proanthocyanidin genes ANR1 and ANR2. *Transgenic Res, 21*(4), 757-771. doi:10.1007/s11248-011-9566-y

Kumari, S., Deng, W., Gunasekara, C., Chiang, V., Chen, H. S., Ma, H., . . . Wei, H. (2016). Bottom-up GGM algorithm for constructing multilayered hierarchical gene regulatory networks that govern biological pathways or processes. *BMC Bioinformatics, 17*(1), 132. doi:10.1186/s12859-016-0981-1

Kumari, S., Nie, J., Chen, H. S., Ma, H., Stewart, R., Li, X., . . . Wei, H. (2012). Evaluation of gene association methods for coexpression network construction

and biological knowledge discovery. *PLoS One, 7*(11), e50411. doi:10.1371/journal.pone.0050411

Kunieda, T., Shimada, T., Kondo, M., Nishimura, M., Nishitani, K., & Hara-Nishimura, I. (2013). Spatiotemporal Secretion of PEROXIDASE36 Is Required for Seed Coat Mucilage Extrusion in Arabidopsis. *Plant Cell, 25*(4), 1355-1367. doi:10.1105/tpc.113.110072

Li, W. X., Oono, Y., Zhu, J. H., He, X. J., Wu, J. M., Iida, K., . . . Zhu, J. K. (2008). The Arabidopsis NFYA5 transcription factor is regulated transcriptionally and posttranscriptionally to promote drought resistance. *Plant Cell, 20*(8), 2238-2251. doi:10.1105/tpc.108.059444

Lin, Y. C., Li, W., Sun, Y. H., Kumari, S., Wei, H., Li, Q., . . . Chiang, V. L. (2013). SND1 transcription factor-directed quantitative functional hierarchical genetic regulatory network in wood formation in Populus trichocarpa. *Plant Cell, 25*(11), 4324-4341. doi:10.1105/tpc.113.117697

Liu, Y., Wei, M., Hou, C., Lu, T., Liu, L., Wei, H., . . . Wei, Z. (2017). Functional Characterization of Populus PsnSHN2 in Coordinated Regulation of Secondary Wall Components in Tobacco. *Sci Rep, 7*(1), 42. doi:10.1038/s41598-017-00093-z

Loh, Y. H., Wu, Q., Chew, J. L., Vega, V. B., Zhang, W., Chen, X., . . . Ng, H. H. (2006). The Oct4 and Nanog transcription network regulates pluripotency in mouse embryonic stem cells. *Nat Genet, 38*(4), 431-440. doi:10.1038/ng1760

Lotkowska, M. E., Tohge, T., Fernie, A. R., Xue, G. P., Balazadeh, S., & Mueller-Roeber, B. (2015). The Arabidopsis Transcription Factor MYB112 Promotes Anthocyanin Formation during Salinity and under High Light Stress. *Plant Physiol, 169*(3), 1862-1880. doi:10.1104/pp.15.00605

Lu, S., Li, Q., Wei, H., Chang, M. J., Tunlaya-Anukit, S., Kim, H., . . . Chiang, V. L. (2013). Ptr-miR397a is a negative regulator of laccase genes affecting lignin content in Populus trichocarpa. *Proc Natl Acad Sci U S A, 110*(26), 10848-10853. doi:10.1073/pnas.1308936110

Lv, Q., Cheng, R., & Shi, T. L. (2014). Regulatory network rewiring for secondary metabolism in Arabidopsis thaliana under various conditions. *BMC Plant Biology, 14*. doi:Artn 180

10.1186/1471-2229-14-180

Mahmood, K., Xu, Z., El-Kereamy, A., Casaretto, J. A., & Rothstein, S. J. (2016). The Arabidopsis Transcription Factor ANAC032 Represses Anthocyanin Biosynthesis in Response to High Sucrose and Oxidative and Abiotic Stresses. *Front Plant Sci, 7*, 1548. doi:10.3389/fpls.2016.01548

Marbach, D., Costello, J. C., Kuffner, R., Vega, N. M., Prill, R. J., Camacho, D. M., . . . Stolovitzky, G. (2012). Wisdom of crowds for robust gene network inference. *Nat Methods, 9*(8), 796-804. doi:10.1038/nmeth.2016

Margolin, A. A., Nemenman, I., Basso, K., Wiggins, C., Stolovitzky, G., Dalla Favera, R., & Califano, A. (2006). ARACNE: an algorithm for the reconstruction of gene regulatory networks in a mammalian cellular context. *BMC Bioinformatics, 7 Suppl 1*, S7. doi:10.1186/1471-2105-7-S1-S7

Martens, S., Teeri, T., & Forkmann, G. (2002). Heterologous expression of dihydroflavonol 4-reductases from various plants. *Febs Letters, 531*(3), 453-458. doi:Pii S0014-5793(02)03583-4
Doi 10.1016/S0014-5793(02)03583-4
Masui, S., Ohtsuka, S., Yagi, R., Takahashi, K., Ko, M. S. H., & Niwa, H. (2008). Rex1/Zfp42 is dispensable for pluripotency in mouse ES cells. *Bmc Developmental Biology, 8*. doi:Artn 45
10.1186/1471-213x-8-45
Miles, D. C., de Vries, N. A., Gisler, S., Lieftink, C., Akhtar, W., Gogola, E., . . . van Lohuizen, M. (2017). TRIM28 is an Epigenetic Barrier to Induced Pluripotent Stem Cell Reprogramming. *Stem Cells, 35*(1), 147-157. doi:10.1002/stem.2453
Mitchell-Olds, T. (2010). Complex-trait analysis in plants. *Genome Biology, 11*(4), 113. doi:10.1186/gb-2010-11-4-113
Mitsuda, N., & Ohme-Takagi, M. (2008). NAC transcription factors NST1 and NST3 regulate pod shattering in a partially redundant manner by promoting secondary wall formation after the establishment of tissue identity. *Plant Journal, 56*(5), 768-778. doi:10.1111/j.1365-313X.2008.03633.x
Morey, L., Santanach, A., & Di Croce, L. (2015). Pluripotency and Epigenetic Factors in Mouse Embryonic Stem Cell Fate Regulation. *Molecular and Cellular Biology, 35*(16), 2716-2728. doi:10.1128/Mcb.00266-15
MURPHY, S. M. (1999). Modelling gene expression data using dynamic Bayesian networks. *Technical report, Computer Science Division, University of California, Berkeley, CA*.
Niakan, K. K., Ji, H. K., Maehr, R., Vokes, S. A., Rodolfa, K. T., Sherwood, R. I., . . . Eggan, K. (2010). Sox17 promotes differentiation in mouse embryonic stem cells by directly regulating extraembryonic gene expression and indirectly antagonizing self-renewal. *Genes & Development, 24*(3), 312-326. doi:10.1101/gad.1833510
Nishitani, K., & Demura, T. (2015). An Emerging View of Plant Cell Walls as an Apoplastic Intelligent System. *Plant and Cell Physiology, 56*(2), 177-179. doi:10.1093/pcp/pcv001
Ohashi-Ito, K., Oda, Y., & Fukuda, H. (2010). Arabidopsis VASCULAR-RELATED NAC-DOMAIN6 Directly Regulates the Genes That Govern Programmed Cell Death and Secondary Wall Formation during Xylem Differentiation. *Plant Cell, 22*(10), 3461-3473. doi:10.1105/tpc.110.075036
Papp, B., & Plath, K. (2012). Pluripotency re-centered around Esrrb. *Embo Journal, 31*(22), 4255-4257. doi:10.1038/emboj.2012.285
Persson, S., Wei, H., Milne, J., Page, G. P., & Somerville, C. R. (2005). Identification of genes required for cellulose synthesis by regression analysis of public microarray data sets. *Proc Natl Acad Sci U S A, 102*(24), 8633-8638. doi:10.1073/pnas.0503392102
Poovaiah, C. R., Nageswara-Rao, M., Soneji, J. R., Baxter, H. L., & Stewart, C. N. (2014). Altered lignin biosynthesis using biotechnology to improve lignocellulosic biofuel feedstocks. *Plant Biotechnology Journal, 12*(9), 1163-1173. doi:10.1111/pbi.12225

Rahantamalala, A., Rech, P., Martinez, Y., Chaubet-Gigot, N., Grima-Pettenati, J., & Pacquit, V. (2010). Coordinated transcriptional regulation of two key genes in the lignin branch pathway - CAD and CCR - is mediated through MYB- binding sites. *BMC Plant Biology, 10*. doi:Artn 130
10.1186/1471-2229-10-130

Reshef, D. N., Reshef, Y. A., Finucane, H. K., Grossman, S. R., McVean, G., Turnbaugh, P. J., . . . Sabeti, P. C. (2011). Detecting Novel Associations in Large Data Sets. *Science, 334*(6062), 1518-1524. doi:10.1126/science.1205438

Rodda, D. J., Chew, J. L., Lim, L. H., Loh, Y. H., Wang, B., Ng, H. H., & Robson, P. (2005). Transcriptional regulation of nanog by OCT4 and SOX2. *Journal of Biological Chemistry, 280*(26), 24731-24737. doi:10.1074/jbc.M502573200

Ruiz-Perez, M. V., Henley, A. B., & Arsenian-Henriksson, M. (2017). The MYCN Protein in Health and Disease. *Genes (Basel), 8*(4). doi:10.3390/genes8040113

Ruklisa, D., Brazma, A., & Viksna, J. (2005). Reconstruction of gene regulatory networks under the finite state linear model. *Genome Inform, 16*(2), 225-236.

Schäfer, J., & Strimmer, K. (2005). *Learning large-scale graphical Gaussian models from genomic data*. Aveiro, PT, August 2004.: The American Institute of Physics.

Schneidman, E., Still, S., Berry, M. J., & Bialek, W. (2003). Network information and connected correlations. *Physical Review Letters, 91*(23). doi:ARTN 238701
10.1103/PhysRevLett.91.238701

Seo, P. J., & Park, C. M. (2010). MYB96-mediated abscisic acid signals induce pathogen resistance response by promoting salicylic acid biosynthesis in Arabidopsis. *New Phytol, 186*(2), 471-483. doi:10.1111/j.1469-8137.2010.03183.x

Sham, P. C., & Purcell, S. M. (2014). Statistical power and significance testing in large-scale genetic studies. *Nat Rev Genet, 15*(5), 335-346. doi:10.1038/nrg3706

Sharov, A. A., Masui, S., Sharova, L. V., Piao, Y., Aiba, K., Matoba, R., . . . Ko, M. S. (2008). Identification of Pou5f1, Sox2, and Nanog downstream target genes with statistical confidence by applying a novel algorithm to time course microarray and genome-wide chromatin immunoprecipitation data. *BMC Genomics, 9*, 269. doi:10.1186/1471-2164-9-269

Shu, J., Wu, C., Wu, Y. T., Li, Z. Y., Shao, S. D., Zhao, W. H., . . . Deng, H. K. (2015). Induction of Pluripotency in Mouse Somatic Cells with Lineage Specifiers (vol 153, pg 963, 2013). *Cell, 161*(5), 1229-1229. doi:10.1016/j.cell.2015.05.020

Shuai, B., Reynaga-Pena, C. G., & Springer, P. S. (2002a). The lateral organ boundaries gene defines a novel, plant-specific gene family. *Plant Physiol, 129*(2), 747-761. doi:10.1104/pp.010926

Singh, K. B. (1998). Transcriptional Regulation in Plants: The Importance of Combinatorial Control. *Plant Physiology, 118*, 1111-1120.

Singh, S. K., Kagalwala, M. N., Parker-Thornburg, J., Adams, H., & Majumder, S. (2008). REST maintains self-renewal and pluripotency of embryonic stem cells. *Nature, 453*(7192), 223-U211. doi:10.1038/nature06863

Springob, K., Nakajima, J., Yamazaki, M., & Saito, K. (2003). Recent advances in the biosynthesis and accumulation of anthocyanins. *Nat Prod Rep, 20*(3), 288-303.

Sweetlove, L. J., Last, R. L., & Fernie, A. R. (2003). Predictive metabolic engineering: A goal for systems biology. *Plant Physiology, 132*(2), 420-425. doi:10.1104/pp.103.022004

Tanaka, Y., Sasaki, N., & Ohmiya, A. (2008). Biosynthesis of plant pigments: anthocyanins, betalains and carotenoids. *Plant Journal, 54*(4), 733-749. doi:10.1111/j.1365-313X.2008.03447.x

Thevenin, J., Pollet, B., Letarnec, B., Saulnier, L., Gissot, L., Maia-Grondard, A., . . . Jouanin, L. (2011). The simultaneous repression of CCR and CAD, two enzymes of the lignin biosynthetic pathway, results in sterility and dwarfism in Arabidopsis thaliana. *Mol Plant, 4*(1), 70-82. doi:10.1093/mp/ssq045

Vanholme, R., Demedts, B., Morreel, K., Ralph, J., & Boerjan, W. (2010). Lignin biosynthesis and structure. *Plant Physiol, 153*(3), 895-905. doi:10.1104/pp.110.155119

Watkinson, J., Liang, K. C., Wang, X. D., Zheng, T., & Anastassiou, D. (2009). Inference of Regulatory Gene Interactions from Expression Data Using Three-Way Mutual Information. *Challenges of Systems Biology: Community Efforts to Harness Biological Complexity, 1158*, 302-313. doi:10.1111/j.1749-6632.2008.03757.x

Wu, R., Wang, T., McGie, T., Voogd, C., Allan, A. C., Hellens, R. P., & Varkonyi-Gasic, E. (2014). Overexpression of the kiwifruit SVP3 gene affects reproductive development and suppresses anthocyanin biosynthesis in petals, but has no effect on vegetative growth, dormancy, or flowering time. *J Exp Bot, 65*(17), 4985-4995. doi:10.1093/jxb/eru264

Xu, Z. (2016). The Role of Anthocyanins and The GATA Transcription Factors GNC and CGA1 in The Plant Response to Stress.

Yang, C., & Wei, H. (2015). Designing Microarray and RNA-Seq Experiments for Greater Systems Biology Discovery in Modern Plant Genomics. *Molecular Plant, 8*(2), 196-206. doi:https://doi.org/10.1016/j.molp.2014.11.012

Yeung, K. Y., Medvedovic, M., & Bumgarner, R. E. (2004). From co-expression to co-regulation: how many microarray experiments do we need? *Genome Biology, 5*(7). doi:ARTN R48

DOI 10.1186/gb-2004-5-7-r48

Yoon, J., Choi, H., & An, G. (2015). Roles of lignin biosynthesis and regulatory genes in plant development. *J Integr Plant Biol, 57*(11), 902-912. doi:10.1111/jipb.12422

Zhang, Y. Q., Zheng, S., Liu, Z. J., Wang, L. G., & Bi, Y. R. (2011). Both HY5 and HYH are necessary regulators for low temperature-induced anthocyanin accumulation in Arabidopsis seedlings. *Journal of Plant Physiology, 168*(4), 367-374. doi:10.1016/j.jplph.2010.07.025

Zhao, C., Avci, U., Grant, E. H., Haigler, C. H., & Beers, E. P. (2008). XND1, a member of the NAC domain family in Arabidopsis thaliana, negatively regulates lignocellulose synthesis and programmed cell death in xylem. *Plant Journal, 53*(3), 425-436. doi:10.1111/j.1365-313X.2007.03350.x

Zhong, R., Lee, C., Zhou, J., McCarthy, R. L., & Ye, Z. H. (2008). A battery of transcription factors involved in the regulation of secondary cell wall biosynthesis in Arabidopsis. *Plant Cell, 20*(10), 2763-2782. doi:10.1105/tpc.108.061325

Zhong, R., Richardson, E. A., & Ye, Z. H. (2007). The MYB46 transcription factor is a direct target of SND1 and regulates secondary wall biosynthesis in Arabidopsis. *Plant Cell, 19*(9), 2776-2792. doi:10.1105/tpc.107.053678

Zhong, R., & Ye, Z. H. (2012). MYB46 and MYB83 bind to the SMRE sites and directly activate a suite of transcription factors and secondary wall biosynthetic genes. *Plant Cell Physiol, 53*(2), 368-380. doi:10.1093/pcp/pcr185

Zhong, R., & Ye, Z. H. (2015). The Arabidopsis NAC transcription factor NST2 functions together with SND1 and NST1 to regulate secondary wall biosynthesis in fibers of inflorescence stems. *Plant Signal Behav, 10*(2), e989746. doi:10.4161/15592324.2014.989746

Zhou, J., Lee, C., Zhong, R., & Ye, Z. H. (2009). MYB58 and MYB63 are transcriptional activators of the lignin biosynthetic pathway during secondary cell wall formation in Arabidopsis. *Plant Cell, 21*(1), 248-266. doi:10.1105/tpc.108.063321

Zhou, Q., Chipperfield, H., Melton, D. A., & Wong, W. H. (2007). A gene regulatory network in mouse embryonic stem cells. *Proc Natl Acad Sci U S A, 104*(42), 16438-16443. doi:10.1073/pnas.0701014104

Zou, M., & Conzen, S. D. (2005). A new dynamic Bayesian network (DBN) approach for identifying gene regulatory networks from time course microarray data. *Bioinformatics, 21*(1), 71-79. doi:10.1093/bioinformatics/bth463

Chapter 3

TF-mining Pipelines for Identifying Regulatory Genes Controlling a Biological Pathway, Process, or Complex Trait from High-Throughput Gene Expression Data[2]

3.1 Abstract

Online large-scale data analysis platforms to facilitate efficient data mining are becoming increasingly desirable in this era of genomic data explosion. This chapter describes the implementation of a web-based platform, TF-Miner, which is comprised of two data analysis algorithms, TF-Cluster and TF-Finder (TF for transcription factor). The TF-Cluster algorithm includes both a collaborative network construction phase and a network decomposition phase to obtain functionally coordinated clusters of TFs. The collaborative network construction phase of the original TF-Cluster algorithm utilized the Spearman Rank Correlation Coefficient (SRCC). For the web-based TF-Cluster pipeline, the collaborative network construction phase was supplemented with four additional pair-wise association methods: Pearson Product Moment Correlation Coefficient (PPMCC), Weighted Rank Correlation Coefficient (WRCC), Kendall Rank Correlation Coefficient (KRCC) and Maximum Information Coefficient (MIC). The inclusion of these four additional methods facilitated the identification a range of linear and non-linear associations and thereby recognizing coherent clusters of collaborative TFs. Similarly, while the decomposition phase of the original TF-Cluster algorithm utilized only the Triple-link Algorithm, the web-based TF-Cluster pipeline was supplemented with two additional algorithms: Single-Seed Growing Algorithm (SSGA) and Multi-Seed Growing Algorithm (MSGA). In contrast to TF-Cluster, which takes genome-wide expression profiles of all genes and identifies collaborative clusters of TFs, TF-Finder is more focused on the identification of regulatory TFs controlling a specific biological pathway or a process. The original TF-Finder algorithm utilized Adaptive Sparse Canonical Correlation Analysis (ASCCA) with the aid of a user-supplied regulatory TF knowledge base. In the web-based TF-Finder pipeline, the knowledge base requirement was circumvented by using Sparse Partial Least Squares (SPLS) algorithm to predict

---

[2] The material presented in this chapter was submitted to the BMC Genomics journal in 2016. Reviews and comments for improvements have been incorporated in this manuscript and will be resubmitted to the same journal in the near future.

candidate regulatory TFs existing TF knowledge base is not available. Finally, TF-Miner makes these two pipelines accessible to a large number of researchers due to its user-friendly interface and efficient data management portal for both the input datasets and the output results.

3.2 Introduction

Over the last decade, technologies to generate high-throughput genome-wide gene expression data have become less and less expensive and, as a result, researchers around the world have produced a large amount of data. This data deluge has created an urgent need for efficient, user-friendly, web-based analysis pipelines to mine these large datasets and extract biologically relevant information. A web-based gene expression data analysis platform was implemented by integrating two existing algorithms, TF-Cluster (Nie et al., 2011) and TF-Finder (Cui et al., 2010), while adding novel functionalities. TF-Cluster and TF-Finder were previously available only as separate standalone scripts (R, Perl), and a Unix/Linux-based high-performance computational environment was required to use them for analyzing large-scale datasets. Typically, a significant investment of time and effort is needed to configure a high-end computational hardware and install all the prerequisite software libraries, such as multiple R packages, third party Perl libraries, and Eisen's k-means clustering package. For this reason, we hope to facilitate the work of a multitude of researchers by implementing a web-based application for these two software pipelines.

The TF-Cluster and TF-Finder algorithms were implemented as web-based pipelines with a user-friendly web interface and data management portal. With the web-based interface, the users are separated from tedious command line execution details and algorithm details. When a user registers to the TF-Miner online platform, 2GB of storage space is allocated to upload the gene expression datasets for analysis. Once the analysis is completed an email notification is sent to the user to download results. The data management portal provides an efficient mechanism to store the input datasets and output results for up to two months. More importantly, our web-based TF-Mining pipelines platform has been functionally boosted by the addition of several augmentations to the original TF-Cluster and TF-Finder algorithms.

In short, the TF-Cluster was augmented in both the collaborative network construction phase and the decomposition phase. Based on an earlier analysis (Kumari et al., 2012), the efficiencies of different association methods could be contingent upon the data

properties and to a large extent on the biological processes. This knowledge has necessitated applying multiple methods to identify collaborative TFs via the construction of shared so-expression connectivity matrix (SCCM). In this study, we implemented four additional pair-wise association methods; Pearson Product Moment Correlation Coefficient (PPMCC), Weighted Rank Correlation Coefficient (WRCC), Kendall Rank Correlation Coefficient (KRCC) and Maximum Information Coefficient (MIC). For the second step of decomposition of the SCCM, Xiaohui Ji (Ji et al., 2017) developed two additional algorithms, Single-Seed Growing Algorithm (SSGA) and Multi-Seed Growing Algorithm (MSGA). These two novel decomposition algorithms improve the arbitrary nature of the original Triple-link decomposition algorithm that specifically requires three significant edges from a new TF to the TFs within the growing cluster of TFs (Ji et al., 2017).

In contrast to TF-Cluster, The TF-Finder pipeline employs Adaptive Sparse Canonical Correlation Analysis (ASCCA) to identify groups of TFs using a set of pathway genes involved in a biological process as bait or target variables (Cui et al., 2010). The original TF-Finder algorithm required at least a few known regulatory TFs (guide TFs) in the intermediate step to narrow down and eliminate false positives. By adopting this approach, TF-Finder eliminates the arbitrary process of recognizing TFs purely by statistical significance and improves computational efficiency. However, this requirement of knowing at least a few known positive regulators (guide TFs) as an enrichment test, has limited its applicability because some biological processes have no known positive regulators. For this reason, we have augmented the TF-Finder algorithm by incorporating Sparse Partial Least Squares (SPLS) algorithm (Chun, 2008) to computationally predict putative positive TFs that can be used to replace the positive TFs for the enrichment test in the absence of guide TFs.

3.3 Materials and Methods

3.3.1 Microarray Gene Expression Data

The TF-Miner web application presented here is designed to analyze genome-wide microarray or RNA-seq datasets to identify regulatory TFs. To achieve a higher statistical power and better performance in identifying regulatory TFs using TF-Miner, a sufficient number of samples (>30) should be employed (Cui et al., 2010). To increase the sample size, users can combine samples from multiple experiments, but the gene expression

datasets should satisfy the following conditions to maintain coherence and minimize batch effects:

1. Samples used to extract data were from same species and tissue type (e.g. stems or roots).
2. Gene expression data sets were generated from samples exposed to the same environmental conditions and/or experimental treatments (e.g. salt treatment, long day conditions).
3. The high-throughput platform from which the datasets are generated were similar to each other (*e.g.* gene expression datasets were generated using the same microarray technology such as hybridization of Affymetrix 25k ATH1 microarrays).
4. To create a compendium dataset, either time-series or non-time-series data can be utilized. The time-series data with small intervals such as minutes or hours may be more advantageous as it can capture subtle variations. However, data with large intervals such as days can also be used (Ji et al., 2017).

To compare and estimate the performance of the pipelines incorporated into the TF-Miner, multiple data sets from several microarray experiments were downloaded in raw data format (.CEL) from the NCBI GEO (https://www.ncbi.nlm.nih.gov/gds/) repository. The datasets were preprocessed using the robust multi-array analysis (RMA) algorithm available in the Bioconductor R-package (Irizarry et al., 2003). Quality control of datasets was done by a method described previously (Persson et al., 2005). The following microarray datasets were used to evaluate the performance TF-Miner pipelines:

1. Microarray gene expression datasets from *Arabidopsis thaliana* hypocotyledonous stems under short-day conditions were downloaded from the NCBI GEO repository (Accession numbers: GSE 607, GSE6 153, GSE 18985, GSE 2000, GSE 24781, and GSE 5633). These datasets were derived using hybridization of Affymetrix 25k ATH1 microarray technology. The pooled compendium dataset consists of 128 samples.
2. Microarray gene expression datasets from *Arabidopsis thaliana* roots under salt stress condition were downloaded from the NCBI GEO repository (Accession numbers: GSE 7636, GSE 7639, GSE 7641, GSE 7642, GSE 8787, and GSE 5623). These datasets were derived using hybridization of Affymetrix 25k ATH1 microarray technology. The pooled compendium dataset consists of 108 samples.

3.3.2 TF-Miner Web Application Architecture

The TF-Miner web application was implemented using the model-view-controller (MVC) design pattern, a modern methodology in software engineering that separates algorithmic

details from the user while allowing access to the application logic and algorithm through a user-friendly interface. The MVC design pattern divides the web application into three components: internal mechanism (Model), presentation (View) and user interface (Controller). The three-tiered architecture used to implement the MVC design pattern as a web-based application is shown in Figure 3.1. As defined in the presentation logic (1st tier), the web interface collects user input via the client web browser (*e.g.* Internet Explorer, Google Chrome). When the user first directs the web browser to the address of the web server where the web application is hosted (http://sys.bio.mtu.edu/cluster), specifications to display the web interface in the client web browser are sent as Hypertext Markup Language (HTML) and Cascading Style Sheets (CSS) elements. The HTML and CSS elements specify the structure of the information displayed in the browser and functionality to input user information. The CSS specification allows arrangement of the HTML by enforcing general software engineering practices such as modularity and code reusability. After establishing a connection with the client's web browser, a secure communication channel is established after the user submits login information. For a new user, "Registration" web interface is loaded to create the user profile and add the user information to the database on the server. After successful login to the website, the input data can be transferred to the web server following the request response cycle of Hypertext Transfer Protocol (HTTP).

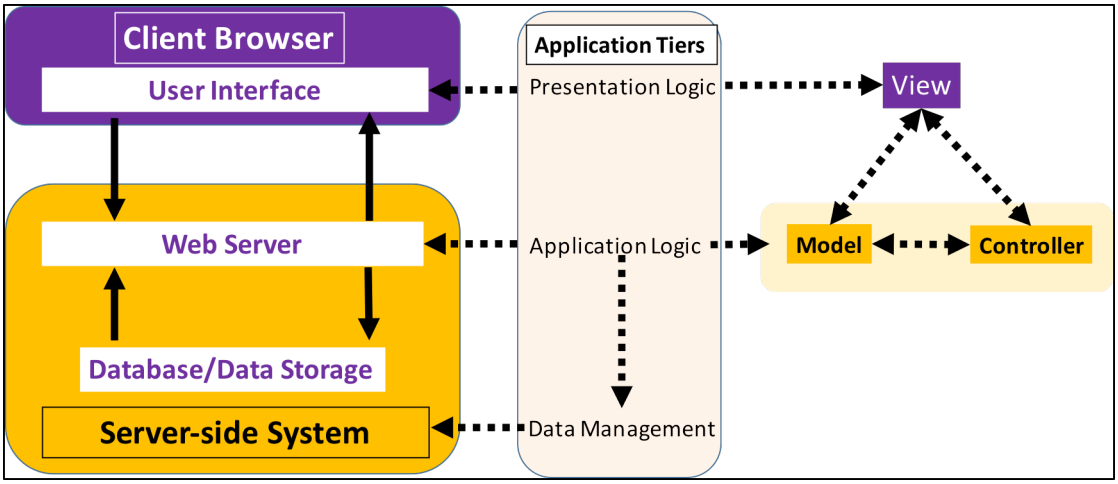


Figure 3.1 Three-tiered dynamic web application framework of model-view-controller (MVC) Architecture. The user interface (View) is defined with instructions from the presentation logic. The application logic defines the Model, and the Controller executes the instructions in the application logic. Data management handles the datasets.

Once the data and required parameters for the desired analysis are submitted to the server, application logic executes the defined model for the requested analysis on the input data. The server-side processing is handled by programming scripts developed in Personal Home Pages (PHP) language and using the open source Apache web server. The TF-Cluster and TF-Finder algorithms were implemented using Perl and R languages. When each data analysis task is completed, the result is sent to the client's web browser as HTML pages, and an electronic message (e-mail) notification is sent to the client's registered e-mail address. This e-mail functionality was implemented with Google's SMTP (Simple Mail Transfer Protocol) facility via a software library called LWP (Library for WWW in Perl) that allows the web server to use a third-party e-mail server (Gmail) to send emails notifications to the user. The Perl package was downloaded from CPAN online Perl package repository. After the analysis is completed, the user has the option to download, delete, or continue to analyze a different dataset. When the user initiates several analysis tasks, each job is encapsulated to run in a separate directory of the user's disk space to allow simultaneous analyses of different data sets. Not only does this data management structure allow one user to submit multiple analysis tasks, but it also allows additional users to submit multiple analysis tasks simultaneously.

3.3.2.1 TF-Miner log-in system

To use the TF-Miner web application, a user must first register through an HTML form via the "Register" link in the web-interface. The username and email address must be unique. Validation scripts written in JavaScript prevent a user from registering if a duplicate record is detected. If the password is forgotten, it can be retrieved through the "forgot password" link in the login web portal by submitting the user email address. Finally, the information collected about a new user is sent to the MySQL database through a PHP script, which executes a SQL (Structured Query Language) command to store the user information in a database.

3.3.2.2 Research data management console

The TF-mining pipelines in this web application typically run from a few minutes to a few hours depending on the size of a user's input dataset. Once a data analysis task is completed, an email notification is sent to the user with a link to the location of the result files on the server. The file manager portal stores input data and output result files under a subdirectory provided by the user (Job ID) when initiating the analysis. The Job ID should follow the convention specified in the web interface; blank spaces and special characters cannot be used. To enforce proper naming of input files and job ID, a dynamic validation system was developed with Asynchronous JavaScript and XML (AJAX)

technology. If the user enters an incompatible name for an input file or Job ID, the interface signals the user with a warning message to prevent the user from running the analysis with incorrect filenames. After the analysis, the data management functionality enables the user to locate and download the result files at any time by logging into the web portal and navigating to the "File Manager" link (Figure 3.2). Each registered user is allocated 2GB of storage space on the web server to store the input datasets and the output files from any of the two analysis pipelines. The stored data can be downloaded and visualized by navigating to the "File Manager" link and opening the subdirectory specified for a given analysis task. If the user does not delete files, the server will automatically remove data files that are older than two months to make room for other users. This function is achieved by triggering a Perl script to scan the dates of available data files whenever a user logs into the system. Every time a given user visits the web page, the Perl script calculates the length of time that each of the user's input and output files has been stored on the server; any files that are older than two months are deleted. If the user has almost reached the quota, a warning message prevents the user from uploading additional datasets until the personal storage space is cleared.

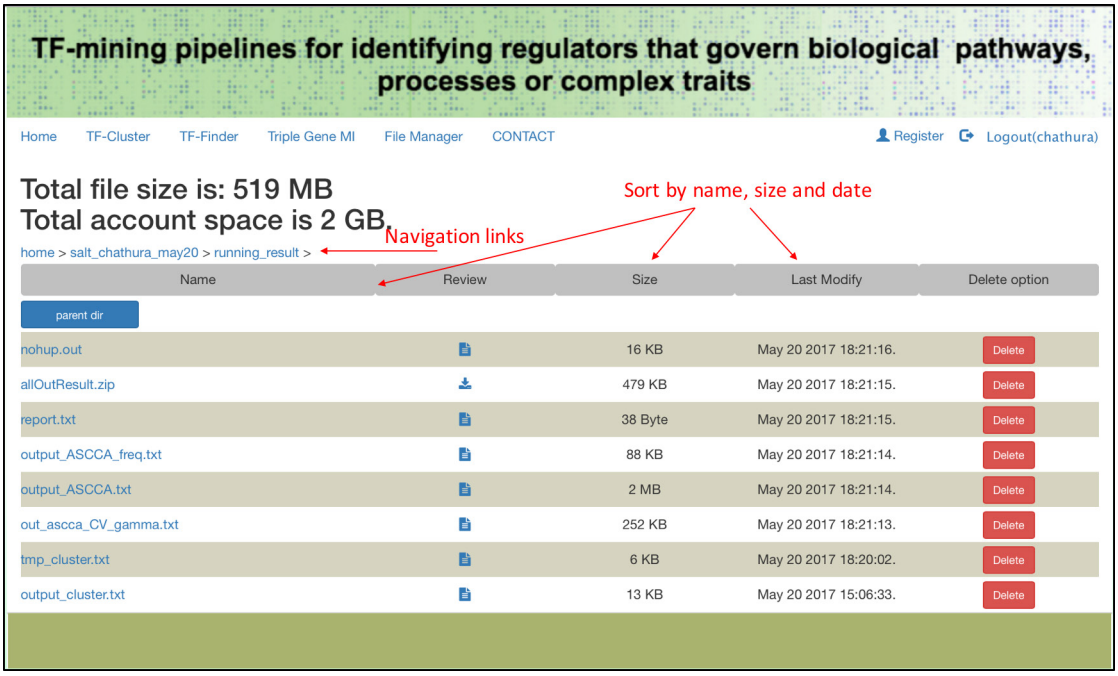


Figure 3.2 The File Manager for the TF-Miner web application. This portal provides access to results from previous analysis job submissions. Exploring the file manager should be done using the navigation links. The buttons in the explorer bar can be used to arrange the files by name, size, or date. This screenshot was obtained from the software created for this dissertation.

3.3.3 TF-Cluster Pipeline

3.3.3.1 TF-Cluster pipeline web interface

The TF-Cluster web interface shown in Figure 3.3 requires a user to upload two files. The first file is a list of all transcription factors (TFs) available for a given species. The second file is a genome-wide gene expression dataset in tab delimited .txt format with the first column containing the locus nomenclature (Meinke & Koornneef, 1997) and subsequent columns containing gene expression samples. The user also has to provide a unique name for the analysis task in the text area labeled as "Job ID". To properly execute the analysis pipeline, the naming should be done according to the specific instructions provided in the web interface. To prevent a user from entering a duplicate Job ID, JavaScript-based validations have been added to the data submission form. If the user provides an invalid Job ID and moves to the next field, an error message indicates this issue near the Job ID field.

The TF-Cluster pipeline first constructs a Shared Co-Expression Connectivity Matrix (SCCM) by calculating correlations between all pairs of TF and non-TF genes using any of the following association methods: Spearman Rank Correlation Coefficient (SRCC), Pearson Product Moment Correlation Coefficient (PPMCC), Weighted Rank Correlation Coefficient (WRCC), Kendall Rank Correlation Coefficient (KRCC) and Maximum Information Coefficient (MIC). Next, a cut-off value specified by the user determines a set of top genes correlated with each TF sorted by p-values; the default is set to pick the top 100 genes. Each element in the SCCM represents the count of common genes between a pair of TFs. The decomposition of the SCCM is implemented with three options: the Triple-Link Algorithm (Nie et al., 2011), SSGA (Ji et al., 2017), or MSGA (Ji et al., 2017). The web interface to select the association method to construct SCCM and input parameters for decomposition algorithms are shown in Figure 3.3.

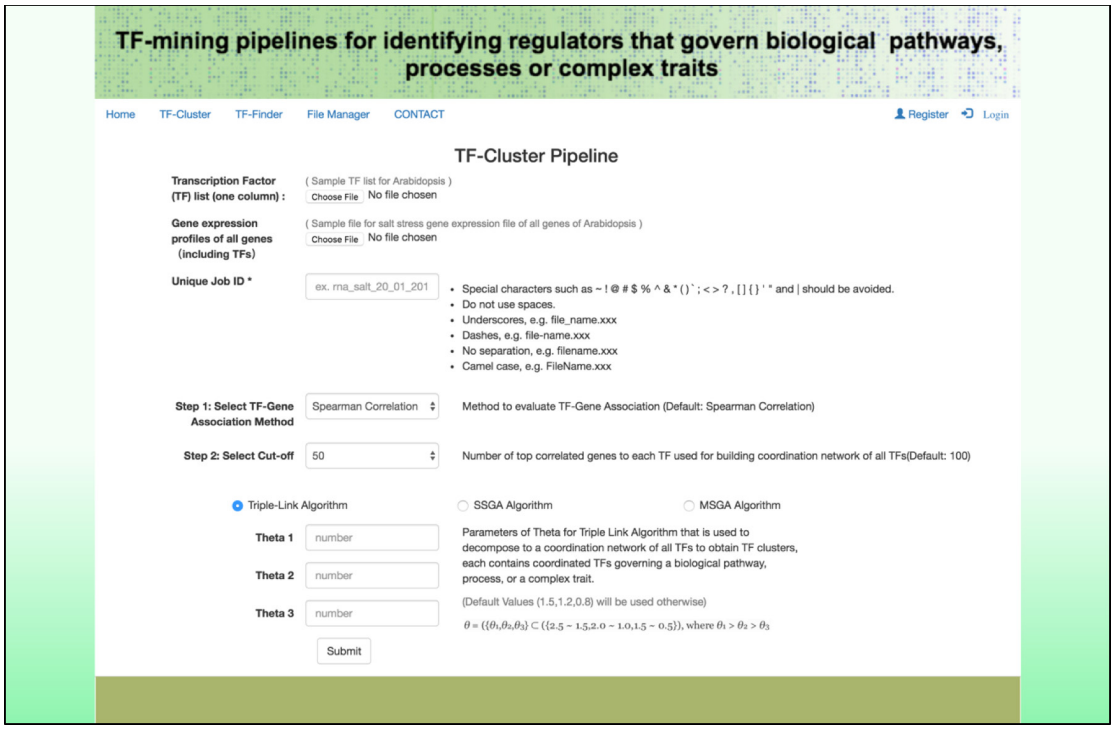


Figure 3.3 TF-Cluster web application interface. A registered user can upload a list of TFs and a genome-wide gene expression dataset. Each analysis task is given a unique Job ID. The user also specifies the TF-gene association method and cut-off for collaborative network construction and the parameters required for decomposition. **Triple Link algorithm parameters**: $\theta = (\theta 1, \theta 2, \theta 3) \subset (0.5{\sim}1.0, 1{\sim}1.5, 1.5{\sim}2.5)$, **SSGA algorithm parameters**; Threshold parameter (θ); default value = 0.6, Average edge weight (w); default value = 1, Where θ =Ratio of authentic connections and theoretical connections w = Connect candidate node to the seed if the average edge weight between candidate node to the nodes in the seed are greater than w. **MSGA algorithm parameters**; Threshold parameter (θ); default value = 0.6, S_{wd} = determine to merge two seed cores based on average weights, T = Number of individual nodes should be added to a merged seed score. This screenshot was obtained from the software created for this dissertation.

3.3.3.2 TF-Cluster algorithm

The TF-Cluster algorithm can be broadly described in two steps: Step 1 is the construction of a Shared Co-Expression Connectivity Matrix (SCCM), and Step 2 is the decomposition of the SCCM into coordinated clusters, where each cluster contains coordinated TFs. The original TF-Cluster algorithm (Nie et al., 2011) is shown Algorithm 3.1.

1: **procedure** BUILDING SCCM
2: Build TF-Gene correlation matrix using one of four gene association methods
3: Find the top one hundred most tightly coexpressed genes to each TF
4: Create a TF-TF SCCM by entering the common genes coexpressed to both TFs in the row and column
5: **end procedure**

1: **procedure** DECOMPOSE-SCCM (TRIPLE-LINK ALGORITHM)
2: For a pair of TFs with maximal connectivity:
3: Add a third TF that has significant connectivity to both TFs satisfying $n_c > \mu + \theta\delta$, where $\theta = ([\theta_1, \theta_2] \subset (2.0,\ 1.5,)$,
4: Add next TF that has at least three significant links to any TFs already in cluster satisfying $n_c > \mu + \theta\delta$, where $\theta = ([\theta_1, \theta_2, \theta_3] \subset ([1.5 \sim 2.5, 1.0 \sim 2.0, 0.5 \sim 1.5])$,
5: Do until no more TFs can be added, the first cluster of TFs is generated.
6: Removal all TFs in the first cluster from SCCM to obtain new SCCM
7: Call Decompose-SCCM with the new SCCM until no more clusters can be decomposed
8: **end procedure**

Algorithm 3.1 TF-Cluster Algorithm with Triple Link Decomposition Method.

To construct the SCCM, correlation analysis between each TF_i (where $i = (1, 2, 3, \ldots n)$ and n is the number of all available TFs) and all other genomic genes is conducted. This correlation analysis can be done with one of the five available methods: Spearman Rank Correlation Coefficient (SRCC), Pearson Product Moment Correlation Coefficient (PPMCC), Weighted Rank Correlation Coefficient (WRCC), Kendall Rank Correlation Coefficient (KRCC) and Maximum Information Coefficient (MIC). The TF-gene relationships indicated by the correlation analysis are sorted according to p-values (lowest to highest), and a list of the top k genes most tightly co-expressed with TF_i is obtained (where k is the cut-off designated by the user; for example, $k = 50, 100, 150, 200$). Similarly, for all n TFs, the top k genes are extracted and kept separately. Next, for all pairs of i,j where $i = (1, 2, 3, \ldots n)$ and $j = (1, 2, 3, \ldots n)$ the number of shared genes (N_C) is calculated as shown in Figure 3.4. N_c represents the shared co-expression between TF_i and TF_j. To construct the SCCM, shared genes among all pairs of TFs are counted and stored.

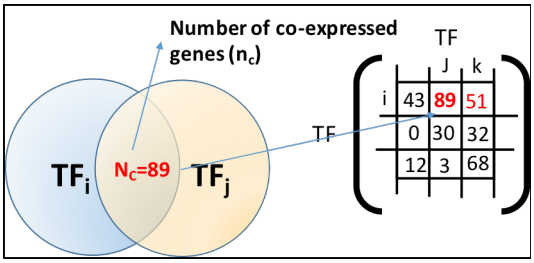


Figure 3.4 Calculation of shared co-expression Connectivity (N_c) between two TFs. N_c represents the coordination of two TFs in the context of other genes. The N_c value of 89 shown above means that for the top 100 genes correlated with TF_i and TF_j, they have 89 genes in common.

The second step of the TF-Cluster algorithm is the decomposition of the SCCM matrix to identify sub-networks of collaborative TFs that function coordinately to regulate a biological process or a complex trait. Currently, three algorithms have been developed to decompose an SCCM collaborative network. The Triple-Link decomposition algorithm was part of the original TF-Cluster algorithm (Nie et al., 2011); Single-Seed Growing Algorithm (SSGA) and Multi-Seed Growing Algorithm (MSGA) (Ji et al., 2017) are new decomposition algorithms that have been added to TF-Cluster. In short, the Triple-Link Algorithm searches through the SCCM and finds the pair of TFs with the maximum value (N_c) in the SCCM matrix (see Figure 3.4). This pair of TFs (TF_i and TF_j) is connected with an edge and serves as the initial seed to grow a coordinated cluster of TFs. A third TF, TF_k, is joined to the original pair of TFs if it has significant connection weights with both TF_i and TF_j. A link is considered as a significant edge if the weight connecting the two TFs is larger than a threshold of $\mu + \theta\delta$, where μ and δ are the mean and the standard deviation of non-zero edge weights contained in SCCM. $\theta = (\theta 1, \theta 2) \subset (0.5{\sim}1.0, 1{\sim}1.5)$. Adding a fourth TF is similar to the previous step, except that now three thresholds have to be met with $\mu + \theta\delta$, where $\theta = (\theta 1, \theta 2, \theta 3) \subset (0.5{\sim}1.0, 1{\sim}1.5, 1.5{\sim}2.5)$. The range of values was chosen empirically based on optimal results obtained by analyzing several datasets. The web-based pipeline provides functionality for users to modify the parameters. $\theta 1$ is the least and $\theta 3$ is the most stringent parameter when adding a node to a cluster. After the third node, any additional TF is added to a cluster if any of the three edges satisfy the requirement; this process continues until all of the TFs are evaluated as candidates. Two additional algorithms, SSGA and MSGA, were developed for the decomposition of the SCCM collaborative network (Ji et al., 2017) to achieve more efficient and accurate results. SSGA and MSGA differ from Triple Link in that they employ a more flexible network-growing approach. This complete TF-Cluster pipeline is shown in Figure 3.5.

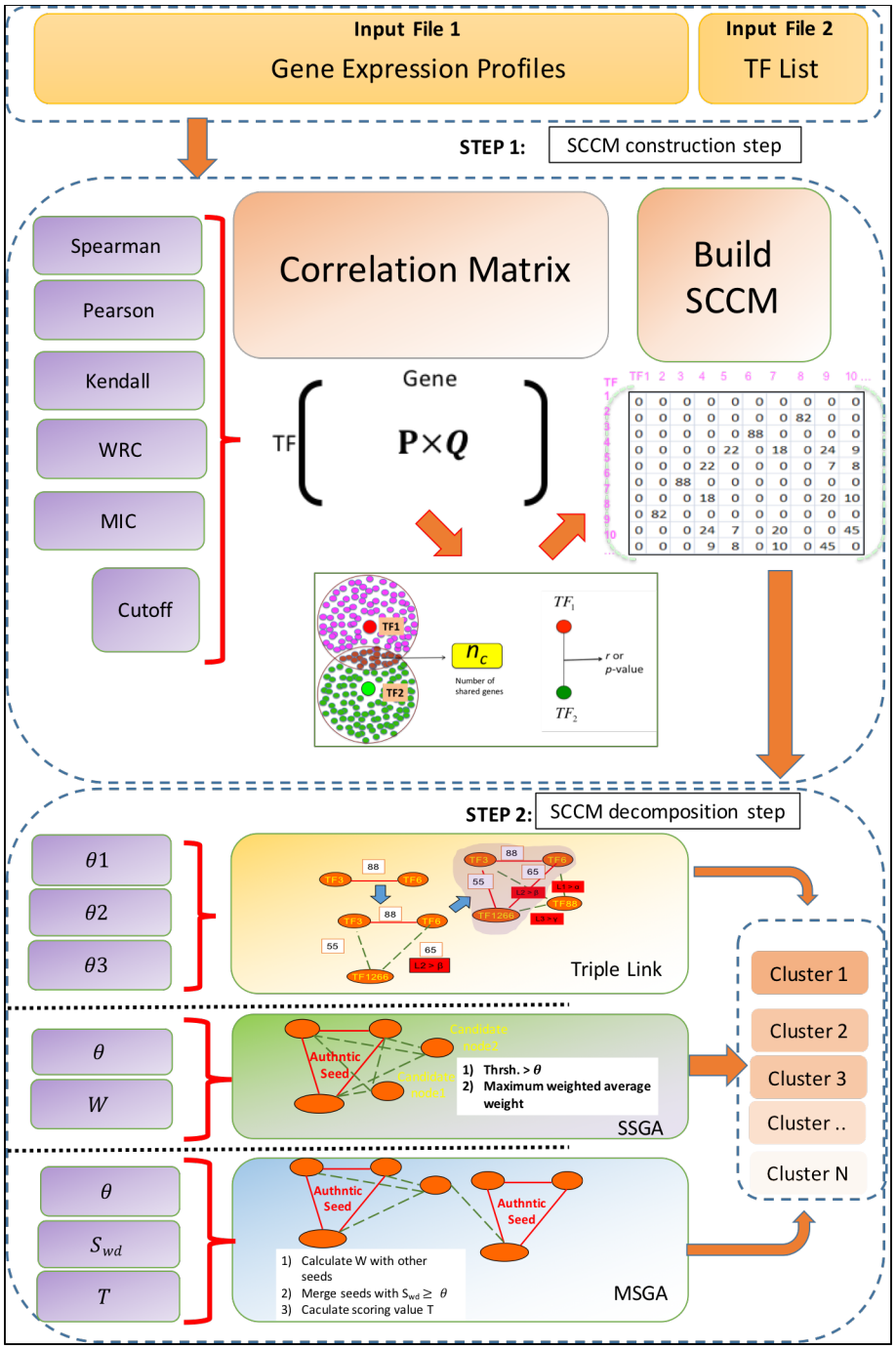


Figure 3.5 A flowchart illustrating the workflows of TF-Cluster. Step 1 shows the construction of the Shared Co-Expression Connectivity Matrix (SCCM), which is the matrix form of the coordination network of all transcription factors (TFs). Step 2 shows the three decomposition methods and parameters for constructing collaborative clusters of TFs.

3.3.3.3 Pair-wise association methods for SCCM construction

The strength of statistical methods in identifying associations between genes using expression data depends on data properties and the biological process. Complex and noisy nature of gene expression data dataset requires multiple methods of analysis, which can capture the varying degree of linear and non-linear associations. The inclusion of four additional association methods to TF-Cluster, facilitated more robust usability of the TF-Cluster pipeline across many different datasets. In a previous publication (Kumari et al., 2012), an extensive comparison was carried out to show the efficiencies in identifying pair-wise associations accross eight association methods; including, Spearman Rank Correlation Coefficient (SRCC), Pearson Product Moment Correlation Coefficient (PPMCC), Weighted Rank Correlation Coefficient (WRCC), Kendall Rank Correlation Coefficient (KRCC). These four pair-wise association methods were incorporated to the web-based TF-Cluster along with Maximum Information Coefficient (MIC), which is a novel measure that can be used for modeling the relationship between two variables including both linear and nonlinear bivariate relationships (Reshef et al., 2011). Because MIC is mutual information based, it can measure associations equally well with both linear and non-linear relationships, including linear, cubic, exponential, monotonic, parabolic, and sinusoidal relationships. This procedure allows MIC method to associate genes with various types of relationships including those that can hardly be distinguished by linear methods. MIC is based on the concept that if a bivariate association exists, a grid can be drawn in a way that partitions the data points of a scatterplot to encapsulate the relationship. Intuitively, the idea is to explore all grid resolutions up to a maximal grid level that captures the highest mutual information associating two variables. Depending on the number of samples, a finite set N of ordered pairs are partitioned into x bins by x-values and y bins by y-values. These bins creates an x-by-y grid. D|G is the notation used to specify the distribution of points D given the grid G.

Definition 1: For a dataset (D) of 2-dimensions where $D \subset \mathbb{R}^2$ and x, y be

$$I(D, x, y) = maxI(D|G)$$

where G is the 2-dimensional bins with x columns and y rows.

Definition 2: The characteristic matrix (M) of set D data is:

$$M(D)_{x,y} = \frac{I^*(D, x, y)}{log\ min[x, y]}$$

Definition 3: Maximum Information Coefficient (MIC):

$$MIC(D) = \max_{XY<B(n)} M(D)_{X,Y} = \max_{XY<B(n)} \frac{I(D,X,Y)}{log(min\, X,Y)}$$

where, $B(n) = n^{\alpha}$

Statistical significance of the associations captured by MIC is determined as follows. If the null hypothesis is that the variables X and Y are statistically independent, the p-value for a given MIC score for a pair-wise association is determined using permutations of the dataset to generate $1/\alpha - 1$ surrogate datasets for $\alpha = 0.05$.

3.3.4 TF-Finder Pipeline

3.3.4.1 TF-Finder web interface

In this web-based pipeline, the TF-Finder algorithm has been incorporated into a simple, user-friendly interface (see Figure 3.6). A user is required to submit gene expression profiles of all transcription factors (TFs), pathway genes (PWGs) from a canonical or a non-canonical pathway and known positive TFs (guide TFs) of the biological pathway (canonical or non-canonical) or complex trait being analyzed(For option 1). The data file must be in the tab delimited ".txt" format with the first column containing the locus nomenclature and subsequent columns containing gene expression samples. The web interface provides three options for the enrichment test. If the guide TFs are not available a user can select option 2 (use SPLS algorithm to predict guide TFs) or option 3 (execute the TF-Finder without the enrichment test).

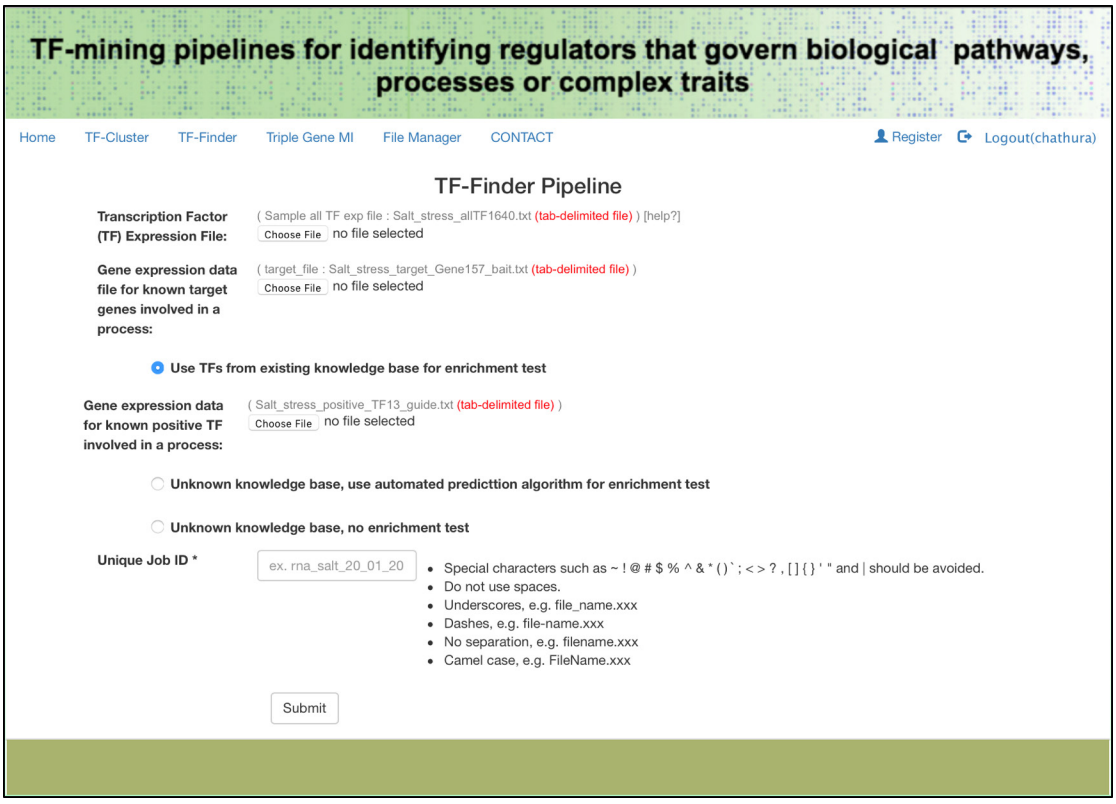


Figure 3.6 TF-Finder web application interface. A unique Job ID must be given to avoid overwriting existing files by following the provided naming convention in the web interface. A user can select any of the three options available for the enrichment test. This screenshot was obtained from the software created for this dissertation.

The enrichment test restricts false positive TFs from entering the predicted TFs set by cross checking TFs obtained by the Adaptive Sparse Canonical Correlation Analysis (ASCCA) with a set of known positive TFs (guide TFs) supplied by the user (as an input file). However, this requirement limits the application of the TF-Finder to situations in which at least a few TFs of the particular biological pathway are known. The new TF-Finder implementation addresses this issue by utilizing an additional algorithm, Sparse Partial Least Squares (SPLS) (Chun, 2008) to predict TF and use these predicted TFs in place of known positive TFs (guide TFs). For a user who does not want to use either known positive TFs or SPLS predicted TFs in the enrichment test, a third option is included to by-pass this enrichment step altogether and keep all TF sets generated by the ASCCA algorithm.

3.3.4.2 TF-Finder algorithm

The original TF-finder (Cui et al., 2010) is shown in Algorithm 3.2. The following sections introduce the Adaptive Sparse Canonical Correlation Analysis (ASCCA) and enrichment test along with modifications added to the web-based TF-Finder pipeline. As shown in Figure 3.7, the TF-Finder pipeline accepts three input files: (1) The target genes file contains expression data for genes that are known to be involved in a biological process or pathway(canonical or non-canonical), (2) Expression data for positive TFs (guide TFs) known to participate in this process, and (3) Expression data for all other TFs. The known positive TFs are those that have been experimentally validated for involvement in regulating the biological process being studied or of interest. The algorithm can identify a set of candidate positive TFs from all other TFs along with the known positive TFs.

1: **procedure** TF-FINDER ALGORITHM
2: **Input 1:** 1. A set of target genes involved in a biological process of interest.
3: **Input 2:** A set of differentially expressed TFs or all TFs across the genome (N)
4: **Input 3:** A set of positive TFs known to be involved in the above biological process (N_pos)
5: Performing Eisens K-means clustering to generate clusters of various sizes between $4 \sim 25$
6: **procedure** PERFORMING ASCCA
7: **for each** cluster: **do**
8: set cluster as Y
9: **end for**
10: candidate $TFs(NASCCA) \leftarrow ASCCA(Y, X)$
11: **if** enrichment test passed **then**
12: $size\_of\_hooked\_TF\_sets \leftarrow size\_of\_hooked\_TF\_sets + 1$
13: **else**
14: discard the current cluster
15: **end if**
16: Sort the baited TFs by the frequency in decreasing order
17: **end procedure**
18: **end procedure**

Algorithm 3.2 TF-Finder algorithm with guide TF-knowledge base.

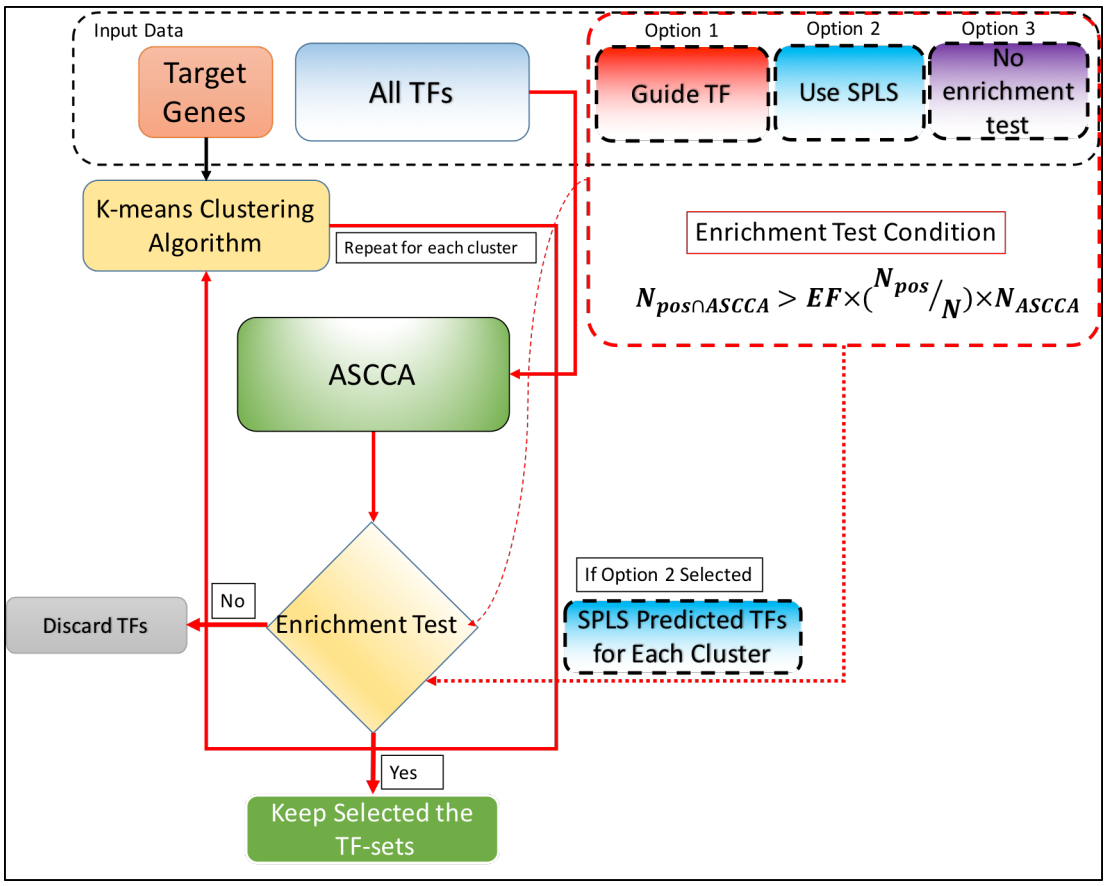


Figure 3.7 TF-Finder pipeline. Three types of data files, namely expression data for target genes, all TFs, and known positive TFs should be submitted. The ASCCA algorithm is performed on each cluster obtained from Eisen's k-means clustering, and the results from ASCCA along with known positive TFs are submitted to the enrichment test. Finally, the pipeline outputs new positive TFs. The enrichment test is supplemented with additional two options; SPLS algorithm or by passing the enrichment test.

3.3.4.2.1 Adaptive sparse canonical correlation analysis (ASCCA)

The first step of the TF-Finder pipeline is clustering the target pathway genes into several groups based on the correlations between gene expression samples. Eisen's k-means clustering algorithm has been implemented with cluster size, k, ranging from 4 to an upper bound. The upper bound is determined as follows. For example, for 100 target genes, with k = 4, results in four groups with each group consisting of 25 genes. The k is gradually increased until the minimum number of genes in a cluster is 4. As shown in Figure 3.7 Each pathway gene cluster is applied to the Adaptive Sparse Canonical Correlation Analysis (ASCCA) as a multivariate response variable (Y), and all the TFs

are supplied as predictor variables (X). Mathematical details of the ASCCA algorithm can be found in the original TF-Finder publication (Cui et al., 2010). Briefly, the ASCCA extracts a set of TFs that coordinately regulates each of the target pathway gene clusters. Importantance of candidate TFs are identified by the frequencies of extraction by the ASCCA algorithm for each target pathway gene cluster. Intuitively, if a particular TF is extracted by in almost all target pathway gene clusters, then that TF should be ranked high by the frequency of extraction for the particular target pathway gene cluster.

3.3.4.2.2 Enrichment test

The TF sets extracted by the ASCCA for each target pathway gene cluster are further evaluated by the enrichment test that takes advantage of any of published studies linking particular set of TFs (guide TFs) to the regulation of the pathway genes of interest (*see* Figure 3.7). This procedure has conceptually similar reasoning to the enrichment test (I. Rivals, Personnaz, Taing, & Potier, 2007). The idea is to develop a test to keep or discard a set of TFs obtained for a particular target pathway gene cluster by evaluating for each extracted TF set as follows:

The following notation is used to introduce this test.

$N = The\ total\ number\ of\ TFs\ in\ X$

$N_{pos} = The\ number\ of\ known\ positive\ TFs$

$N_{ASCCA} = The\ number\ of\ TF\ extracted\ by\ ASCCA\ for\ a\ particular\ cluster$

$N_{pos \cap ASCCA} = The\ number\ of\ TFs\ in\ TFs_{known\ positive} \cap TFs_{Extracted\ by\ ASCCA}$

EF = enrichment factor, a user defined value between 1 - 5

$$N_{pos \cap ASCCA} > EF \times \left(\frac{N_{pos}}{N}\right) \times N_{ASCCA}$$

However, in the event of no known positive TFs for a particular biological pathway of interest, the Sparse Partial Least Squares (SPLS) algorithm (Chun, 2008) is employed to predict a set of candidate positive TFs. The SPLS takes a set of target pathway genes and all TFs as inputs. As shown in Figure 3.7, for each iteration, all the TFs predicted by the SPLS are used as positive TFs (guide TFs) for the enrichment test. If the enrichment test

($N_{pos \cap ASCCA} > EF \times \left(\frac{N_{pos}}{N}\right) \times N_{ASCCA}$) is passed, the set of TFs identified by ASCCA is kept for further steps of the algorithm.

The SPLS algorithm used in the optional enrichment test in the TF-Finder pipeline is briefly described in this section. The concept of SPLS relies on Partial Least Squares (PLS) which is a supervised dimension reduction algorithm. PLS is comparable to Principal Component Analysis (PCA), which reduces the dimension of a dataset to a set of latent components, which can capture the maximum variation in the original dataset. The PLS considers a set of response variables (Y) when obtaining the set of latent components thereby retaining the most relevant components to the response variable. The SPLS imposes a sparsity when obtaining the latent PLS component and limits the number of unimportant variables. The shrinkage of the coefficient in unimportant variables to zero is achieved with a L1 norm penalty constraint in the covariance maximization problem, following the Lasso principle developed by Tibshirani (Tibshirani, 2011). The SPLS predictive model has two primary tuning parameters: $L1$ penalty, η, *is* set between 0 and 1, and K is the number of latent components which can be set between 1 and min (p, (v-1) n/v), where v is the number of folds for cross-validation. In general, lower values of η represent less sparsity (and thus more variables tend to be selected), whereas higher values imply more sparsity. However, the choice of K also affects variable selection in conjunction with η (lower values of K tend to result in fewer chosen variables). To facilitate the choice of K and η, cross validation should be used where the "optimal" K and η are those with the lowest mean squared prediction error.

3.4 Results and Discussion

3.4.1 TF-Cluster Results

3.4.1.1 *Arabidopsis thaliana* roots under salt stress tolerance

This analysis was conducted on a microarray compendium dataset pooled from six salt stress microarray experiments using *Arabidopsis thaliana* roots. The datasets were downloaded from the NCBI GEO repository with accession numbers GSE 7636, GES 7639, GES 7642, GES 8787, GES 5623, and GES 7641. The combined dataset includes 108 samples. The TF-Cluster pipeline was used to analyze the dataset utilizing each of the association methods and keeping the cut-off at the default value of 100. The Triple Link Algorithm was executed with theta values (0.8, 1.2, 1.5). Using the existing

literature, we identified positive TFs known to control a given biological process, pathway, or trait under salt stress conditions. To evaluate the performance of the TF-Cluster pipeline for each association method we counted the number of positive TFs identified in each cluster by each association method. A stacked barplot was used to visualize and compare the efficiency of each association method, as shown in Figure 3.8. Also, the positive TFs identified by each method in top 20 clusters were compared as shown in Table 3.1.

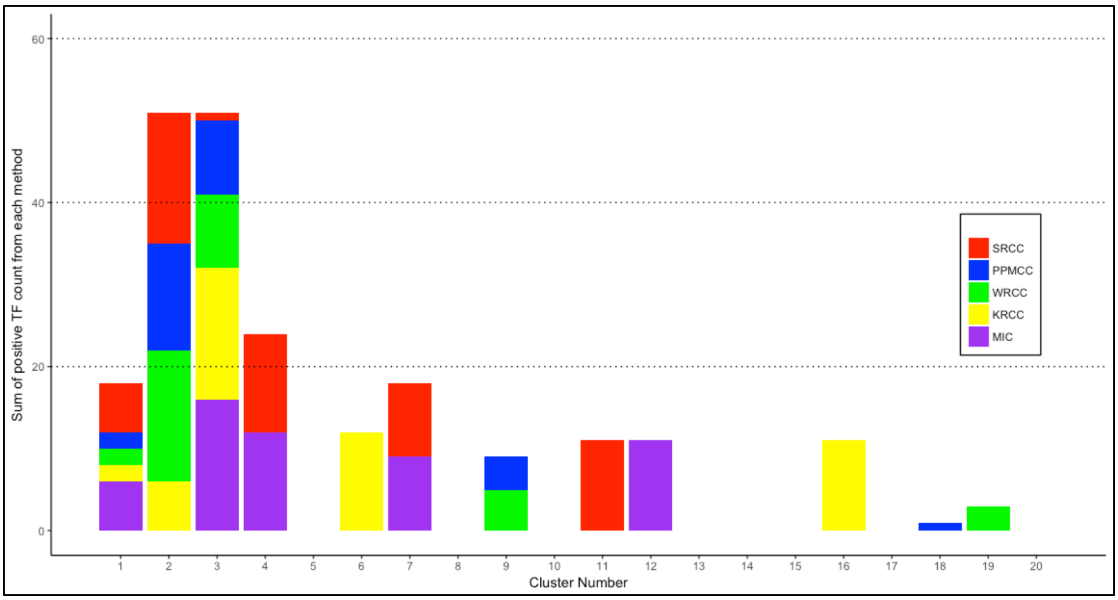


Figure 3.8 Comparison of the number of TFs identified by each association method for TF-Cluster pipeline using *Arabidopsis thaliana* (roots). Positive TFs identified in the top 20 clusters using Spearman Rank Correlation (SRCC), Pearson Product Moment Correlation Coefficient (PMMCC), Weighted Rank Correlation Coefficient (WRCC), Kendall Rank Correlation Coefficient (KRCC), Maximum Information Coefficient (MIC)

Table 3.1 Comparison of the number of positive TFs identified clusters using each association method with the microarray dataset from *Arabidopsis thaliana* (roots).

Cluster No.	Identified Positive TFs	Biological Process controlled by TFs in each cluster	Run Time
Spearman Rank Correlation Coefficient			
Cluster 1	6	Root hair growth	~45min
Cluster 2	16	Root cap growth	
Cluster 4	12	Secondary cell wall growth	
Cluster 7	9	Root cell cycle & growth	
Cluster 11	11	Drought stress in response to ABA	
Pearson Product Moment Correlation Coefficient			
Cluster 1	2	Root hair growth	~30min
Cluster 2	13	Root cap growth	
Cluster 3	9	Secondary cell wall growth	
Weighted Rank Correlation Coefficient			
Cluster 1	2	Root hair growth	~2hr
Cluster 2	16	Root cap growth	
Cluster 3	9	Secondary cell wall growth	
Kendall Rank Correlation Coefficient			
Cluster 1	2	Root hair growth	~3hr
Cluster 2	6	Root cap growth	
Cluster 4	16	Secondary cell wall growth	
Cluster 6	12	Root cell cycle & growth	
Maximum Information Coefficient			
Cluster 1	6	Root hair growth	~5hr
Cluster 3	16	Root cap growth	
Cluster 4	12	Secondary cell wall growth	
Cluster 7	8	Root cell cycle & growth	
Cluster 12	11	Drought stress in response to ABA	

3.4.1.2 *Arabidopsis thaliana* short-day hypocotyledonous stem tissues

This analysis was conducted on a microarray dataset compendium pooled from six functionally related experiments in *Arabidopsis thaliana*. The datasets were downloaded from the NCBI GEO repository with accession numbers: GSE 607, GSE 6153, GSE 18985, GSE 2000, GSE 24781 and GSE 5633. The experiments were conducted on hypocotyledonous stem tissues under short-day conditions known to induce secondary wood formation(Chaffey et al., 2002). The compendium dataset of 128 samples was

analyzed with each of the association method separately, and the cut-off was set to the top 100 genes. The Triple Link Algorithm was executed with theta values (0.8, 1.2, and 1.5). The comparison between clusters are shown in Figure 3.9. The biological processes governed by the positive TFs in top two clusters and the number of positive TFs in each cluster is shown in Table 3.2.

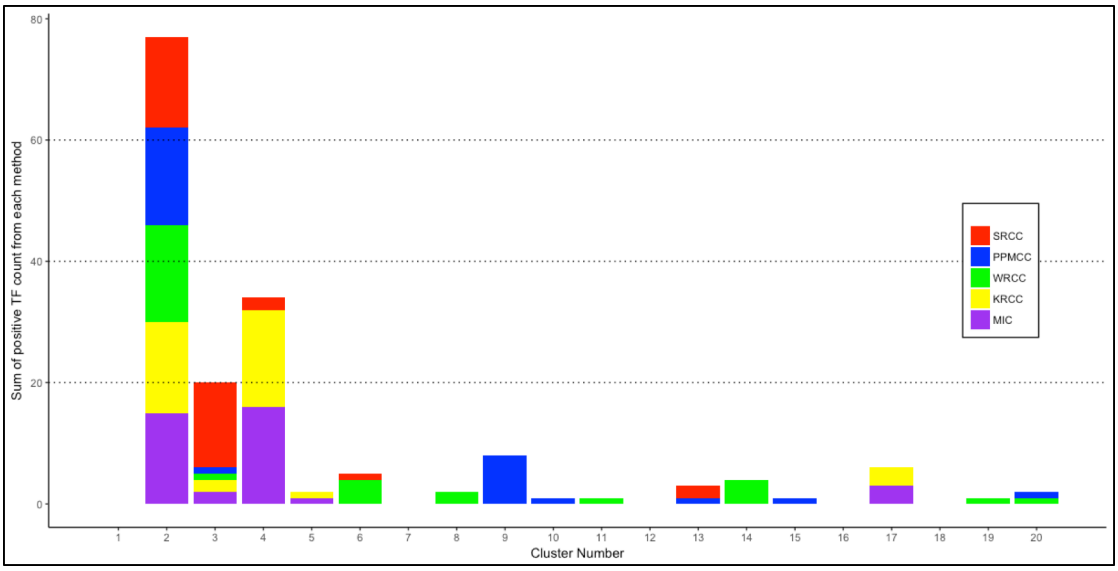


Figure 3.9 Comparison of the number of TFs identified by each association method for TF-Cluster pipeline using *Arabidopsis thaliana* stem tissues. Positive TFs identified in top 20 clusters using Spearman Rank Correlation, Pearson Product Moment Correlation Coefficient (PMMCC), Weighted Rank Correlation Coefficient (WRCC), Kendall Rank Correlation Coefficient (KRCC), Maximum Information Coefficient (MIC)

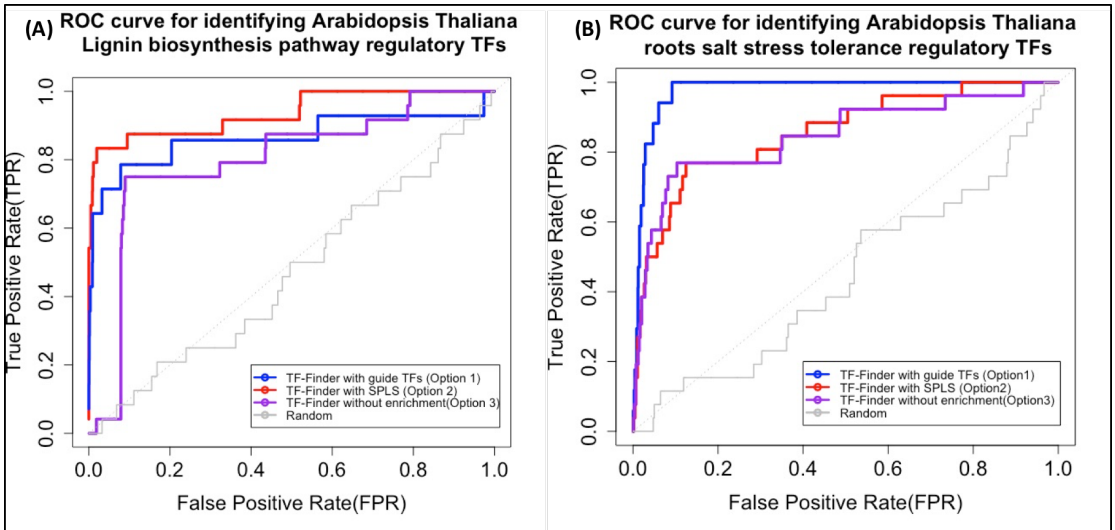
Table 3.2 Comparison of the number of positive TFs identified in clusters using each association method with the microarray dataset from *Arabidopsis thaliana* (stems).

Cluster No.	Identified Positive TFs	Biological Process controlled by TFs in each cluster	Run Time
Spearman Rank Correlation Coefficient			
Cluster 2	15	Secondary cell wall growth and lignin synthesis	~1hr
Cluster 3	12	Vascular patterning and phyllotaxy, Anthocyanin synthesis	
Cluster 4	3	Anthocyanin synthesis	
Pearson Product Moment Correlation Coefficient			
Cluster 2	16	Secondary cell wall growth and lignin synthesis	~40min
Cluster 9	8	Vascular patterning and phyllotaxy, Anthocyanin synthesis	
Weighted Rank Correlation Coefficient			
Cluster 2	16	Secondary cell wall growth and lignin synthesis	~2.5hr
Cluster 6	4	Anthocyanin synthesis	
Kendall Rank Correlation Coefficient			
Cluster 2	15	Secondary cell wall growth and lignin synthesis	~4.2hr
Cluster 4	14	Anthocyanin synthesis	
Maximum Information Coefficient			
Cluster 2	15	Secondary cell wall growth and lignin synthesis	~6hr
Cluster 3	2	Anthocyanin synthesis	
Cluster 4	16	Vascular patterning and phyllotaxy, Anthocyanin synthesis	

3.4.2 TF-Finder Results

TF-Finder was used to identify candidate TFs that are involved in lignocellulosic biosynthesis in stem tissues and salt stress tolerance in roots in *Arabidopsis thaliana* using the two microarray compendium datasets consisting of 128 and 108 samples respectively. ROC curves in Figure 3.10 shows the two datasets have different accuracies

based on the option selected for the enrichment test. Following 3.4.2.1 and 3.4.2.2 sections provide more details about this analysis.



Dataset	TF-Finder with guide TFs (Option1)	TF-Finder with SPLS (Option 2)	TF-Finder without enrichment(Option3)
(A)	0.8594154	0.9295616	0.7373091
(B)	0.9756141	0.8546125	0.8456296

Figure 3.10 ROC curves for the comparison of accuracies of the three enrichment options available for TF-Finder. The blue curve indicates ROC of the original TF-Finder (enrichment using known TFs in Option 1 of the web interface), the red curve indicates ROC with the SPLS algorithm used for enrichment in Option 2 of the web interface. When the enrichment test is ignored (purple curve in Option 3 of the web interface). (A) Identification of Lignin biosynthesis pathway using Arabidopsis thaliana 128 microarray samples generated from stem tissues under long day conditions. (B) Identification of salt stress tolerance regulators using Arabidopsis thaliana 108 microarray samples generated from roots under salt stress conditions.

3.4.2.1 *Arabidopsis thaliana* short-day stem tissues (Lignin biosynthesis)

The accuracy of the TF-Finder pipeline was tested by analyzing a microarray dataset containing 128 samples that were collected from hypocotyledonous stem tissues under short-day condition (See 3.3.1 for more details). A set of 22 pathway genes involved in lignin biosynthesis, were used as target genes. To generate ROC curves to compare the accuracy of identifying novel TFs using Option1 (with enrichment test) with those of other methods, ten guide TFs, out of 20 positive TFs known to regulate the lignin pathway, were employed in the enrichment test. The guide TFs were GATA12, LBD15,

LBD30, MYB85, SND1, NST1, MYB58, NST2, SND2, GRF3 (Kumari et al., 2016). The TF-Finder pipeline identified six positive TFs, namely MYB103, MYB43, MYB46, MYB52, SND3, MYB63 (Deng et al., 2017; Kumari et al., 2016). Using Option 2 (SPLS) for the enrichment test, the TF-Finder pipeline identified all of the TFs obtained using Option 1 as well as additional TFs. Option 3 introduced more noise, and only four positive TFs were identified. ROC curves, which compare the accuracies for identifying TFs using each enrichment test option are shown in Figure 3.10A. For this dataset, the SPLS assisted greatly in comparison to the guide TFs which yielded an AUROC of 0.9295 and not using enrichment test (Option 3) yielded an AUROC of 0.7373.

3.4.2.2 *Arabidopsis thaliana* roots under salt stress (Salt stress response and tolerance)

The accuracy of the TF-Finder pipeline was tested by analyzing the 108-sample microarray dataset from *Arabidopsis thaliana* root tissues (See 3.3.1 for more details). A set of 157 target genes that are known to be involved in salt stress response and tolerance, 1640 *Arabidopsis thaliana* TFs available in Affymetrix ATH1 array and 13 known positive TFs as guides for the enrichment test in Option 1 were used to generate results. When the ASCCA algorithm was used with the 13 TFs (AT1G01520, AT1G35515, AT1G52890, AT2G27300, AT2G30250, AT2G38470, AT2G40950, AT2G47190, AT3G19580, AT3G55980, AT4G28110, AT5G39610, AT5G67450) using Option 1 for the enrichment test, 11 out of 18 salt tolerance TFs were identified by the TF-Finder pipeline. The ROC curves shown in Figure 3.10B shows the comparision of accuracy of option 1 (0.9756) with those of other two options. For this dataset, the existing knowledge base assisted greatly compared to the SPLS algorithm (Option 2) which yielded an AUROC of 0.8546125; Option 3 (no enrichment test) yielded an AUROC of 0.8456296.

3.5 Conclusion

The web-based implementation of TF-Miner discussed in this chapter will facilitate the recognition of major regulatory TFs that govern important biological processes/pathways of interest from large-scale gene expression datasets. The web interface of TF-Mining pipelines enable the users to modify the parameters of two pipelines for extracting more biologically relevant TFs from the datasets. For TF-Cluster pipeline, multiple augmentations were accomplished through the integration of four additional gene association methods for construction of collaborative networks of TFs, in contrast to only one method, namely Spearman Rank Correlation, in the original pipeline. Also, the

decomposing phase was supplemented with additional SSGA and MSGA algorithms. TF-Finder pipeline was equipped with the Sparse Partial Least Squares (SPLS) that is added as an option for the enrichment test when positive TF knowledge base is not available. With the user-friendly and efficient web-based platform along with novel functionalities, the TF-Miner will be an indispensable tool to a multitude of biologists who need to unearth important TFs of interest from gene expression datasets and discover novel biological knowledge.

3.6 Reference List

Chaffey, N., Cholewa, E., Regan, S., & Sundberg, B. (2002). Secondary xylem development in Arabidopsis: a model for wood formation. *Physiol Plant, 114*(4), 594-600.

Chun, H. (2008). *Sparse partial least squares regression for simultaneous dimension reduction and variable selection with applications to high dimensional genomic data.* (Ph D), University of Wisconsin--Madison.

Cui, X., Wang, T., Chen, H. S., Busov, V., & Wei, H. (2010). TF-finder: a software package for identifying transcription factors involved in biological processes using microarray data and existing knowledge base. *BMC Bioinformatics, 11*, 425. doi:10.1186/1471-2105-11-425

Deng, W., Zhang, K., Busov, V., & Wei, H. (2017). Recursive random forest algorithm for constructing multilayered hierarchical gene regulatory networks that govern biological pathways. *PLoS One, 12*(2), e0171532. doi:10.1371/journal.pone.0171532

Irizarry, R. A., Hobbs, B., Collin, F., Beazer-Barclay, Y. D., Antonellis, K. J., Scherf, U., & Speed, T. P. (2003). Exploration, normalization, and summaries of high density oligonucleotide array probe level data. *Biostatistics, 4*(2), 249-264. doi:10.1093/biostatistics/4.2.249

Ji, X., Chen, S., Li, J. C., Deng, W., Wei, Z., & Wei, H. (2017). SSGA and MSGA: two seed-growing algorithms for constructing collaborative subnetworks. *Sci Rep, 7*(1), 1446. doi:10.1038/s41598-017-01556-z

Kumari, S., Deng, W., Gunasekara, C., Chiang, V., Chen, H. S., Ma, H., . . . Wei, H. (2016). Bottom-up GGM algorithm for constructing multilayered hierarchical gene regulatory networks that govern biological pathways or processes. *BMC Bioinformatics, 17*(1), 132. doi:10.1186/s12859-016-0981-1

Kumari, S., Nie, J., Chen, H. S., Ma, H., Stewart, R., Li, X., . . . Wei, H. (2012). Evaluation of gene association methods for coexpression network construction and biological knowledge discovery. *PLoS One, 7*(11), e50411. doi:10.1371/journal.pone.0050411

Meinke, D., & Koornneef, M. (1997). Community standards: A new series of guidelines for plant science - Community standards for Arabidopsis genetics. *Plant Journal, 12*(2), 247-253. doi:DOI 10.1046/j.1365-313X.1997.12020247.x

Nie, J., Stewart, R., Zhang, H., Thomson, J. A., Ruan, F., Cui, X., & Wei, H. (2011). TF-Cluster: a pipeline for identifying functionally coordinated transcription factors via network decomposition of the shared coexpression connectivity matrix (SCCM). *BMC Syst Biol, 5*, 53. doi:10.1186/1752-0509-5-53

Persson, S., Wei, H., Milne, J., Page, G. P., & Somerville, C. R. (2005). Identification of genes required for cellulose synthesis by regression analysis of public microarray data sets. *Proc Natl Acad Sci U S A, 102*(24), 8633-8638. doi:10.1073/pnas.0503392102

Reshef, D. N., Reshef, Y. A., Finucane, H. K., Grossman, S. R., McVean, G., Turnbaugh, P. J., . . . Sabeti, P. C. (2011). Detecting Novel Associations in Large Data Sets. *Science, 334*(6062), 1518-1524. doi:10.1126/science.1205438

Rivals, I., Personnaz, L., Taing, L., & Potier, M. C. (2007). Enrichment or depletion of a GO category within a class of genes: which test? *Bioinformatics, 23*(4), 401-407. doi:10.1093/bioinformatics/btl633

Tibshirani, R. (2011). Regression shrinkage and selection via the lasso: a retrospective. *Journal of the Royal Statistical Society Series B-Statistical Methodology, 73*, 273-282.

Chapter 4
ExactSearch: A Web-based Plant Motif Search Tool[3]

4.1 Abstract

ExactSearch is a web-based software application that utilizes an efficient suffix tree based search algorithm to locate a set of short motif sequences in a set of large target sequences. The target sequences can be regulatory regions associated with specific transcription or translation factors in the proximal promoters or 3′ untranslated regions (3′ UTR) of a set of plant genes of interest. Our algorithm preprocesses the long target sequences into suffix-tree data structures that allow searching for a short motif sequences of length m (m < 20bp) in O(m) time. The algorithm can execute an exhaustive search of 100 motifs against 35,000 target sequences (2 kb in length) in 4.2 minutes. Our web application currently includes a repository of target sequences from proximal promoter regions of 50 plant species, including regions 0.6kb downstream and 0.5kb, 1.5kb, and 2kb upstream. Additionally, the application hosts about 400 available motif sequences from these 50 plant species. When a user submits a search task by uploading/selecting a set of motif sequences and uploading/selecting a set of target sequences, the web portal completes the search operations and sends the result file to the user's email address. The ExactSearch web tool is accessible at this URL: http://sys.bio.mtu.edu/motif.

---

[3] The material presented in this chapter has been published in BMC Plant Methods journal. "Gunasekara, C., Subramanian, A., Avvari, J. V., Li, B., Chen, S., & Wei, H. (2016). ExactSearch: a web-based plant motif search tool. Plant Methods, 12, 26. doi:10.1186/s13007-016-0126-6"

4.2 Introduction

Motifs are short nucleotide sequences located in the promoter region of genes presumed to play a role in regulating gene expression. Transcription factors (TFs) binding to sequence-specific motifs in the promoter regions can to either activate or repress gene expression. These putative regulatory sequence motifs are discovered by searching for overrepresented DNA patterns upstream of functionally related genes employing both experimental and computational approaches (D'Haeseleer, 2006). Identifying known motif sequences in the promoter region of a particular gene can give important clues about regulatory relationships, including predicting target genes for particular transcription factors. Given their regulatory role, biologists often work to locate putative binding motif sequences in the proximal promoters of a set of gene sequences(E. Rivals, Salmela, & Tarhio, 2011). To do this, biologists need to identify appropriate algorithms, download flanking sequences of candidate target genes, and manipulate pattern-matching tasks; this typically occurs in a command line environment foreign to many biologists. Also, searching for numerous motifs in thousands of target sequences can quickly become an overwhelming task, and this approach is especially tedious when degenerate motifs are encountered. Degeneracy is a phenomenon whereby some positions within the DNA sequence must be strictly adhered to and the presence of a designated base is mandated, while other nucleotide positions can be occupied by more than one base, each having an equal probability. As a result, degeneracy increases the number of possible motif sequences that need to be searched. Moreover, to effectively carry out exact pattern matching, several steps are necessary: developing programming scripts, setting up a computational environment, preparing the required input files, and then extracting essential information from outputs. The ExactSearch web application simplifies this entire process by automating repetitive tasks to avoid mistakes. In addition, by using a suffix-tree based algorithm to make the search task faster than is possible with existing tools. The challenge of searching for a large number of short DNA motif sequences in thousands of larger target DNA sequences can be addressed with the efficient implementation of algorithms available in the computer science literature. We adopted the Ukkonen's linear-time suffix-tree construction algorithm (Galil & Ukkonen, 1995) to preprocess each target sequence into suffix-tree data structure, and implemented a suffix tree search algorithm originally proposed by Gusfield (Gusfield & Kao, 1999).

Briefly, locating a DNA motif of length m in a larger target sequence of length n can be done using a naïve approach. This method consists of checking every character of the target sequence between the first (0) and last (n-m) character by sliding a window of length m across the target sequence. If the first nucleotide of the motif sequence matches, then the next character is examined. If the next character does not match, then the

window is shifted one character to the right and compared again. This approach has the time complexity of O (mn) (Lecroq, 2007). The Rabin-Karp (RK) algorithm is another popular string search algorithm which has some similarity to the naïve sliding window approach (Lecroq, 2007). The RK algorithm improves upon the naïve approach by optimizing the sliding window movements across the target sequence. The RK algorithm computes a hash value for the motif sequence pattern and compares it with the hash values generated for each substring in the target sequence. If the hash values match, then individual characters are matched to find the pattern in the target sequence. The RK has the time-complexity of O(m+n) (Lecroq, 2007).

The suffix tree data structure-based search algorithm that we implemented can rapidly search for many motif sequences once the target sequence is preprocessed and saved in the computer's memory. This dynamic programming approach reduced the run time significantly in comparison to the naive and RK algorithms and the existing Regulatory Sequence Analysis Tool (RSAT) (Thomas-Chollier et al., 2011). To make our search algorithm more widely accessible, we developed a user-friendly, web-based application that allows users to carry out search tasks either by uploading their own target sequences or by selecting any of the 50 plant species whose flanking gene sequences that are stored in our database. The user has the option of searching flanking sequences 0.6kb downstream or 1kb, 1.5kb, or 2kb upstream with respect to the coding regions. Also, the web application permits the user to select genes to search for by providing a annotation file for each plant species. We also incorporated 400 known plant motifs into our web application so users can choose to search in custom target sequences or in the proximal sequences of the 50 plant species stored in our sequence database.

4.3 Materials and Methods

4.3.1 Degenerate Motif Sequences

The degeneracy of a motif sequence is a phenomenon whereby some positions within the DNA sequence must be strictly adhered to and the presence of a designated base is mandated, while other nucleotide positions can be occupied by more than one base, each having an equal probability of occupying that position. For example, W (weak) positions have an equal probability of adenine (A) and thymine (T). A weight matrix with a probability weight vector for each position of the sequence can be assigned for each nucleotide as shown in Table 4.1.

Table 4.1 IUPAC notation of Ambiguous characters in nucleotide sequences. The degenerate bases that are not A,T,C,G nucleotides have other equally likely representations.

IUPAC Code	Nucleotides	Name	[p_A,p_C,p_G,p_T]
A	A	Adenine	[1, 0, 0, 0]
C	C	Cytosine	[0, 1, 0, 0]
G	G	Glutamine	[0, 0, 1, 0]
T	T	Thymine	[0, 0, 0, 1]
S	C or G	Strong	[0, ½, ½, 0]
W	A or T	Weak	[½, 0, 0, ½]
R	A or G	PuRine	[½, 0, ½, 0]
Y	C or T	pYrimidine	[0, ½, 0, ½]
M	A or C	aMino group	[½, ½, 0, 0]
K	G or T	Keto group	[0, 0, ½, ½]
B	C or G or T	Not A	[0, ⅓, ⅓, ⅓]
D	A or G or T	Not C	[⅓, 0, ⅓, ⅓]
H	A or C or T	Not G	[⅓, ⅓, 0, ⅓]
V	A or C or G	Not T	[⅓, ⅓, ⅓, 0]
N	A, C, G or T	aNy base	[¼, ¼, ¼, ¼]

For example, a degenerate motif sequence, RTCRYNNNNACGR, takes many forms and each possible sequence combination needed to be checked against the target sequences. For RTCRYNNNNACGR, 4096 possible motifs that can be combined (see Table 4.2). When multiple degenerate motif sequences are present, the number of possible motif sequences increases rapidly.

Table 4.2 Number of possible nucleotides for each base in the motif sequence

R	**T**	**C**	**R**	**Y**	**N**	**N**	**N**	**N**	**A**	**C**	**G**	**R**
2	1	1	2	2	4	4	4	4	1	1	1	2

Input motif sequence file should follow the format shown in Figure 4.1. This format includes the “>” character followed by a unique motif identification name that begins with a lower/upper case letter and followed by any alphanumeric characters. The identifier is followed on the next line by the motif sequence itself.

```
>S000305
TACACTTTTGG
>S000314
CAACA
>S000315
CACCTG
>S00213
............
............
```

Figure 4.1 Sample motif sequence file. A unique motif identification name should be preceded by a ">" character. The motif sequence is given on the next line.

4.3.2 Target DNA Sequences

Target sequences can be uploaded to the ExactSearch web tool in two ways. The first option is to compose target sequences into a .txt file in FASTA format, as shown in Figure 4.2. Each target sequence must start with a header line, which acts as a sequence identifier. This identifier should not contain white spaces and special characters. The sequence identifier must begin with the ">" sign as shown in Figure 4.1. The nucleotide sequence starts on the next line, with each line consisting of a maximum of 80 characters (according to the standard). The target sequences should not contain gaps or any alignment characters. If lower case characters were submitted, these would be mapped to uppercase characters.

```
>AT5G44030
GGTTGTGAAATCATATTTAAAACATTAATAGGTATTTATGTCTAATTTGGGGACAAAATAGTGGAATTCT
TTATCATATCTAGCTAGTTCTTATCGAGTTTGAACTCGGGTTATGATTATGTTACATGCATTGGTCCATA
TAAATCTATGAGCAATCAATATAATTCGAGCATTTTGGTATAACATAATGAGCCAAGTATAACAAAAGTA
TCAAACCTATGCAGGGGAGAAGATGATGAAAAGAAGAGTGTGAGCCAATACAAAGCAGATTTGAGGACAT
GGCTTACAAGTCTTGGGTACAGAGTTTGGGGAGTGATGGGTGCACAATGGAACAGCTTCTCTGGTTGTCC
AGTTCCCAAGAGAACCTTCAAGCTCCCTAACTCCATCTACTATGTCGCCTGATTAAATCTTAT
>AT4G18780
ACGTAGAAACCCATAACTTTAGTATTCTTCAACCCTTACAACTTATCTGAGCAAAATCAGAAGGTCGAAT
TTGATGGATGGTTTTGCTGTATTTGGTCAACGGTTTTATTTGAGACAGTAGACCAGAGGAAACTCAGATG
TGATGATGCAAAGACTGAATTGGTTAAGAGTGTAGATTGATTTGTTCTAACATTGCAAATGTAGAGTAGA
ATTATGCAAAAAACGTTAATGAACAGAGAAGTGATTAAGCAGAAACAAAATTAGAGAAGTGATATTATAT
CTCAAAATTTATTTTTGGTA
```

Figure 4.2 Sample file of target sequences in FASTA format. The target gene identifiers (ids) should be unique and preceded by a ">" character. The nucleotide sequence begins on the next line and continues with 80 nucleotides per line.

The second option is to select target sequences from our repository of proximal promoter regions from 50 plant species. We have included 4 proximal promoters: 0.6kb downstream and 1kb, 1.5kb, 2kb upstream of all genes in the 50 plant species. The presence of certain sequence motifs in these proximal promoter regions is necessary to control gene expression (Maston, Evans, & Green, 2006). We downloaded the sequences from the online repository Phytozome.org (Goodstein et al., 2012). Automated scripts were developed to download the sequence data using the Biomart Perl API (http://www.ensembl.org/info/data/biomart/biomart_perl_api.html).

4.3.3 ExactSearch Algorithm

The ExactSearch algorithm first separates non-degenerate and degenerate motif sequences into separate files by traversing through the input motif file. The non-degenerate motif file is directly sent to the next phase of the algorithm while the degenerate motif file is further processed to generate candidate motif sequences. As shown in the Algorithm 4.1, a lookup table substitutes each of the degenerate characters by applying a recursive algorithm. In short, this recursive algorithm starts with the first character, and, if it is a degenerate nucleotide (R, Y, S, W, K, M, B, D, H, V, N), then the first letter is replaced with appropriate (A, T, C, G) base characters. For example, RCGMK will generate two sequences: ACGMK and GCGMK. At this point, the second nucleotide of each newly generated sequence is considered. Since the second character is not degenerate in this example, the algorithm moves on to the next character, this continues until all combinations of the degenerate motif sequences have been copied to a file. After this step, the generated sequence combinations are sent to the next phase of the ExactSearch algorithm.

```
1: string input ← degenerated motif sequence
2: generate code lookup table = (
        "R" => ["A","G"],
        "Y" => ["C","T"],
        "S" => ["G","C"],
        "W" => ["A","T"],
        "K" => ["G","T"],
        "M" => ["A","C"],
        "B" => ["C","G","T"],
        "D" => ["A","G","T"],
        "H" => ["A","C","T"],
        "V" => ["A","C","G"],
        "N" => ["A","C","G","T"]
   )


3: procedure GENERATE
4:     Input (string)
5:     initialize sequences: array
6:     if find the first degenerated character in the input then
7:         lookup the degenerated characters
8:         generate all combinations and store in the sequences
9:         for each seqence ∈ sequences do
10:            generate(sequence)
11:        end for
12:    else
13:        no degenerated characters
14:        return input
15:    end if
16: end procedure


17: print the list of motifs from output of generate(input sequence)
```

Algorithm 4.1 Generate all possible combinations from the degenerated motif sequence using IUPAC notation for ambiguity nucleotides.

The next step of the ExactSearch algorithm is to preprocess each of the target sequences into suffix-tree data structures. The ExactSearch algorithm employs Ukkonen's suffix-tree construction method (Ukkonen, 1995), for this preprocessing step for each target sequence. Briefly, characters are represented in a tree starting from a root node with edges extending out to new nodes and expanding until a terminating character ($ sign) is reached. Figure 4.3 illustrates how the algorithm builds a suffix tree from a sample target sequence (T) of 11 bases (T= `TAACAGAGTGAC`). Then the search of a short motif sequence (m = `ACAGAG`) using the suffix-tree is demonstrated.

Step 1: Suffixes are generated by removing one character at a time from the left side of the sequence (T), moving from the left to right, while separating each sub-sequence from the remaining sequence. The $ character is added to the right side end target sequence before suffix generation to signify the end of the sequence. The algorithm builds each new suffix from the previous suffix as generalized by the sequence T of length n as shown in Step 1 of Figure 4.3.

Step 2: The suffix tree can be represented in a dictionary format in which, the fixed set of characters that exist in the target sequence are ordered in a predefined pattern. For the nucleotide sequences, we defined the characters in our target sequence in the order shown:

$ > A > C > G > T

The suffixes are then sorted according to this predefined order as shown in Step 2 of Figure 4.3.

Step 3: After sorting, the suffixes are arranged into a tree structure in which the root node diverges into five branches, one for each character ($, A, C, G, T). The suffixes built in the previous step are stored in the tree in such a way that each node contains information about subsequent nodes. The first character of the substring to be added is checked at the first level of the suffix tree; if no match is found, a node with this new character is added immediately under the current node, and the rest of the substring is stored under this newly created node. In this manner, all suffixes are stored in the data structure as shown in Step 3 of Figure 4.3.

Step 4: Once the target sequence is represented in the suffix-tree and stored in the computer's memory, this can repeatedly be used to search for as many motif sequences as needed (Figure 4.3, Step 4). In this example, the stored suffix tree generated from the target sequence, T, is searched for the motif pattern 'ACAGAG'. In traversing the initial portion of this motif, A -> C -> A, the algorithm determines that this sub-pattern be in node 3, thereby determining the starting location of the motif sequence in the target sequence. If the search function successfully maps the motif sequence to a node of the suffix-tree, then that sorted rank of the node (established in Step 2), that is stored at the end of each path (represented by green circles in Figure 4.3, Step 3) is returned. This rank represents the starting location of the motif sequence in the target sequence.

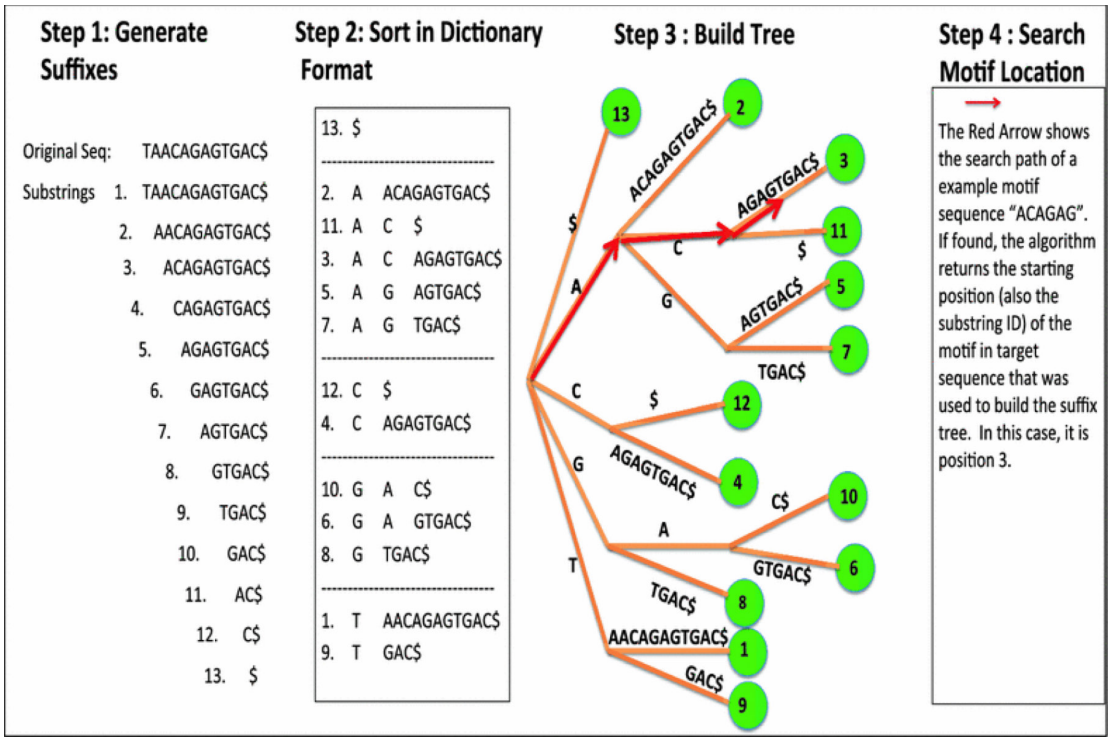
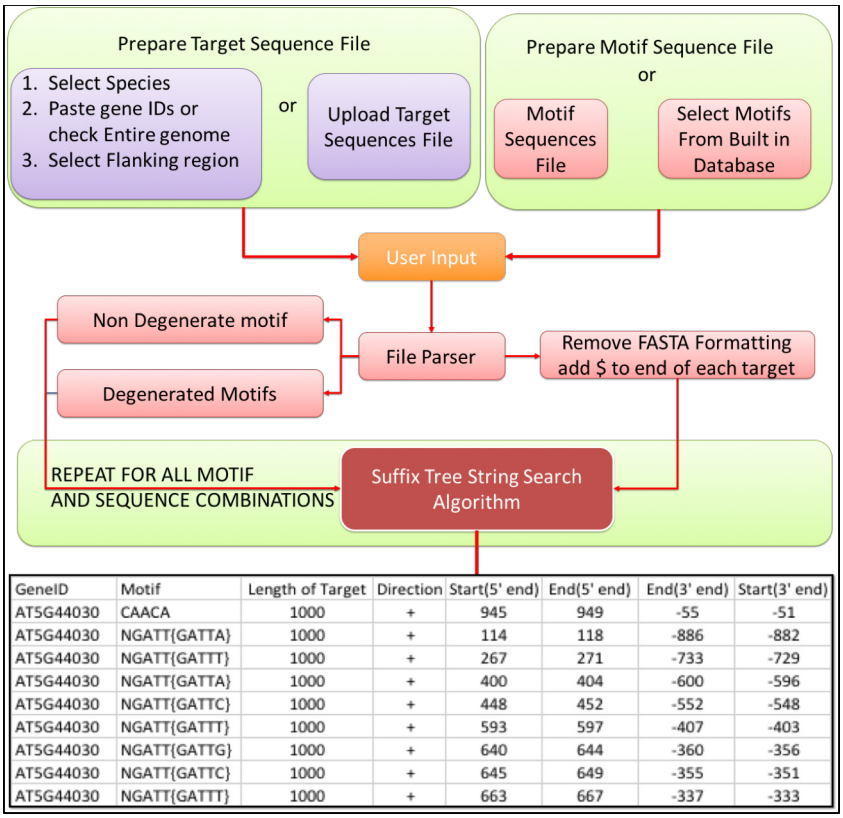


Figure 4.3 An illustration of suffix-tree search algorithm using a sample target sequence (Steps 1-3) and how to perform a motif search (Step 4). A simple target sequence T = (TAACAGAGTGAC) is used for demonstration purposes. The generated suffixes are sorted by the predefined order and represented in the tree structure. This figure was reproduced from the manuscript published in BMC Plant Methods open access journal (Gunasekara et al., 2016). Copyright documentation is attached in Appendix (A.1.1)

4.4 Web-based Implementation

Figure 4.4 shows the overall workflow of the web application. At the center of the web application resides the suffix tree-based string search algorithm compiled into a command line executable program developed in C++. Depending on the requirements, users can either upload a set of target sequences in a txt file or select target sequences from available proximal promoter sequences in the web tool. The proximal promoter sequences from genes of any of the 50 plant species can be chosen from the web interface. Specific genes can be searched by providing the gene IDs of interest or all genes.



GeneID	Motif	Length of Target	Direction	Start(5' end)	End(5' end)	End(3' end)	Start(3' end)
AT5G44030	CAACA	1000	+	945	949	-55	-51
AT5G44030	NGATT{GATTA}	1000	+	114	118	-886	-882
AT5G44030	NGATT{GATTT}	1000	+	267	271	-733	-729
AT5G44030	NGATT{GATTA}	1000	+	400	404	-600	-596
AT5G44030	NGATT{GATTC}	1000	+	448	452	-552	-548
AT5G44030	NGATT{GATTT}	1000	+	593	597	-407	-403
AT5G44030	NGATT{GATTG}	1000	+	640	644	-360	-356
AT5G44030	NGATT{GATTC}	1000	+	645	649	-355	-351
AT5G44030	NGATT{GATTT}	1000	+	663	667	-337	-333

Figure 4.4 The flowchart of ExactSearch algorithm. The input files can be uploaded or selected from the available target or motif sequences on the web interface. The file parser reads the motif sequences; degenerate sequences are converted into all possible motifs with A, T, C, G (non-degenerated) bases. Each target sequence and the converted motif file are then fed to the suffix-tree search algorithm. The suffix-tree search algorithm implemented in a C++ executable file identifies the motif sequences in the target sequence in both forward and reverse strands and saves the locations to a text file.

The web application was developed using the Model-View-Controller (MVC) design pattern implemented in a three-tier architecture. The user interacts with the ExactSearch algorithm through the web interface implemented in Hyper Text Markup Language (HTML), Cascading Style Sheets (CSS) and JavaScript languages. Figure 4.5 shows the web interface to upload or paste in motifs and target sequences. Client-side data validations using JavaScript prevent the user from submitting incorrect data formats to the web server. A user can submit multiple search jobs while another search job is executing, but the filenames must be different to avoid overwriting. When the user clicks the submit button, the algorithm starts executing with the two input files: the target sequence file and the motif sequence file. First, each of the target sequences is preprocessed into suffix tree data structures. Then, the algorithm searches for each of the motif sequences in the preprocessed target sequences in the tree; non-degenerate motifs are searched immediately, but degenerate motifs are further processed to generate all

possible candidate motif sequences before the search is executed. Once all motifs are searched for in each target sequence, the result file is emailed to the user. The web interface shown in Figure 4.6 is used to search for the presence of a set of motif sequences in particular genes of a species genome. The user has the option to restrict the search regarding which genes and which flanking regions to search. When a species name is selected from the drop-down list, the user can download the annotation file for all the genes of that species. After the user identifies a set of genes, the gene ids are copied to the text area provided to upload the gene list. Additionally, a genome-wide search can be done by selecting the appropriate checkbox. The online database of the ExactSearch web application stores known motif sequences that users can select and search for in a user-provided, uploaded sequence file. To date, we have collected approximately 400 known motif sequences across the 50 sequenced plant species.

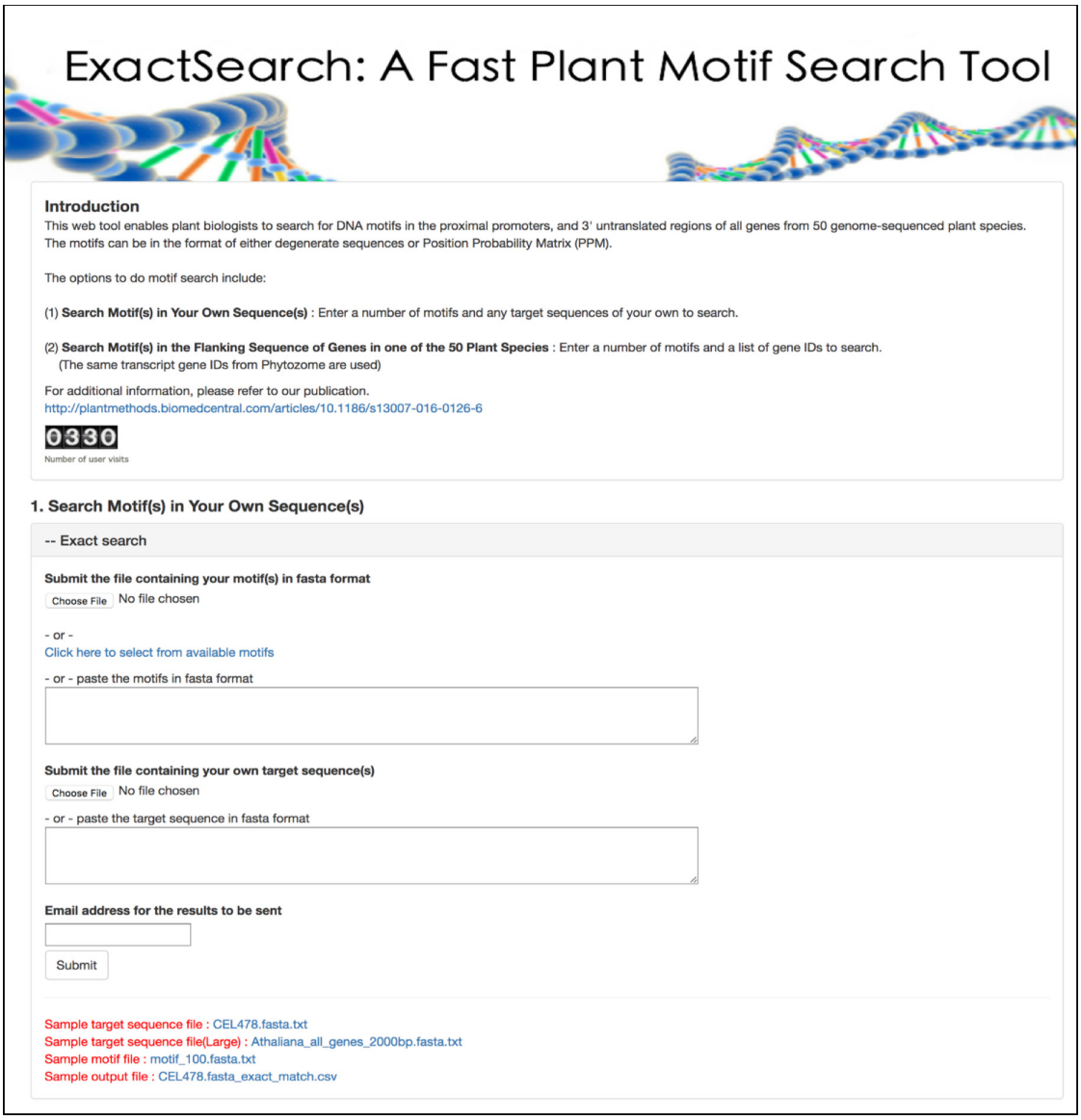


Figure 4.5 Web interface to upload motif and sequence files. The user has the option to upload a file in the designated format or paste into text areas. An email address is required to send the result files to the user once the analysis is complete. This screenshot was obtained from the software created for this dissertation.

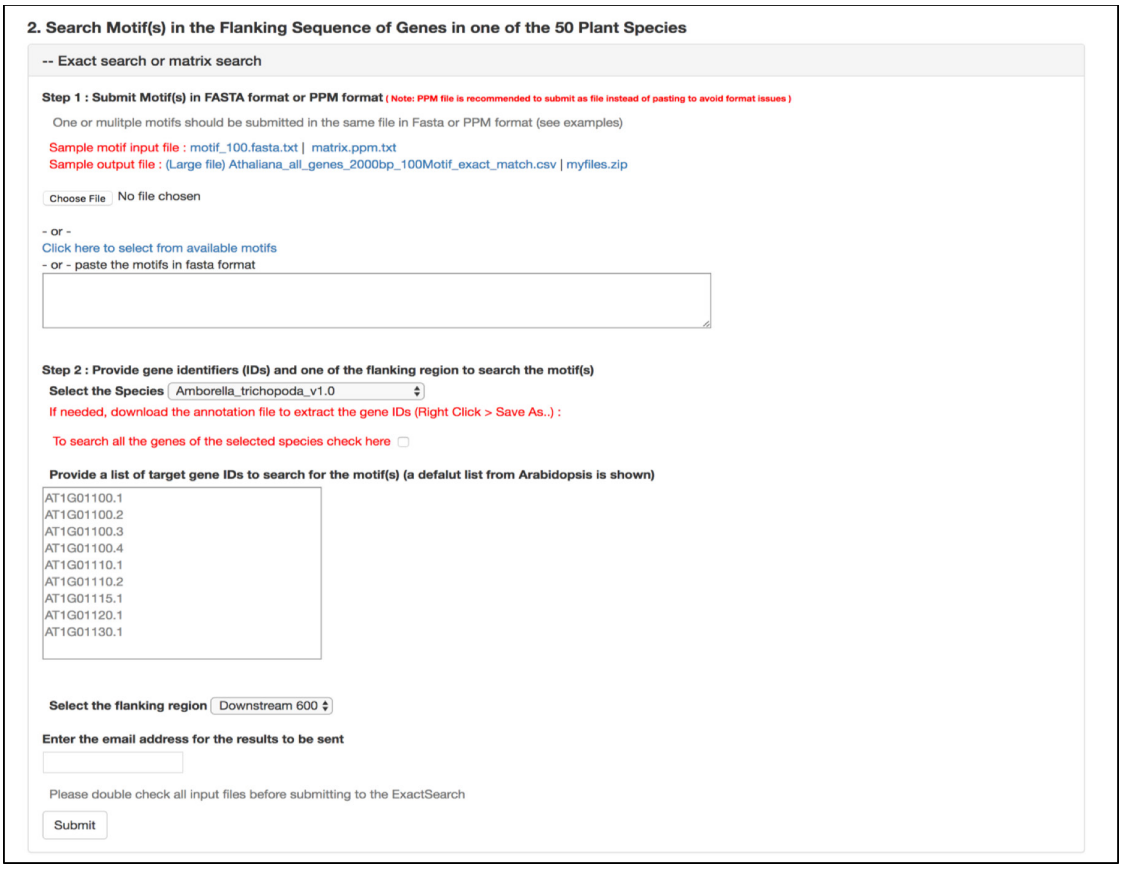


Figure 4.6 User interface for selecting target sequences from 50 plant species. In Step 1, users input sequence motifs by uploading a file or selecting from our motif database. In Step 2, users select which species genome the target sequence should be extracted from and what genes and flanking regions are to be included in the search. This screenshot was obtained from the software created for this dissertation.

4.5 Results and Discussion

4.5.1 Search results

After a user submits input files/select sequences, the search operation is transferred to the web server to execute the ExactSearch algorithm. The sample results shown in Table 4.3 is part of a output file from a genome-wide search of several degenerate motif sequences in the 2kb upstream flanking regions of *Arabidopsis thaliana*. The first column shows the gene IDs of the target sequence file. The second column (Motif) shows the degenerate motif used for the search followed, in curly brackets, by the exact motif found. The third column indicates the length of submitted target sequences. The fourth column (Direction) indicates which strand the motif is located. Moreover, fifth to eight columns indicate the

locations of the motif sequences from the 5' or 3' end of the target gene sequence (*see* Table 4.3).

Table 4.3 A portion of the results of a genome-wide search in the upstream 2kb flanking region of *Arabidopsis thaliana*. The location of the motif sequence in the target sequence is shown from 5` end and from 3` end. Within the curly braces in the Motif column, the exact degenerated sequences matched to the target sequences are shown.

GeneID	Motif	Length of Tai	Direction	Start(5' end)	End(5' end)	End(3' end)	Start(3' end)
ATMG00900	MACGYGB{ACGTGCT}	2000	+	314	320	-1686	-1680
ATMG00900	NGATT{GATTT}	2000	+	392	396	-1608	-1604
ATMG00900	NGATT{GATTT}	2000	+	672	676	-1328	-1324
ATMG00900	NGATT{GATTG}	2000	+	711	715	-1289	-1285
ATMG00900	NGATT{GATTT}	2000	+	716	720	-1284	-1280
ATMG00900	NGATT{GATTC}	2000	+	825	829	-1175	-1171
ATMG00900	NGATT{GATTA}	2000	+	967	971	-1033	-1029

4.6 Discussion

An algorithm is a set of instructions to achieve a well-defined goal. Because the actual run time of an algorithm is affected by uncontrollable variables such as CPU power, the current load of the system, and the programming language used to implement the algorithm, running time is instead measured using complexity analysis. Complexity analysis eliminates the uncertainty in uncontrollable variables by measuring the run time as a function of the size of the input data to the algorithm. For example, in a pattern matching algorithm, if the runtime increases linearly with the length of the input target sequence (n), then the complexity of the pattern matching algorithm is O(n). Table 4.4 shows the comparison of the complexity of ExactSearch algorithm to those of naïve and Rabin-Karp algorithms. The time complexity calculation for both the naïve and Rabin-Karp algorithms factors in the size of the target sequence (n), whereas the time complexity of ExactSearch depends only on the length of the motif sequence (m). Since m is, significantly lower than n, (m<<<n), the search time is greatly improved for ExactSearch in comparison to these other algorithms (O(m) <<< O(n)). The preprocessing time for the ExactSearch algorithm is O(n), but once a target sequence is preprocessed and saved in computer's memory in a tree data structure, it can be repeatedly used to search all other motifs. But, the space complexity of ExactSearch is $O(n^2)$. However, nowadays, computer memory is relatively inexpensive, and a typical high-end computational server with adequate amount of memory will not limit the implementation.

Table 4.4 Complexities of the ExactSearch algorithm compared to those of Naive and Robin-Karp string search algorithms. The Preprocessing Time Complexity, Run Time Complexity, and Space Complexity are compared. The length of a target sequence is given as n, and the length of a motif sequence is given as m.

Algorithm	**Naïve**	**Robin-Karp**	**ExactSearch**
Preprocessing Time	O(m)	O(m)	O(n)
Time Complexity (worst	O(mn)	O(m(n−m+1))	O(m)
Space Complexity	O(m)	O(m)	$O(n^2)$

In this study, we implemented the ExactSearch algorithm into a web application that plant biologists can efficiently use in several search scenarios. These scenarios include:

1. When a list of degenerate motif sequences are needed to be searched in a set of target sequences specified by the user.
2. When a set of degenerate motif sequences is to be searched in promoter regions (0.6 kb downstream or 1.0, 1.5, and 2.0 kb upstream) in the genome of any of the 50 plant species available in ExactSearch web tool.

The ExactSearch algorithm is a very efficient in comparison to the naïve and Rabin-Karp string search algorithms. The ExactSearch web tool was also compared to the existing Regulatory Sequence Analysis Tool (RSAT-DNA) (Thomas-Chollier et al., 2008) using the same input files. For this comparison, we submitted 100 motif sequences and 35,000 target sequences (2kb in length) from *Arabidopsis thaliana* to both the ExactSearch and RSAT-DNA web applications. ExactSearch completed the analysis in 4.2 minutes while the RSAT web-based program took about 45 minutes to produce the same output results. Additionally, in contrast to ExactSearch, the RSAT-DNA web-based program does not provide a built-in target sequence repository for any species. There is currently no comparable web application that has all of the functionalities implemented in our ExactSearch web tool. The other existing motif discovery platforms such as MEME suite (http://meme-suite.org), FIMO (Grant, Bailey, & Noble, 2011), MCAST (Bailey & Gribskov, 1998), and MAST (Bailey & Noble, 2003) require the motifs to be in letter-probability-matrix formats not necessarily similar to ExactSearch web tool. Another tool, GLM2Scan (also in MEME suite), is capable of searching for gapped local alignments of motifs if the motifs are represented in MEME motif format, which is specific to that software tool. Therefore, in comparison to existing software applications in terms of capabilities and file formats, ExactSearch is a different tool. Our web tool caters to the need for fast motif sequence identification in a larger set of target sequences or user-selected genes in proximal promoters from 50 plant species. In light of this, we can

confidently say that the ExactSearch web application will be an indispensable tool for biologists to efficiently conduct motif search tasks.

4.7 Conclusion

We developed a web-based application, ExactSearch that incorporates an efficient pattern-matching algorithm to search for degenerate motifs in the flanking regions of 50 plant genome sequences or user-specified target sequences. Additionally, 400 known plant motifs are stored in our database, which can be searched for in target sequences. Our ExactSearch algorithm has proven to be very fast in locating a set of motif sequences in a set of target sequences. We compared the ExactSearch web application with the Regulatory Sequence Analysis Tool (RSAT)-DNA program by searching for 100 motifs in the flanking regions of 35,000 *Arabidopsis* genes, each having a length of 2kb. The ExactSearch web application completed the search task in about 4.2 minutes, while the RSAT-DNA web tool required about 45 minutes completing the same task. The web application was published in the *Plant Methods* journal (Gunasekara et al., 2016) and is accessible to users via the link: http://sys.bio.mtu.edu/motif. In the future, additional flanking and motif sequences from other plant species can be added to increase the usability of the web application.

4.8 Reference List

Bailey, T. L., & Gribskov, M. (1998). Combining evidence using p-values: application to sequence homology searches. *Bioinformatics, 14*(1), 48-54. doi:DOI 10.1093/bioinformatics/14.1.48

Bailey, T. L., & Noble, W. S. (2003). Searching for statistically significant regulatory modules. *Bioinformatics, 19*, Ii16-Ii25. doi:10.1093/bioinformatics/btg1054

D'Haeseleer, P. (2006). How does DNA sequence motif discovery work? *Nat Biotech, 24*(8), 959-961.

Galil, Z., & Ukkonen, E. (1995). *Combinatorial pattern matching : 6th annual symposium, CPM 95, Espoo, Finland, July 5-7, 1995 : proceedings*. Berlin ; New York: Springer-Verlag.

Goodstein, D. M., Shu, S., Howson, R., Neupane, R., Hayes, R. D., Fazo, J., . . . Rokhsar, D. S. (2012). Phytozome: a comparative platform for green plant genomics. *Nucleic Acids Res, 40*(Database issue), D1178-1186. doi:10.1093/nar/gkr944

Grant, C. E., Bailey, T. L., & Noble, W. S. (2011). FIMO: scanning for occurrences of a given motif. *Bioinformatics, 27*(7), 1017-1018. doi:10.1093/bioinformatics/btr064

Gunasekara, C., Subramanian, A., Avvari, J. V., Li, B., Chen, S., & Wei, H. (2016). ExactSearch: a web-based plant motif search tool. *Plant Methods, 12*, 26. doi:10.1186/s13007-016-0126-6

Gusfield, D., & Kao, M. Y. (1999). Computational biology - Guest editors' foreword. *Algorithmica, 25*(2-3), 141-141. doi:Doi 10.1007/Pl00008271

Lecroq, T. (2007). Fast exact string matching algorithms. *Information Processing Letters, 102*(6), 229-235. doi:10.1016/j.ipl.2007.01.002

Maston, G. A., Evans, S. K., & Green, M. R. (2006). Transcriptional regulatory elements in the human genome. *Annu Rev Genomics Hum Genet, 7*, 29-59. doi:10.1146/annurev.genom.7.080505.115623

Rivals, E., Salmela, L., & Tarhio, J. (2011). Exact Search Algorithms for Biological Sequences. In E. Mourad & Y. Z. Albert (Eds.), *Algorithms in Computational Molecular Biology: Techniques, Approaches and Applications* (pp. 91-111): John Wiley & Sons, Inc.

Thomas-Chollier, M., Defrance, M., Medina-Rivera, A., Sand, O., Herrmann, C., Thieffry, D., & van Helden, J. (2011). RSAT 2011: regulatory sequence analysis tools. *Nucleic Acids Res, 39*(Web Server issue), W86-91. doi:10.1093/nar/gkr377

Thomas-Chollier, M., Sand, O., Turatsinze, J. V., Janky, R., Defrance, M., Vervisch, E., . . . van Helden, J. (2008). RSAT: regulatory sequence analysis tools. *Nucleic Acids Res, 36*(Web Server issue), W119-127. doi:10.1093/nar/gkn304

Ukkonen, E. (1995). On-line construction of suffix trees. *Algorithmica, 14*(3), 249-260. doi:10.1007/bf01206331

Chapter 5

A Web Based Genome Browser for Visualizing Gene Expression Data of miRNA Silencing Lines Generated with Short Tandem Target Mimic (STTM) Technology in Arabidopsis, Rice, Soybean, and Maize[4]

5.1 Abstract

This chapter presents an implementation of a web-based RNA-Seq data visualization platform, STTM JBrowse. This web-based platform can be used to compare alterations in gene expression resulting from the silencing of microRNA (miRNA) by Short Tandem Target Mimic (STTM) technology. Currently STTM JBrowse includes data from several STTM transgenic lines; these include STTM 166, STTM 165/166, STTM 156/157 transgenic lines in *Arabidopsis thaliana*, STTM-MiR165/166 transgenic lines in rice and, transgenic lines from miRNA-targeted soybean. STTM JBrowse has been configured to accept data from transgenic Maize plants, and will be displayed once the data become available. We used the STTM JBrowse to visualize altered gene expression as a result of the silencing of each one of the four microRNAs (miRNAs) in *Arabidopsis Thaliana*, which modulate their target mRNA abundance levels through pairings. Differentially expressed genes were identified by comparing the gene expression levels of STTM transgenic lines with those of wild type (WT) lines. The alternations of gene expression in several biological pathways were examined using the STTM JBrowse visualization platform, which was adopted from an open source genome browser called JBrowse. JBrowse was chosen because its open source nature facilitates easy customization, the capability to handle multiple data tracks for comparisons and lightweight hardware requirements for both clients and servers.

[4] A manuscript presenting STTM JBrowse is currently under review for publication.

5.2 Introduction

Recent advances in high-throughput next generation RNA sequencing technologies have created a new era in transcriptome research. With the ability to generate large volumes of data, researchers now face the challenge of efficiently and effectively visualizing, analyzing, and sharing RNA-Seq data, including scrutinizing thousands of genes (Mangan, Williams, Kuhn, & Lathe, 2014). In recent years, a number of researchers from around the world have been generating RNA-Seq data from Short Tandem Target Mimic (STTM) transgenic lines. STTM technology is used to design a STTM structure based on one or two plant microRNAs (miRNAs). When a STTM structure carried by a binary vector is transferred into plant cells, the STTM structure can recognize miRNA(s) with the complementary sequence(s) for destruction or repression of its translation, resulting in the up-regulation of miRNA targeted genes (G. Tang, 2010; G. Tang et al., 2012). STTM technology has been widely applied to many model and crop plants as a means of altering gene expression (Jia et al., 2015; Teotia, Singh, Tang, & Tang, 2016; Yan et al., 2012).

Currently no unified repository exists that allows plant scientists to efficiently share resources and results generated in STTM RNA-Seq studies. To address this need, I implemented a web-based STTM JBrowse platform for visualizing and sharing RNA-Seq data generated specifically from STTM studies. This platform currently includes data from several STTM transgenic lines, and visualizations which show altered gene expression as a result of miRNA silencing (G. Tang et al., 2012). These include STTM 166, STTM 165/166, STTM 156/157 transgenic lines in *Arabidopsis thaliana*, STTM-MiR165/166 transgenic lines in rice and, transgenic lines from miRNA targeted soybean. In the future, transgenic lines in maize will be added when data become available. This chapter discussed implementation details of this web-based system including customizations added to incorporate transgenic lines from four plant species, methods for data conversion, and modifications to display interface. We adopted an open source genome browser, JBrowse (Skinner et al., 2009), to implement the STTM JBrowse visualization system. JBrowse was chosen because of its JavaScript-based open source technology, which facilitates easy customization and lightweight hardware requirements for distributing large-scale visualizations over the Internet. In comparison to its desktop counterpart Integrative Genome Viewer (IGV) (Robinson et al., 2011; Thorvaldsdottir, Robinson, & Mesirov, 2013), the JBrowse platform offers greater customization and online availability. IGV is a desktop software application that provides individual access for displaying and visualizing RNA-Seq data. The University of Santa Cruz (UCSC) Genome Browser (GBrowse) (Mangan et al., 2014; Stein, 2013) is a web-based RNA-Seq data visualization platform comparable to JBrowse. However, GBrowse is based on a request-response web application architecture which is much slower because more

processing is done on the web server and tracks that are generated must be transferred over the Internet (Rat Genome Browser, 2015). Compared to UCSC GBrowser, the JBrowse has the following advantages:

1) JBrowse is an open-source software platform, which the complete source code is available for customizations, as needed.
2) JBrowse can be implemented on a server as a standalone web application, or it can be embedded into a web application.
3) Complex interactive manipulations can be performed quickly and can be used to generate publication-quality figures.
4) Multiple tracks can be displayed simultaneously without lagging, using a standard desktop computer over the internet.

Using the STTM JBrowse, users have precise access to the sequence reads aligned to any genomic regions of specific interest (*e.g.* location of a gene). This allows easy comparison of changes between gene expression levels of STTM lines and WT lines. After navigating to a particular region, secondary tracks provide genome sequence details and supplementary annotation information. Recently, our STTM JBrowse platform was used to prepare figures for a manuscript titled "*A resource for inactivation of microRNAs using Short Tandem Target Mimic technology in model and crop Plants*". We believe our STTM JBrowse platform will help researchers to visualize the changes of miRNA-targeted genes, and reveal vital information that will aid STTM researchers to better understand regulatory mechanisms involved by miRNAs.

5.3 Materials and Methods

5.3.1 RNA-Seq datasets

Total RNA extracted from wild-type (WT) and STTM transgenic lines was used for RNA-Seq (Wang, Gerstein, & Snyder, 2009). The number of short sequence reads of a particular transcript in an RNA sample can be used to represent the abundance of that mRNA transcript in a given tissue or cell type. Upon aligning the reads to a transcript, a "normalized sequencing depth" can be obtained to represent the expression level of the transcript in a plant sample that was subjected to different treatments, for example, miRNA destruction via STTM technology. Differentially expressed genes in the plant sample that was subjected to a specific miRNA destruction via STTM technology can be identified in comparison with a wild-type(WT) sample. The sequencing reads generated

from high-throughput sequencers can be stored in FASTQ format as raw output. As shown Figure 5.1 in the each short read is associated with a read identifier in the first line, which starts with @ sign. The second line contains read sequence, the third line indicates the read strand, and the fourth line gives quality scores (as symbols) for each base in the short read in Line 2 (Cock, Fields, Goto, Heuer, & Rice, 2010).

A sample short read sequence

```
1° @K00136:41:H35MWBBXX:7:1101:1944:1562 1:N:0
2° CTTTACAGGTACAGCTGCAGTATCTTCAGGAGCTTTCACTTGTACTTTA
3° +
4° ``e[[KKK`ee[K`[e``Vejjj`VjKKKVKeVKKKVKK`KK`[e```K
```

Figure 5.1 A sample short read sequence. Line 1, starting with an @ sign, is a read identifier. Line 2 is the short read RNA/mRNA sequence. Line 3 begins with + sign and optionally follows another identification pattern. Line 4 gives a quality score for each base

The Table 5.1 shows currently available datasets in the STTM JBrowse platform submitted by users from various institutions. Due to the large file size, each FASTQ file is typically compressed in .gz format before transferring over the internet to our web server.

Table 5.1 Currently available datasets displayed in the STTM JBrowse. The resources are accessible through the http://blossom.ffr.mtu.edu. As more data become available, the online STTM JBrowse resource will be more comprehensive.

Species	miRNA Silencing line/ condition	Number of STTM Samples	Number of WT Samples	Contributor
Arabidopsis thaliana	STTM166	3	3	Dr. Guiliang Tang
Arabidopsis thaliana	STTM 156/157	1	1	Dr. Ting Peng
Arabidopsis thaliana	STTM 165/166	1	1	Dr. Ting Peng
Arabidopsis thaliana	STTM 172	1	1	Dr. Ting Peng
Rice	STTM166	3	3	Dr. Guiliang Tang
Soybean	Transgenic plants	24	0	Dr. Harold Trick
Soybean	Water treatment	1	1	Dr. Wenbo Ma
Soybean	Phytophthora sojae infected	1	1	Dr. Wenbo Ma

5.3.2 Data Processing and STTM JBrowse Deployment

5.3.2.1 Setting up STTM JBrowse server environment

The Apache web server accepts requests from a client's browser (*e.g.*, Google Chrome, Internet Explorer) and the Apache web server sends the necessary page layout instructions to client web browser to generate visualizations as requested. STTM JBrowse works similar to a typical web application, and when a client requests access, it transfers page layout instructions using HTML and CSS elements to a client web browser to display RNA-Seq data tracks. As shown in Figure 5.2, STTM JBrowse has been developed using off-the-shelf open source software platforms; the front-end is

implemented using HTML, CSS, and JavaScript, and the backend data processing and user request handling is done with PHP and Perl scripts. The Apache open source web server is used to host the PHP and Perl scripts on the server computer, and a client's RNA-Seq data files are stored in the server using a separate file system that is linked to the STTM JBrowse web application via configuration files which stores information about data files, track types, track names and locations.

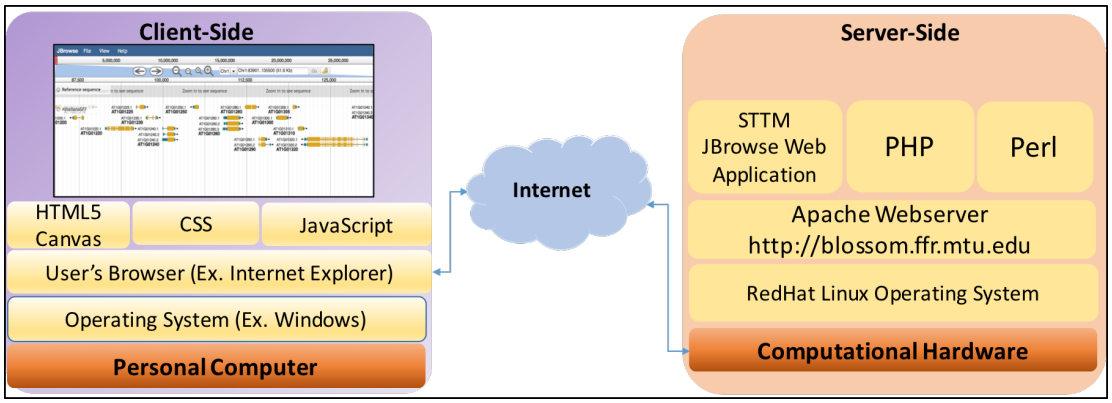


Figure 5.2 Technology stack of STTM JBrowse implementation. The client-side functions as a typical web-based application for which a user can direct the web browser to the application via a web address. The server-side implements the JBrowse application as a web application.

The Apache server provides /var/www in the root directory to be used to store the JBrowse application scripts. To deploy JBrowse, the base package from www.jbrowse.org is first downloaded to the /var/www/JBrowse directory, which is created using, command 1 and 2 shown in Procedure 5.1. The mime_magic module in the Apache web server was disabled so that the JBrowse software would function correctly. If not disabled, this module causes the visualization tracks to incorrectly read the data files and typically results in errors. The commands 3 to 7 on (Procedure 5.1) were run to properly install the required Perl modules on the web server.

Download and install JBrowse based package from online repository

```
1° sudo mkdir /var/www/JBrowse1.11.5;
2° sudo chown `whoami` /var/www/JBrowse1.11.5;
3° cd /var/www/JBrowse1.11.5;
4° curl -O http://jbrowse.org/releases/JBrowse-1.11.5.zip
5° unzip JBrowse-x.x.x.zip
6° cd JBrowse1.11.5
7° sudo ./setup.sh
```

Procedure 5.1 Summary of system commands for the initial installation of the JBrowse software on the web server.

5.3.2.2 Configuration of STTM JBrowse for each species (Arabidopsis thaliana, Rice, Soybean, Maize)

After setting up the JBrowse base packages and Perl libraries, the URL http://blossom.ffr.mtu.edu/jbrowse/index.html showed a blank interface of JBrowse platform; the absence of any error messages indicated that the installation completed correctly. For illustration and demonstration purposes, the configurations and set up of the *Arabidopsis thaliana* reference genome track and General Feature Format (GFF3) annotation track will be described in subsequent sections. Configuration and set up with rice, soybean, and maize reference genomes and annotation tracks followed the same process used for *Arabidopsis thaliana*. The reference genomes and GFF3 annotations for each of the four plant species were obtained from the online repository www.Phytozome.org (Goodstein et al., 2012). The miRNA database file, also in GFF3 format, was downloaded from the online repository http://www.mirbase.org (Kozomara & Griffiths-Jones, 2014). The JBrowse1.11.5 directory contains all the files required for the JBrowse system configurations. As shown in Figure 5.3, the `JBrowse1.11.5/bin` directory contains preprocessing Perl scripts while the `JBrowse1.11.5/species` directory contains annotation tracks, reference genome and converted data files.

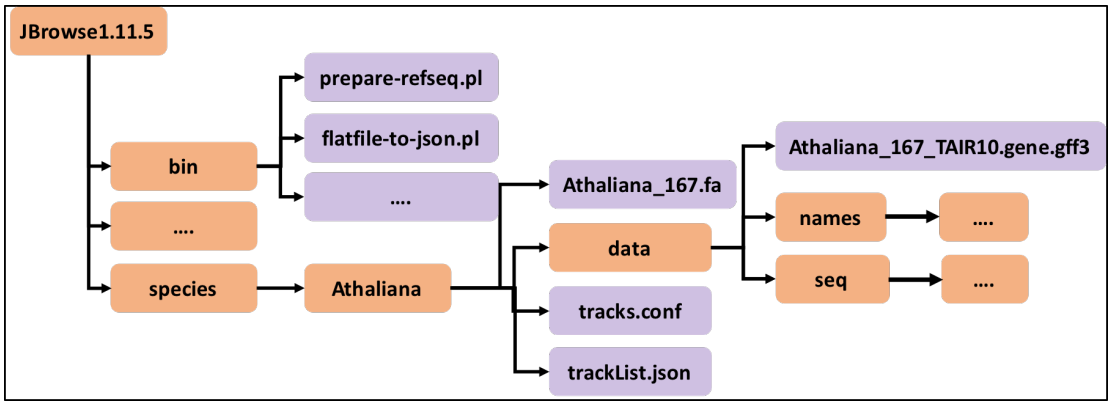


Figure 5.3 STTM JBrowse directory structure for *Arabidopsis thaliana* after initial installation of reference genome sequence and annotation tracks. Light brown colored shapes are directories, and purple colored shapes are files. The `prepare-refseq.pl` script converts the `Athaliana_167.fa` file to JavaScript Object Notation (JSON) format and stores in the `data/seq` directory. The GFF3 file `Athaliana_167_TAIR10.gene.gff3` contains the gene annotations for this reference genome. The flatfile-to-json.pl script in the bin directory converts the GFF3 file to JSON format that STTM JBrowse can read efficiently. The converted JSON files are stored the `data/names` directory.

5.3.2.3 Conversion of genome sequences to display the reference sequence track on the STTM JBrowse interface

The reference sequence track of each species determines a common coordinate system that serves as the basis for respective annotations and data tracks in the visualization system. To generate this track, the `bin/prepare-refseq.pl` Perl script file shown in Figure 5.3, was used in the command line with the *Arabidopsis thaliana* reference genome file `Athaliana_167.fa` as input parameter. The commands in Procedure 5.2 were followed for this preprocessing step. Commands 1-3 created the required directory. Command 4 changes the current path into the location of `Athaliana` directory where reference genome file is located. Command 5 converts the reference genome and the stores in the JSON format in `data/seq` directory (Figure 5.3). After running the commands, the JBrowse interface showed an option to turn on/off the reference genome track and the genome sequence characters were displayed as in Figure 5.4.

Create reference genome tracks for *Arabidopsis Thaliana*

```
1° cd /var/www/JBrowse1.11.5/species;
2° mkdir Athaliana;
3° sudo chown `whoami` Athaliana;
4° cd /var/www/JBrowse1.11.5/species/Athaliana;
5° bin/prepare-refseqs.pl --fasta ./Athaliana_167.fa
```

Procedure 5.2 Instructions set to create the reference sequence track for *Arabidopsis thaliana*.

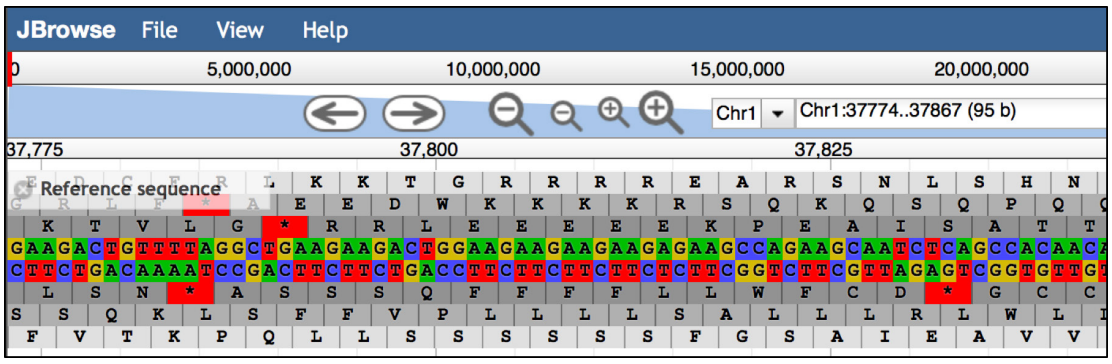


Figure 5.4 STTM JBrowse interface after the *Arabidopsis thaliana* reference genome was preprocessed and integrated to application. To display the sequence characters, a user needs to zoom in on the track until the desired details become visible. This screenshot was obtained from the software created for this dissertation.

5.3.2.1 Conversion of GFF3 file to display gene annotation tracks on the STTM JBrowse interface

The latest version of the *Arabidopsis thaliana* GFF3 file was downloaded from www.phytozome.org. This file is in tab delimited format (*see* Figure 5.5). Each entry consists of nine fields: "seqid" - chromosome number, "source" - the repository the data was extracted from, "type" - type of feature, "start" - start position, "end" - end position, "score" - a quality score floating point number, "strand" - forward (+) or reverse (-), "phase" - 0, 1, or 2 for base of codon, "attributes" - additional features.

```
Chr1    phytozomev10    gene    3631    5899    .    +    .
        ID=AT1G01010.TAIR10;Name=AT1G01010
Chr1    phytozomev10    mRNA    3631    5899    .    +    .
        ID=AT1G01010.1.TAIR10;Name=AT1G01010.1;pacid=19656964;longest=1;Parent=AT1G01010.TAIR10
Chr1    phytozomev10    five_prime_UTR    3631    3759    .    +    .
        ID=AT1G01010.1.TAIR10.five_prime_UTR.1;Parent=AT1G01010.1.TAIR10;pacid=19656964
Chr1    phytozomev10    CDS    3760    3913    .    +    0
        ID=AT1G01010.1.TAIR10.CDS.1;Parent=AT1G01010.1.TAIR10;pacid=19656964
Chr1    phytozomev10    CDS    3996    4276    .    +    2
        ID=AT1G01010.1.TAIR10.CDS.2;Parent=AT1G01010.1.TAIR10;pacid=19656964
```

Figure 5.5 First five genes from *Arabidopsis thaliana* genome in GFF3 format. Each entry contains nine fields. If a field is empty it is shown as "."

Information about genes, such as IDs, coordinates, etc. available in this GFF3 file are converted to JSON format by running the commands shown in Procedure 5.3. The command 1 is executed from the data directory (*see* Figure 5.3) with the following input parameters:

--gff : location of *Arabidopsis thaliana* GFF3 file

--trackType : specified type for JBrowse platform (CanvasFeatures)

--trackLabel : name for the GFF3 annotation track in the JBrowse interface (AT_GFF)

The command 2 generates an index from the GFF3 file, which allows searching for genes by name in the search box provided in the JBrowse interface. The same commands in Procedure 5.3 was followed for the miRNA database file, which is also in GFF3 format. After running the commands in Procedure 5.3, the JBrowse interface showed an option to turn on/off the GFF tracks and the annotations was visualized as shown in Figure 5.6.

Create annotation tracks using gff3 flat file

```
1° ../../../bin/flatfile-to-json.pl –gff
./Athalina_167_tair10.gene.gff3 --trackType CanvasFeatures --
trackLabel AT_GFF
2° ../../../bin/generate-names.pl -v
```

Procedure 5.3 Instructions to create annotation tracks using GFF3 file for *Arabidopsis thaliana*.

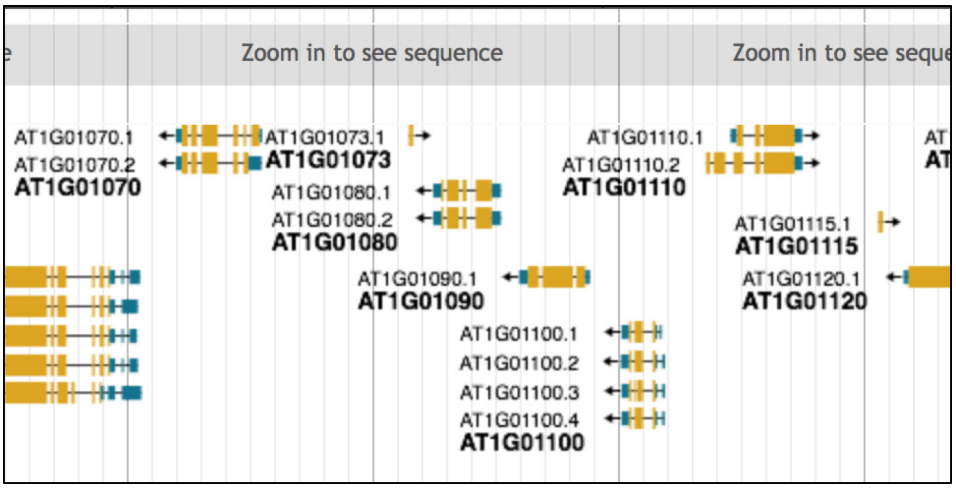


Figure 5.6 The annotation tracks generated from the *Arabidopsis thaliana* GFF3 file. A user can get more information about each feature by clicking on a feature ID. When the user zooms in, the feature will expand to show the range of each genomic feature. This screenshot was obtained from the software created for this dissertation.

5.3.2.2 RNA-Seq data conversion from FASTQ to Binary Alignment Map (BAM) format

The raw short read RNA-Seq data obtained from high throughput sequencers are stored in FASTQ format (Conesa et al., 2016). FASTQ data was converted to Binary Alignment Map (BAM) format, which can be programmatically parsed by Perl scripts to display data tracks in the STTM JBrowse interface. There are many software tools available to do this conversion. To convert the FASTQ file obtained from sequencing wild type (WT) or STTM samples, we chose to use the following two-step pipelines because the software tools are freely available as open source software:

Step 1: Constructing sequence indexes from the reference sequence using "Bowtie2-build".

To construct an index from the reference genome to which sequencing reads can be aligned; we used Bowtie2-build open source software. This command line executable program is available in the suite of Bowtie software available as an open source tool to align sequencing reads to long reference sequence (Langmead & Salzberg, 2012). A reference genome downloaded from an online repository (*e.g.* Phytozome.org) is used directly as an input parameter to the Bowtie2-builder software, as shown in the following command; In this case, we have utilized the reference genome from *Arabidopsis thaliana* obtained from the Phytozome.org repository (Lamesch et al., 2012).

Generate bowtie index from the reference genome

```
1° bowtie2-build -f ./AT167_index Athaliana_167.fa AT167 --threads 8
```

Procedure 5.4. Create a bowtie index from the *Arabidopsis Thaliana* reference genome.

This command generates six index files with the specified format: AT.rev.1.bt2, AT.rev.2.bt2, AT167.1.bt2, AT167.2.bt2, AT167.3.bt2, AT167.4.bt2. These index files should be placed in a single directory named without spaces or special characters, as shown in Figure 5.7.

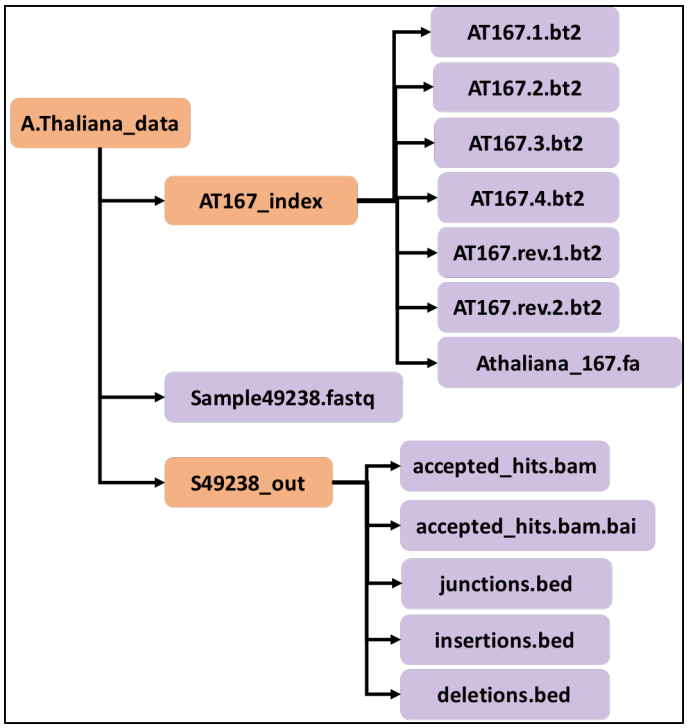


Figure 5.7 File structure for converting FASTQ format to BAM format. Bowtie index files are stored in the AT167_index directory. This directory contains six files auto-generated from Bowtie2 software when indexing the *Arabidopsis thaliana* reference genome (Athaliana_167.fa). TopHat software utilizes the files in AT167_index and generates the S49238_out directory. Samtools software was used to create the BAM index file (.bai file) shown in the S49238_out directory. The light brown color represents the directories.

Step 2: Mapping sequencing reads to the reference genome using TopHat software.

The TopHat tool was used to convert the FASTQ files to BAM format (Kim et al., 2013). To convert the *Arabidopsis thaliana* FASTQ file (`sample49238.fastq`) to BAM

format, Command 1 in Procedure 5.5 was executed from the `A.thaliana_data` directory shown in Figure 5.7.

RNA-Seq data from FASTQ format to BAM format conversion steps
1° `tophat -o ./S49238_out/ -p 8./AT167_index/AT167 ./Sample49238.fastq`
2° `Samtools index ./S49238_out/accepted_hits.bam`

Procedure 5.5 Steps to convert FASTQ data to BAM format and generate bam indexes.

The TopHat command needs the following arguments and options to be set correctly:

<genome_index_base>: This argument is set by providing the relative path to the directory where the genome index created by Bowtie2-build software is located. The path should be appended with index filenames up to the first period that are common to all index files.

-o: The output directory where the resulting BAM file is to be stored. This should be a unique directory name for each FASTQ file to avoid overwriting existing BAM files. The output files in the output directory are `accepted_hits.bam`, `junctions.bed`, `insertions.bed`, and `deletions.bed`.

-p: Number of threads used in the CPU to run the TopHat command.

Samtools software is used to generate an index file for the BAM file (H. Li et al., 2009). The `accepted_hits.bam` file with the index flag is provided to the Samtools software, which produces the `accepted_hits.bam.bai` file. This index is used by the STTM JBrowse platform to search by gene ID and allows the visualization interface to move directly to the location of the gene in the genome efficiently.

5.3.3 Overall workflow from RNA-Seq data to STTM JBrowse visualization

The overall pipeline from initial setting up to display converted RNA-Seq data in the STTM JBrowse interface is shown in Figure 5.8. Building the backend genomic feature tracks using reference genome sequence and GFF3 files needs to be conducted for only one time for each species. The rest processes must be done for each RNA-Seq sample submitted to the STTM JBrowse.

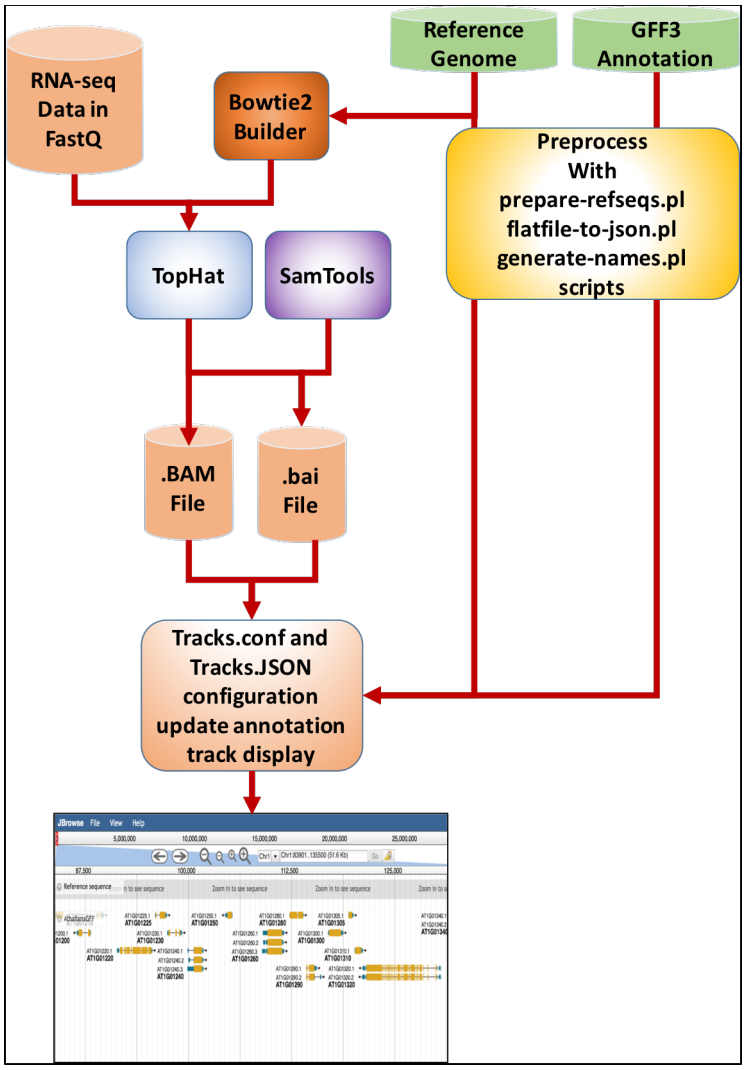


Figure 5.8 Overall work flow required to set up the STTM JBrowse visualization pipeline. The reference genome and GFF3 feature annotation files downloaded from Phytozome.org were preprocessed with the `prepare-refseqs.pl`, `flatfile-to-json.pl`, and `generate-names.pl` Perl script files. Any RNA-Seq files that are submitted by users should be preprocessed with Bowtie2 builder, TopHat, and Samtools software and loaded to the STTM JBrowse web server file system. Tracks.conf and Tracks.JSON files provide the STTM JBrowse system with information regarding the type of each track, category, key names, and location of .BAM and .bai files.

5.3.4 Customizing configuration files for visualization and CSS file adjustments

The converted data files in .bam and .bai format from *Arabidopsis thaliana* samples (WT and STTM) were placed in the Athaliana/data directory, as shown in Figure 5.9. The same process was followed for the other species in their respective directories.

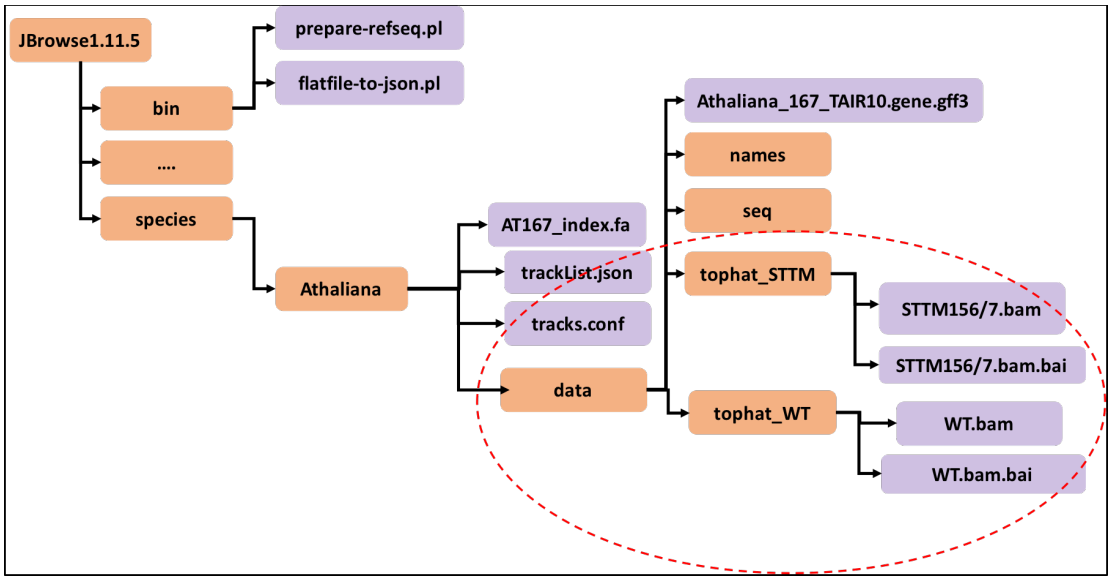


Figure 5.9 File structure for two *Arabidopsis thaliana* data tracks (WT, STTM). `tophat_STTM` and `tophat_WT` directories in the red dashed circle make the data files to be read by the visualization system. The `tracks.conf` defines information about RNA-Seq data the tracks of *Arabidopsis thaliana* transgenic lines.

The `tracks.conf` file is used to define the properties about each data track displayed in the STTM JBrowse visualization (one for each species). The `tracks.conf` file is parsed by the STTM JBrowse system to extract information such as the location of the converted BAM files, track type, name, category, *etc*. Each BAM file is displayed using two types of tracks: 1) "Alignment2" track type show mapped read alignments to the reference genome. The BAM files were sorted by the leftmost coordinate for rapid alignment, when producing the visualization. Alignment2 tracks require most of the interface area to visualize; as a result, the comparison of multiple tracks is difficult. 2) "SNP Coverage" track type show the overall histogram of sequencing read alignments for each mapped gene. With this track type, it is easy to compare multiple samples. As shown in Figure 5.10. The `tracks.conf` file stores the information about configurations needed for each track about the STTM and WT samples.

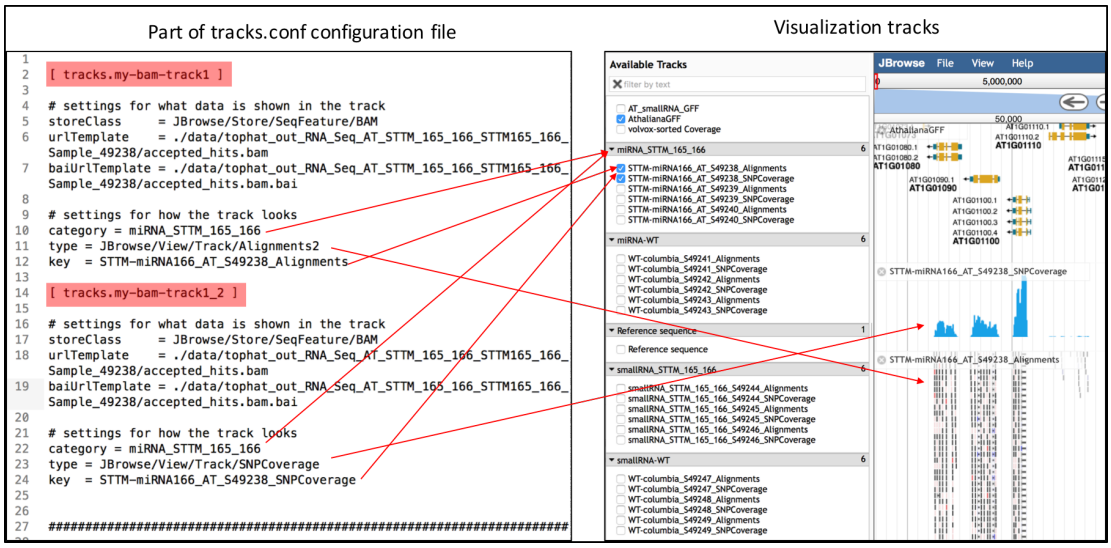


Figure 5.10 Illustration of Configuration file; the `tracks.conf` (left panel) and corresponding visualization tracks (right panel) defined by `tracks.conf`. The `tracks.conf` file used in here is the one that defines the tracks of RNA-seq data from *Arabidopsis thaliana* transgenic lines. This screenshot was obtained from the software created for this dissertation.

5.3.5 Coverage histogram scale and color modifications to customize visualizations

The "SNP Coverage" track type adjusts the histogram scale according to the maximum value for each histogram individually for each track. Since we are interested in comparison between of tracks that usually correspond to a STTM line and a WT line, the histograms can be easily compared when the same y-scale range is used. Two parameters ("Min_value" and "Max_value") were introduced to limit the range of the y-axis in both tracks based on the largest density value obtained from both histograms. This modification was added to the `trackList.json` file, as shown Figure 5.11A, which made each histogram to share the same scale for y-axis range. The "min value", "max_value", "style:color" parameters in the `trackList.json` file is read by the STTM JBrowse system files, and used to define the CSS format, leading to the modified histogram shown in Figure 5.11B.

(A)

```
        {
        "storeClass"  :  "JBrowse/Store/SeqFeature/BAM",
        "urlTemplate" : "./S156157/S156157.sorted.bam",
        "label"       : "STTM_sorted.bam_coverage",
        "type"        : "JBrowse/View/Track/FeatureCoverage",
        "min_score"   : 0,
        "max_score"   : 700,
        "key"         : "STTM_SNP_Coverage",
        "style": { "color": "darkblue" }
    },{
        "storeClass"  :  "JBrowse/Store/SeqFeature/BAM",
        "urlTemplate" : "./WT/WT.sorted.bam",
        "label"       : "WT_bam_coverage",
        "type"        : "JBrowse/View/Track/FeatureCoverage",
        "min_score"   : 0,
        "max_score"   : 700,
        "key"         : "WT_SNP_Coverage",
        "style": { "color": "darkblue" }
    }
```

(B)

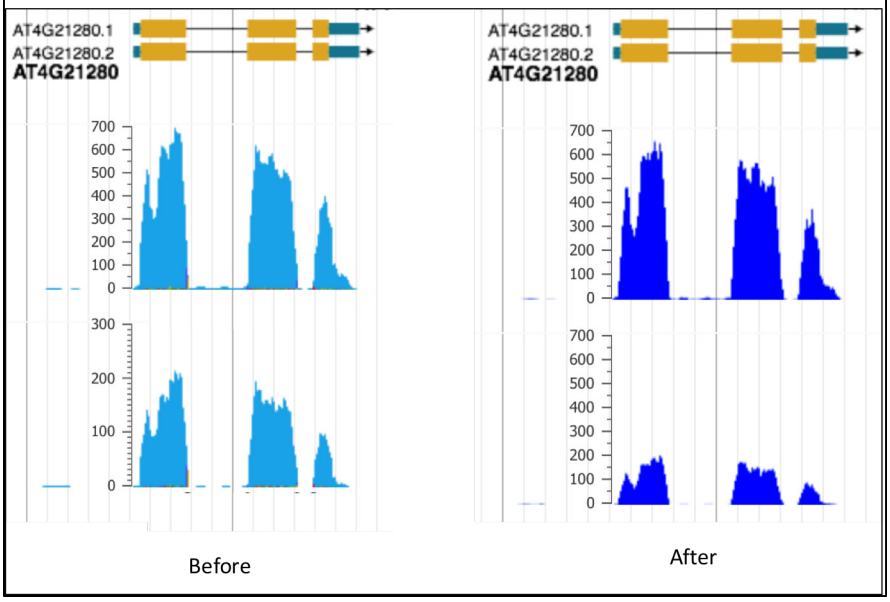


Figure 5.11 Modifications to visualization configuration file for easier comparisons and visualization. A) Overriding the default parametes of y-axis scale and histogram color by adding min_value/max_value and style parameters to the `trackList.json` file, respectively. The AT4G21280 gene from *Arabidopsis thaliana* was used as an example to compare the visualization effects before (left) and after (right) customizing the coverage histogram parameters. B) The changes in the histograms show comparisons are easier with the modified parameters. This screenshot was obtained from the software created for this dissertation.

5.4 Results and Discussion

5.4.1 Arabidopsis thaliana

5.4.1.1 Photosynthesis and anthocyanin biosynthesis pathways (STTM156/157)

Utilizing the STTM JBrowse visualization platform, expression of genes involved in photosynthesis and anthocyanin biosynthesis pathways was compared using WT and STTM tracks. Figure 5.12A shows in STTM 156/157 transgenic lines in comparison to WT for photosynthesis pathway genes. Following genes show a significant upregulation in STTM 156/157 transgenic lines; PSB27 (L. L. Wei et al., 2010), PSB28 (Sakata, Mizusawa, Kubota-Kawai, Sakurai, & Wada, 2013), PSB29 (Keren, Ohkawa, Welsh, Liberton, & Pakrasi, 2005), PSB-01/02 (Spence et al., 2014), PSBP-1/2 (Ifuku, Yamamoto, Ono, Ishihara, & Sato, 2005), PSBQ-2 (Yi, Hargett, Frankel, & Bricker, 2006), PSBTN (Shi & Schroder, 2004), PSBQA (Gaur & Tyagi, 2004), PSBX (Funk, 2000), PSBW (Garcia-Cerdan et al., 2011), and PSBY (Neufeld, Zinchenko, Stephan, Bader, & Pistorius, 2004). In examining the anthocyanin pathway genes shown in Figure 5.12B it is clear that SPL9 (Gou, Felippes, Liu, Weigel, & Wang, 2011b) and TTG1 (L. L. Zhou, Shi, & Xie, 2012) are up-regulated while PAP1 (Shin et al., 2015), TT8, do not show a noticeable difference in STTM156/157 transgenic plants compared to WT plants. These results indicate that STTM JBrowse can be used to visually identify differentially expressed genes and to create figures for use in publications or manual securitizing effects of miRNAs on genes.

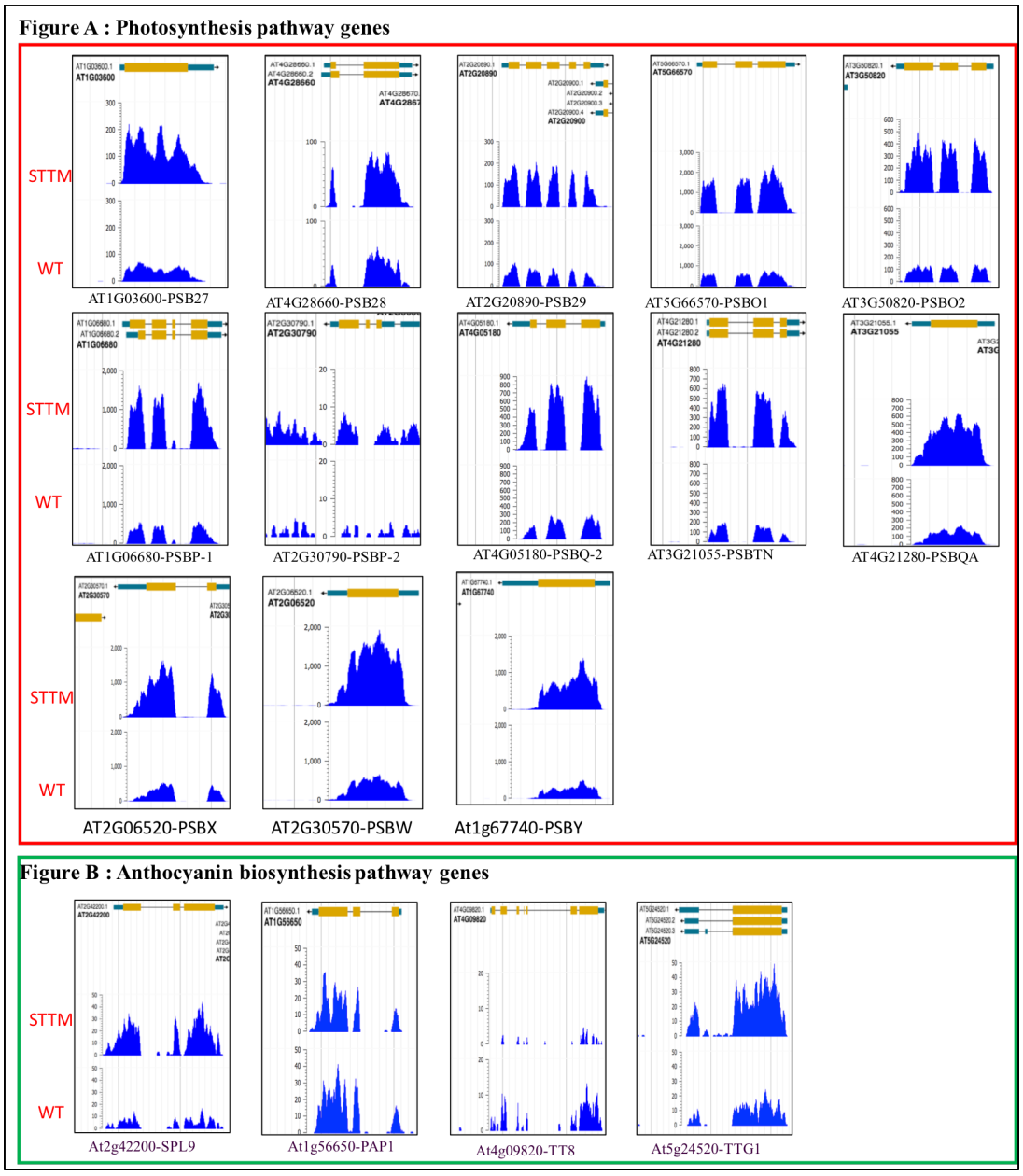


Figure 5.12 Visualization of differentially expressed genes using the STTM JBrowse platform Photosynthesis and Anthocyanin biosynthesis pathway genes. The data were from Arabidopsis thaliana wild-type (WT) and STTM156/157 (STTM) Arabidopsis thaliana transgenic plan

5.4.1.2 ABA and Auxin biosynthesis and signaling pathway genes (STTM 165/166)

Utilizing the STTM JBrowse visualization platform, expression of genes involved in various aspects of ABA biosynthesis and signaling pathway was compared in WT and STTM transgenic plants. NCED4 (Huo, Dahal, Kunusoth, McCallum, & Bradford, 2013), a gene involved in ABA biosynthesis, is significantly up-regulated in the STTM transgenic line in comparison to wild-type plants (Figure 5.13 A). TAA1, CYP79B2, and CYP79B3, genes related to auxin biosynthesis pathway(Falkenberg et al., 2008; P. P. Liu et al., 2007; Mashiguchi et al., 2011), were up-regulated as shown in Figure 5.13 B. Also Figure 5.13 C & Figure 5.13 D show a few YUC family genes (Hofmann, 2011) and a few GH3 family of genes (Park et al., 2007) related to auxin biosynthesis in STTM165/166 miRNA transgenic plants.

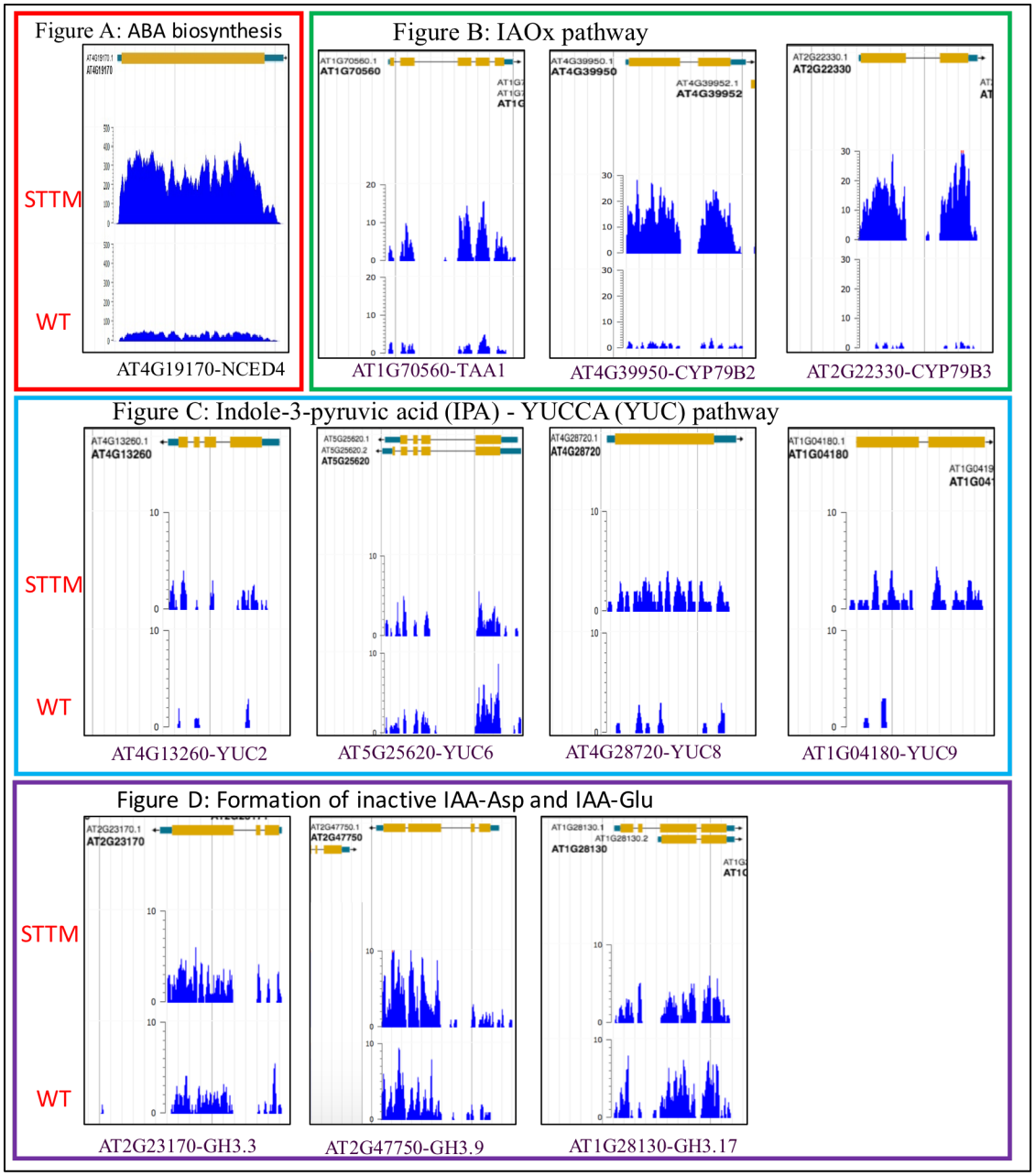


Figure 5.13 Visualization of differentially expressed genes using the STTM JBrowse platform for ABA and Auxin biosynthesis pathway genes. The comparisons between wild-type (WT) and STTM165/166 transgenic *Arabidopsis thaliana* plants. The genes shown in this figure are involved in: A) ABA biosynthesis pathway B) IAOx pathway C) IPA-YUC pathway D) Inactivation of IAA biosynthesis. Users can easily navigate by searching for a specific gene ID in the search bar of the visualization interface. Screen captures can be used to save figures on a local computer

5.4.2 Rice

The rice reference genome (`Osativa_204_v7.0.transcript.fa`) and GFF3 annotation (`Osativa_204_v7.0.gene.gff3`) files were downloaded from the www.Phytozome.org repository (Goodstein et al., 2012). The same preprocessing steps specified for *Arabidopsis thaliana* were followed for processing the rice reference genome, annotation files and data files. The resulting genome-wide visualizations are shown in Figure 5.14.

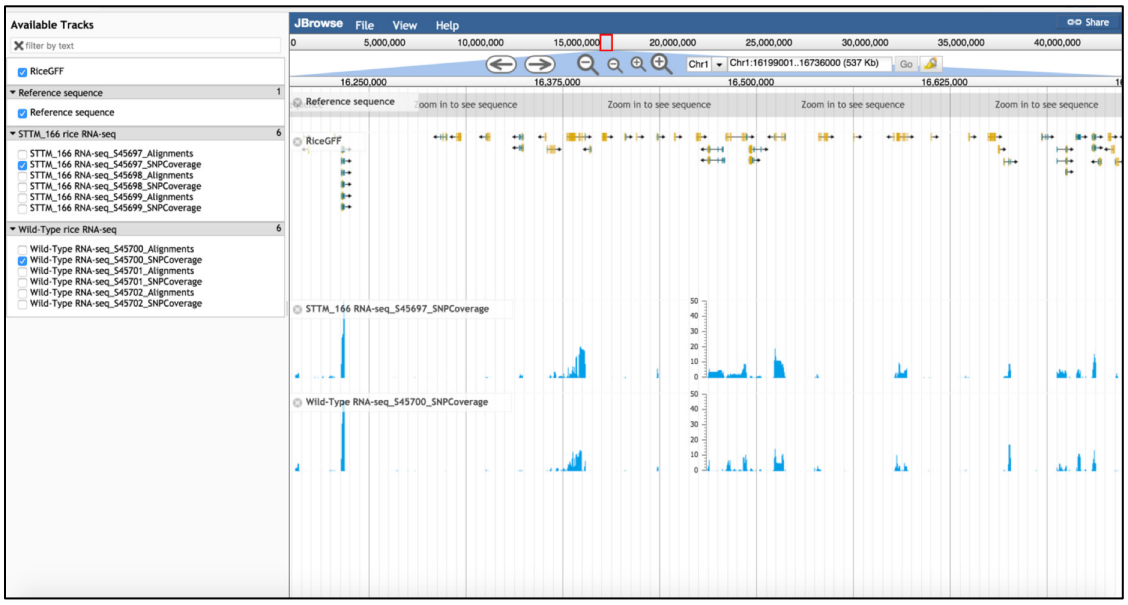


Figure 5.14 STTM JBrowse visualization for transgenic lines in rice. Six RNA-Seq (3-WT, 3-STTM) samples from rice were added to this visualization. The panel on the left can be used to turn on/off specific tracks for comparison. This screenshot was obtained from the STTM JBrowse software created for this dissertation.

5.4.3 Soybean

The soybean reference genome and GFF3 annotation (`transcript.fa` and `Gmax_189_gene.gff3`) files were downloaded from the www.Phytozome.org repository (Goodstein et al., 2012). The same preprocessing steps specified for *Arabidopsis thaliana* were followed for the Soybean reference genome, annotation files and data files. The resulting genome-wide visualizations are shown in Figure 5.15.

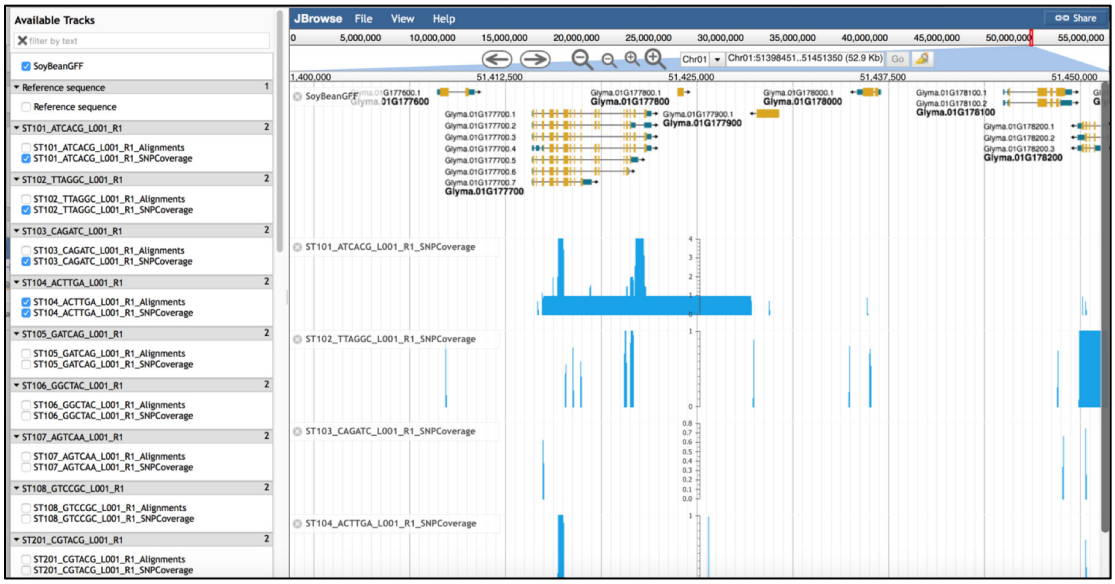


Figure 5.15 STTM JBrowse visualization from transgenic lines in soybean. The left panel can turn on/off 24 RNA-Seq data tracks from soybean which were added to this visualization. The panel on the left can also be used to turn on/off specific tracks for comparisons. This screenshot was obtained from the STTM JBrowse software created for this dissertation.

5.4.4 Maize

The software platform for integrating RNA-Seq data from Maize STTM transgenic and WT plants has been configured. However, unfortunately data has not been available yet to display in the STTM JBrowse platform. Once the data is available, the STTM JBrowse resource maintainer will be able to upload the data for visualization easily.

5.5 Conclusion

A web-based visualization platform called STTM JBrowse was developed to visualize RNA-Seq data generated from STTM transgenic plants. JBrowse, a next generation open source genome browser with many advantages over UCSC GBrowse or IGV, was used to build the visualization platform, with the implementation details of JBrowse platform being provided. Currently, our STTM JBrowse visualization platform harbors RNA-Seq data from *Arabidopsis thaliana*, rice, soybean, and maize STTM mutant lines. It can serve as a hub for researchers who study miRNA functions via STTM technology around the world and are intended to share their findings and visualizations. The STTM JBrowse platform provides a user-friendly, seamless platform to submit data and to share research findings regarding miRNA functionalities in plants. STTM JBrowse will greatly facilitate the progress of STTM research in the future.

5.6 Reference List

Cock, P. J., Fields, C. J., Goto, N., Heuer, M. L., & Rice, P. M. (2010). The Sanger FASTQ file format for sequences with quality scores, and the Solexa/Illumina FASTQ variants. *Nucleic Acids Res, 38*(6), 1767-1771. doi:10.1093/nar/gkp1137

Conesa, A., Madrigal, P., Tarazona, S., Gomez-Cabrero, D., Cervera, A., McPherson, A., . . . Mortazavi, A. (2016). A survey of best practices for RNA-seq data analysis. *Genome Biology, 17*. doi:ARTN 13
10.1186/s13059-016-0881-8

Falkenberg, B., Witt, I., Zanor, M. I., Steinhauser, D., Mueller-Roeber, B., Hesse, H., & Hoefgen, R. (2008). Transcription factors relevant to auxin signalling coordinate broad-spectrum metabolic shifts including sulphur metabolism. *Journal of Experimental Botany, 59*(10), 2831-2846. doi:10.1093/jxb/ern144

Funk, C. (2000). Functional analysis of the PsbX protein by deletion of the corresponding gene in Synechocystis sp. PCC 6803. *Plant Molecular Biology, 44*(6), 815-827.

Garcia-Cerdan, J. G., Kovacs, L., Toth, T., Kereiche, S., Aseeva, E., Boekema, E. J., . . . Schroder, W. P. (2011). The PsbW protein stabilizes the supramolecular organization of photosystem II in higher plants. *Plant Journal, 65*(3), 368-381. doi:10.1111/j.1365-313X.2010.04429.x

Gaur, T., & Tyagi, A. K. (2004). Analysis of Arabidopsis PsbQA gene expression in transgenic tobacco reveals differential role of its promoter and transcribed region in organ-specific and light-mediated regulation. *Transgenic Res, 13*(2), 97-108.

Goodstein, D. M., Shu, S., Howson, R., Neupane, R., Hayes, R. D., Fazo, J., . . . Rokhsar, D. S. (2012). Phytozome: a comparative platform for green plant genomics. *Nucleic Acids Res, 40*(Database issue), D1178-1186. doi:10.1093/nar/gkr944

Gou, J. Y., Felippes, F. F., Liu, C. J., Weigel, D., & Wang, J. W. (2011b). Negative regulation of anthocyanin biosynthesis in Arabidopsis by a miR156-targeted SPL transcription factor. *Plant Cell, 23*(4), 1512-1522. doi:10.1105/tpc.111.084525

Hofmann, N. R. (2011). YUC and TAA1/TAR proteins function in the same pathway for auxin biosynthesis. *Plant Cell, 23*(11), 3869. doi:10.1105/tpc.111.231112

Huo, H., Dahal, P., Kunusoth, K., McCallum, C. M., & Bradford, K. J. (2013). Expression of 9-cis-EPOXYCAROTENOID DIOXYGENASE4 Is Essential for Thermoinhibition of Lettuce Seed Germination but Not for Seed Development or Stress Tolerance. *The Plant Cell Online*. doi:10.1105/tpc.112.108902

Ifuku, K., Yamamoto, Y., Ono, T.-a., Ishihara, S., & Sato, F. (2005). PsbP Protein, But Not PsbQ Protein, Is Essential for the Regulation and Stabilization of Photosystem II in Higher Plants. *Plant Physiology, 139*(3), 1175-1184. doi:10.1104/pp.105.068643

Jia, X., Ding, N., Fan, W., Yan, J., Gu, Y., Tang, X., . . . Tang, G. (2015). Functional plasticity of miR165/166 in plant development revealed by small tandem target mimic. *Plant Sci, 233*, 11-21. doi:10.1016/j.plantsci.2014.12.020

Keren, N., Ohkawa, H., Welsh, E. A., Liberton, M., & Pakrasi, H. B. (2005). Psb29, a conserved 22-kD protein, functions in the biogenesis of photosystem II complexes in Synechocystis and Arabidopsis. *Plant Cell, 17*(10), 2768-2781. doi:10.1105/tpc.105.035048

Kim, D., Pertea, G., Trapnell, C., Pimentel, H., Kelley, R., & Salzberg, S. L. (2013). TopHat2: accurate alignment of transcriptomes in the presence of insertions, deletions and gene fusions. *Genome Biology, 14*(4), R36. doi:10.1186/gb-2013-14-4-r36

Kozomara, A., & Griffiths-Jones, S. (2014). miRBase: annotating high confidence microRNAs using deep sequencing data. *Nucleic Acids Research, 42*(D1), D68-D73. doi:10.1093/nar/gkt1181

Lamesch, P., Berardini, T. Z., Li, D. H., Swarbreck, D., Wilks, C., Sasidharan, R., . . . Huala, E. (2012). The Arabidopsis Information Resource (TAIR): improved gene annotation and new tools. *Nucleic Acids Research, 40*(D1), D1202-D1210. doi:10.1093/nar/gkr1090

Langmead, B., & Salzberg, S. L. (2012). Fast gapped-read alignment with Bowtie 2. *Nat Methods, 9*(4), 357-359. doi:10.1038/nmeth.1923

Li, H., Handsaker, B., Wysoker, A., Fennell, T., Ruan, J., Homer, N., . . . Genome Project Data Processing, S. (2009). The Sequence Alignment/Map format and SAMtools. *Bioinformatics, 25*(16), 2078-2079. doi:10.1093/bioinformatics/btp352

Liu, P. P., Montgomery, T. A., Fahlgren, N., Kasschau, K. D., Nonogaki, H., & Carrington, J. C. (2007). Repression of AUXIN RESPONSE FACTOR10 by microRNA160 is critical for seed germination and post-germination stages. *Plant Journal, 52*(1), 133-146. doi:10.1111/j.1365-313X.2007.03218.x

Mangan, M. E., Williams, J. M., Kuhn, R. M., & Lathe, W. C. (2014). The UCSC Genome Browser: What Every Molecular Biologist Should Know. *Current protocols in molecular biology / edited by Frederick M. Ausubel ... [et al.], 107*, 19.19.11-19.19.36. doi:10.1002/0471142727.mb1909s107

Mashiguchi, K., Tanaka, K., Sakai, T., Sugawara, S., Kawaide, H., Natsume, M., . . . Kasahara, H. (2011). The main auxin biosynthesis pathway in Arabidopsis. *Proc Natl Acad Sci U S A, 108*(45), 18512-18517. doi:10.1073/pnas.1108434108

Neufeld, S., Zinchenko, V., Stephan, D. P., Bader, K. P., & Pistorius, E. K. (2004). On the functional significance of the polypeptide PsbY for photosynthetic water oxidation in the cyanobacterium Synechocystis sp strain PCC 6803. *Molecular Genetics and Genomics, 271*(4), 458-467. doi:10.1007/s00438-004-0997-5

Park, J. E., Park, J. Y., Kim, Y. S., Staswick, P. E., Jeon, J., Yun, J., . . . Park, C. M. (2007). GH3-mediated auxin homeostasis links growth regulation with stress adaptation response in Arabidopsis. *Journal of Biological Chemistry, 282*(13), 10036-10046. doi:10.1074/jbc.M610524200

Rat Genome Browser. (2015). GBrowse-to-JBrowse Comparison.

Robinson, J. T., Thorvaldsdottir, H., Winckler, W., Guttman, M., Lander, E. S., Getz, G., & Mesirov, J. P. (2011). Integrative genomics viewer. *Nature Biotechnology, 29*(1), 24-26. doi:10.1038/nbt.1754

Sakata, S., Mizusawa, N., Kubota-Kawai, H., Sakurai, I., & Wada, H. (2013). Psb28 is involved in recovery of photosystem II at high temperature in Synechocystis sp PCC 6803. *Biochimica Et Biophysica Acta-Bioenergetics, 1827*(1), 50-59. doi:10.1016/j.bbabio.2012.10.004

Shi, L. X., & Schroder, W. P. (2004). The low molecular mass subunits of the photosynthetic supracomplex, photosystem II. *Biochim Biophys Acta, 1608*(2-3), 75-96. doi:10.1016/j.bbabio.2003.12.004

Shin, D. H., Cho, M., Choi, M. G., Das, P. K., Lee, S. K., Choi, S. B., & Park, Y. I. (2015). Identification of genes that may regulate the expression of the transcription factor production of anthocyanin pigment 1 (PAP1)/MYB75 involved in Arabidopsis anthocyanin biosynthesis. *Plant Cell Rep, 34*(5), 805-815. doi:10.1007/s00299-015-1743-7

Skinner, M. E., Uzilov, A. V., Stein, L. D., Mungall, C. J., & Holmes, I. H. (2009). JBrowse: a next-generation genome browser. *Genome Res, 19*(9), 1630-1638. doi:10.1101/gr.094607.109

Spence, A. K., Boddu, J., Wang, D., James, B., Swaminathan, K., Moose, S. P., & Long, S. P. (2014). Transcriptional responses indicate maintenance of photosynthetic proteins as key to the exceptional chilling tolerance of C(4) photosynthesis in Miscanthus × giganteus. *Journal of Experimental Botany, 65*(13), 3737-3747. doi:10.1093/jxb/eru209

Stein, L. D. (2013). Using GBrowse 2.0 to visualize and share next-generation sequence data. *Briefings in Bioinformatics, 14*(2), 162-171. doi:10.1093/bib/bbt001

Tang, G. (2010). Plant microRNAs: an insight into their gene structures and evolution. *Semin Cell Dev Biol, 21*(8), 782-789. doi:10.1016/j.semcdb.2010.07.009

Tang, G., Yan, J., Gu, Y., Qiao, M., Fan, R., Mao, Y., & Tang, X. (2012). Construction of short tandem target mimic (STTM) to block the functions of plant and animal microRNAs. *Methods, 58*(2), 118-125. doi:10.1016/j.ymeth.2012.10.006

Teotia, S., Singh, D., Tang, X. Q., & Tang, G. L. (2016). Essential RNA-Based Technologies and Their Applications in Plant Functional Genomics. *Trends in Biotechnology, 34*(2), 106-123. doi:10.1016/j.tibtech.2015.12.001

Thorvaldsdottir, H., Robinson, J. T., & Mesirov, J. P. (2013). Integrative Genomics Viewer (IGV): high-performance genomics data visualization and exploration. *Briefings in Bioinformatics, 14*(2), 178-192. doi:10.1093/bib/bbs017

Trapnell, C., Pachter, L., & Salzberg, S. L. (2009). TopHat: discovering splice junctions with RNA-Seq. *Bioinformatics, 25*(9), 1105-1111. doi:10.1093/bioinformatics/btp120

Wang, Z., Gerstein, M., & Snyder, M. (2009). RNA-Seq: a revolutionary tool for transcriptomics. *Nat Rev Genet, 10*(1), 57-63. doi:10.1038/nrg2484

Wei, L. L., Guo, J. K., Ouyang, M., Sun, X. W., Ma, J. F., Chi, W., . . . Zhang, L. X. (2010). LPA19, a Psb27 Homolog in Arabidopsis thaliana, Facilitates D1 Protein Precursor Processing during PSII Biogenesis. *Journal of Biological Chemistry, 285*(28), 21391-21398. doi:10.1074/jbc.M110.105064

Yan, J., Gu, Y., Jia, X., Kang, W., Pan, S., Tang, X., . . . Tang, G. (2012). Effective small RNA destruction by the expression of a short tandem target mimic in Arabidopsis. *Plant Cell, 24*(2), 415-427. doi:10.1105/tpc.111.094144

Yi, X. P., Hargett, S. R., Frankel, L. K., & Bricker, T. M. (2006). The PsbQ protein is required in Arabidopsis for photosystem II assembly/stability and photoautotrophy under low light conditions. *Journal of Biological Chemistry, 281*(36), 26260-26267. doi:10.1074/jbc.M603582200

Zhou, L. L., Shi, M. Z., & Xie, D. Y. (2012). Regulation of anthocyanin biosynthesis by nitrogen in TTG1-GL3/TT8-PAP1-programmed red cells of Arabidopsis thaliana. *Planta, 236*(3), 825-837. doi:10.1007/s00425-012-1674-2

A.1 Appendix

A.1.1 Copyright and Permission to Republish

The material contained in Chapter 4, "ExactSearch: A Web-based Plant Motif Search Tool" has been published in BMC Plant Methods journal under the same title. This is an open access journal; copyright statement for the publication is provided as follows.

Authors' contributions

CG, BL and HW developed the method, CG, JVRKA and SC developed the web interface, downloaded the data, and tested the tool, HW wrote the manuscript with the help of CG. All authors read and approved the final manuscript

Funding

This work was supported by grant from National Science Foundation Advances in Biological Informatics [DBI-1458130] to H.W.

A.1.2 Screenshots of the Software Applications

The screenshots shown in Figure 3.2, Figure 3.3, Figure 3.6, Figure 4.5, and Figure 4.6 were obtained from the software developed for this dissertation work. The screenshots shown in Figure 5.4, Figure 5.14, and Figure 5.15 were obtained from the STTM JBrowse software, which was developed for this dissertation work by adopting JBrowse1.11.5, an open source software. JBrowse1.11.5 is released under the GNU LGPL or the Artistic License, see the JBrowse LICENSE file.