COMPUTATIONAL METHODS FOR PROTEIN STRUCTURE PREDICTION AND NEXT-GENERATION SEQUENCING DATA ANALYSIS

A Dissertation presented to
the Faculty of the Graduate School
at the University of Missouri

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy

by
JILONG LI
Dr. Jianlin Cheng, Dissertation Supervisor
May 2016

The undersigned, appointed by the Dean of the Graduate School, have examined the dissertation entitled:

COMPUTATIONAL METHODS FOR PROTEIN STRUCTURE PREDICTION AND NEXT-GENERATION SEQUENCING DATA ANALYSIS

presented by Jilong Li,

a candidate for the degree of Doctor of Philosophy and hereby certify that, in their opinion, it is worthy of acceptance.

Dr. Jianlin Cheng

Dr. James A. Birchler

Dr. Ye Duan

Dr. Tony Han

ACKNOWLEDGMENTS

I appreciate my Ph.D. advisor, Dr. Jianlin Cheng, for imparting knowledge and enlightening research ideas.

I appreciate the co-authors of the published manuscripts included in this dissertation: Dr. Xin Deng, Dr. Jesse Eickholt, Renzhi Cao, Jie Hou, Dr. Lin Sun, Dr. Jordan M. Wilkins, Dr. Yuan Lu, Dr. Chad E. Niederhuth, Benjamin R. Merideth, Dr. Thomas P. Mawhinney, Dr. Jianlin Cheng, Dr. Valeri V. Mossine, Dr. C. Michael Greenlief, Dr. John C. Walker, Dr. William R. Folk, Dr. Mark Hannink, Dr. Dennis B. Lubahn, and Dr. James A. Birchler.

I appreciate my committee members, Dr. James A. Birchler, Dr. Ye Duan, and Dr. Tony Han, for providing suggestions and comments.

I appreciate my parents for helping me in my life.

TABLE OF CONTENTS

APPENDIX

LIST OF TABLES

Table Page

LIST OF FIGURES

Figure Page

ABSTRACT

With the wide application of next-generation sequencing technologies, the number of protein sequences is increasing exponentially. However, only a tiny portion of proteins have experimentally verified structures. The huge protein sequence-structure gap could be reduced by computational methods including template-based modeling and template-free modeling. Chapter 2 describes a stochastic point cloud sampling method for multi-template protein model generation. The stochastic sampling and simulated annealing protocol in the method has the capability to improve the global quality and reduce atom clashes in protein models.

Two popular approaches for improving protein structure prediction include enlarging the sampling space of template-based modeling and integrating template-based modeling with template-free modeling when no good templates or only partial templates can be found for a target protein. Chapters 3 and 4 introduce a large-scale conformation sampling and evaluation system for protein structure prediction which integrates the two methods.

Next-generation sequencing of RNAs (RNA-Seq) generates hundreds of millions of short reads. Analyzing these reads is increasingly being used to foster novel discovery in biomedical research. Chapter 5 describes a bioinformatics pipeline for RNA-Seq data analysis, which converts gigabytes of raw RNA-Seq data into kilobytes of valuable biological knowledge through a five-step data mining and knowledge discovery process.

Chapter 1

Introduction

This dissertation includes my research in protein tertiary structure prediction and next-generation sequencing data analysis.

Protein is the entity that physically carries out biological functions and makes the whole life system work [1]. In protein science, the key point is that protein sequence specifies protein structure and protein structure determines protein function. Therefore, understanding protein structure is important for elucidating protein function and has fundamental significance in biomedical sciences including protein design, protein engineering, genome annotation, drug design, and disease prevention strategies [2, 3].

With the wide application of next-generation sequencing technologies, the number of protein sequences is increasing exponentially. Experimental techniques such as X-ray crystallography and Nuclear Magnetic Resonance (NMR) can determine protein structure, but they are time-consuming and expensive, and cannot catch up with the pace of increasing protein sequences. Therefore, computational methods that provide a fast way of constructing an approximated structural model for a protein becomes increasingly popular and important to reduce the huge protein sequence-structure gap [1, 2, 3]. Template-based modeling (TBM) and template-free modeling (FM) are two methods for computational protein structure prediction. Template-based modeling [3, 4, 5, 6, 7, 8, 9, 10, 11] aligns the sequence of a target protein to that of another protein with known structure, and builds the target structure according to its sequence divergence with respect to the template. Template-based modeling depends largely on the availability and ability to identify homologous templates for the target as well as the sequence similarity between the target and template. The quality of the predicted models is usually low when only a relatively distant homologous template is available for the target. In this situation, template-free modeling [10, 11, 12, 13] is needed to build protein structures from scratch or from the combination of small structural fragments.

Constructing structural models for a target protein from its sequence alignment with template proteins and template structures is one main step in template-based modeling. In this step, a major challenge is how multiple templates could be integrated to generate better models than single-template. There is often noise in template conformations due to sequence divergence between target and template proteins and erroneous sequence alignments. So multiple templates may generate inconsistent or divergent conformations for the same region of target protein. Another challenge is how to build conformations for gapped regions in a target protein that are not covered by any template. Some popular methods either do not build conformations for long gaps or use a long extended chain to represent their conformations without trying to fold it.

During my PhD study, I developed a stochastic point cloud sampling method

(MTMG) for multi-template protein model generation in order to address the challenges in generating structural models from multiple templates. Chapter 2 of this dissertation describes the method. MTMG integrates two major ideas of protein comparative modeling: averaging template coordinates and using statistical approaches on spatial restraints of templates. The method uses a stochastic point cloud sampling method to sample positions for the residues with uncertain conformations (i.e. unfixed residues) based on a three-dimensional multivariate normal distribution, and uses a simulated annealing protocol to decide to accept or reject the sampled positions. The stochastic sampling and simulated annealing protocol has the capability to improve the global quality and reduce atom clashes in protein models. The content of Chapter 2 is based on a manuscript that has been published as:

Li, J. and Cheng, J. (2016) A Stochastic Point Cloud Sampling Method for Multi-Template Protein Comparative Modeling. ***Scientific Reports****, 6 (25687). [2014 Impact Factor: 5.578]*

Two popular approaches for improving the reliability and robustness of protein structure prediction include enlarging the sampling space of template-based modeling and integrating template-based modeling with template-free modeling when no good templates or only partial templates can be found for a target protein. Chapters 3 and 4 of this dissertation introduce a large-scale conformation sampling and evaluation system for protein structure prediction. The system increases the diversity of template sampling, sequence alignment sampling and model generation, combines template-based modeling and template-free modeling in order to generate a pool of protein models of good quality, and applies an array of protein model quality assessment methods to evaluate and rank the predicted models. The contents of Chapters 3 and

4 are from the manuscripts published as:

Li, J., Cao, R., and Cheng, J. (2015) A Large-Scale Conformation Sampling and Evaluation Server for Protein Tertiary Structure Prediction and its Assessment in CASP11. ***BMC bioinformatics****, 16(337). [2015 Impact Factor: 2.576]*

Li, J., Deng, X., Eickholt, J., and Cheng, J. (2013) Designing and benchmarking the MULTICOM protein structure prediction system. ***BMC structural biology****, 13(2). [2013 Impact Factor: 2.222]*

Next-generation sequencing of RNAs (RNA-Seq) is a powerful technology for transcriptome analysis [14, 15]. It can determine the relationship between the information encoded in a genome, its expression, and phenotypic variation [16, 17]. A RNA-Seq experiment typically generates hundreds of millions of short reads [18]. Analyzing these reads is increasingly being used to foster novel discovery in biomedical research. Chapter 5 describes a bioinformatics pipeline called RNAMiner for RNA-Seq data analysis. RNAMiner converts gigabytes of raw RNA-Seq data into kilobytes of valuable biological knowledge through a five-step data mining and knowledge discovery process. The experiments show that the data mining process successfully produced valuable biological knowledge and reduced the size of the initial data set over a thousand-fold. Chapter 5 is based on a manuscript that has been published as:

Li, J., Hou, J., Sun, L., Wilkins, J.M., Lu, Y., Niederhuth, C.E., Merideth, B.R., Mawhinney, T.P., Mossine, V.V., Greenlief, C.M., Walker, J.C., Folk, W.R., Hannink, M., Lubahn, D.B., Birchler, J.A., and Cheng, J. (2015) From Gigabyte to Kilobyte: A Bioinformatics Protocol for Mining Large RNA-Seq Transcriptomics Data. ***PLoS ONE****, 10(4): e0125000. [2015 Impact Factor: 3.234]*

Chapter 2

A Stochastic Point Cloud Sampling Method for Multi-Template Protein Comparative Modeling

2.1 Abstract

Generating tertiary structural models for a target protein from the known structure of its homologous template proteins and their pairwise sequence alignment is a key step in protein comparative modeling. Here, we developed a new stochastic point cloud sampling method, called MTMG, for multi-template protein model generation. The method first superposes the backbones of template structures, and the Cα atoms of the superposed templates form a point cloud for each position of a target protein, which are represented by a three-dimensional multivariate normal distribution. MTMG stochastically resamples the positions for Cα atoms of the residues whose positions are uncertain from the distribution, and accepts or rejects new position according to a simulated annealing protocol, which effectively removes atomic clashes commonly encountered in multi-template comparative modeling. We benchmarked MTMG on 1,033 sequence alignments generated for CASP9, CASP10 and CASP11 targets, respectively. Using multiple templates with MTMG improves the GDT-TS score and TM-score of structural models by 2.96-6.37% and 2.42-5.19% on the three datasets over using single templates. MTMG's performance was comparable to Modeller in terms of GDT-TS score, TM-score, and GDT-HA score, while the average RMSD was improved by a new sampling approach. The MTMG software is freely available at: http://sysbio.rnet.missouri.edu/multicom_toolbox/mtmg.html.

2.2 Introduction

The tertiary structure of a protein, which can be represented by the coordinates of its atoms, is important for understanding the function and activity of the protein [1, 2]. Experimental techniques such as x-ray crystallography and nuclear magnetic resonance (NMR) can determine protein tertiary structure, but they are time-consuming and expensive, leading to a large gap between the number of known protein sequences (∼100 million) and the number of known protein structures (∼100,000). Therefore, computational protein structure modeling that provides a fast way of constructing an approximated structural model for a protein becomes increasingly popular and important [3].

Computational protein modeling methods are usually classified into two categories: template-based modeling (TBM) that uses known protein structures as templates [3, 4, 5, 6, 7, 8, 9, 10, 11, 19, 20] and template-free modeling (FM) that tries to build models from scratch without referring to any known structure [11, 12, 13]. Template-based modeling constructs protein structures by comparing a target sequence to template sequences in order to find homologous templates with known structures, and then transfer the template structures to the target protein through comparative modeling [3]. Using multiple templates, if available, generally improves the quality of models over using single template as demonstrated in the past Critical Assessments of Techniques for Protein Structure Prediction [21, 22, 23], even though dealing with multiple templates that have some conflicting structural conformations is more difficult than handling a single template.

Generating protein 3D models for a target protein from its sequence alignment with template proteins and template structures is one of the most challenging steps in template-based modeling. Several methods were developed to address this problem, such as Modeller [24, 25], SWISS-MODEL [26], ModSeg/ENCAD [27], NEST [28], etc. But all of these tools were initially developed more than a decade ago. SWISS-MODEL builds the core of a model by averaging the coordinates of backbone atoms in template structures, and uses the constraint space programming to construct the conformation of gaps, the region of a target protein not covered by any template [26]. ModSeg/ENCAD uses the segment match modeling to build models for a target protein [27]. NEST iteratively inserts/deletes one residue into/from template structure in order to build the whole model for target protein by minimizing the energy [28]. Modeller [24, 25] extracts spatial restraints from target-template alignments and template structures, and builds models for target protein by minimizing the restraint violations. Modeller, initially developed more than 20 years ago, is still the most widely used tool for generating structural models from target-template sequence alignments and template structures. The multiple-template threading method in Dr. Xu's group uses a novel probabilistic-consistency algorithm to improve protein 3D modeling by accurately aligning a single protein sequence simultaneously to multiple templates [19]. The method can build protein models with better quality than single-template models even if the models are built from the best single template [19]. The probabilistic multi-template protein homology modeling method [20] computes improved spatial restraints and calls Modeller to build 3D models. The method uses two-component Gaussian mixture distributions to combine density functions by multiplication compared to Modeller's one-component densities. It also proposes new algorithms for computing template weights and selecting templates. Despite the importance of model generation, few new methods have been proposed to address some unsolved challenges associated with it.

One major challenge in comparative model generation is to integrate multiple templates to generate better models than single-template, which is particularly challenging when multiple templates suggest inconsistent or divergent conformations for the same region for target protein [29]. Another challenges including how to handle the noise in template conformations due to sequence divergence between target and template proteins and erroneous sequence alignments, and how to build conformations for gapped regions in a target protein that are not aligned with any template. Gaps may appear either in the middle of a target protein or at its terminals. Several popular methods such as SWISS-MODEL or Modeller either do not build conformations for long gaps or use a long extended chain to represent their conformations without trying to fold it.

In this study, we developed a new stochastic point cloud sampling method for

Multi-Template Model Generation (MTMG) to address the challenges in generating structural models from multiple templates. Different than Modeller that extracts pairwise distance restraints for pairs of atoms or angular restraints of individual residues from templates, our method generate positional (x, y, z coordinate) restraints (point cloud) for each residue from templates centered on the weighted average structure of superimposed template structures. The point cloud is represented by a three-dimensional multivariate distribution, whose variance measure how uncertain the position of a residue is. The position of residue with low variance is generally fixed and that of residues with high variance is resampled from the distribution. A stochastic resampling move is rejected or accepted according to a simulated annealing protocol based on the RW protein energy function [30] and the spatial distance restraints. The resampled model with the lowest energy is selected as the final model for the target protein.

We benchmarked MTMG on 1,033 sequence alignments generated for hundreds of targets used in the 9^{th}, 10^{th} and 11^{th} Critical Assessment of Techniques for Protein Structure Prediction (CASP9, CASP10 and CASP11). MTMG of using multiple templates performed significantly better than using single template. Its performance is also comparable to the state-of-the-art model generation tool - Modeller - in terms of GDT-TS score (Global Distance Test Total Score) [31], TM-score (Template Modeling score) [32], and GDT-HA score (Global Distance Test High Accuracy score) [33, 34]. In terms of RMSD (Root Mean Square Deviation), MTMG performs substantially better than Modeller by folding long gaps in target protein better.

2.3 Results

In this section, we first briefly describe the sampling method of MTMG, and then present an evaluation of its performance from various perspectives.

2.3.1 The sampling method of MTMG

Given a sequence alignment involving a target protein and one or multiple template proteins and the tertiary structures of the templates as input, MTMG first removes structurally inconsistent templates in order to decrease structural noise. The remaining templates are structurally superposed together, and the weighted average coordinates are calculated for each residue of the target. It then uses a stochastic point cloud sampling method to sample positions for the residues with uncertain conformations (i.e. unfixed residues) based on a three-dimensional multivariate normal distribution. The sampled positions iteratively replace the coordinates of the unfixed residues according to a simulated annealing protocol. A model with the lowest energy is selected as final prediction. The details of this modeling process are described in the Methods section.

2.3.2 Benchmark datasets

In order to rigorously evaluate our method, we benchmark MTMG on the three datasets, i.e. 104, 87, and 79 official targets in the 9^{th}, 10^{th} and 11^{th} Critical Assessment of Techniques for Protein Structure Prediction (CASP9, CASP10 and CASP11), separately. In total these datasets have 1,033 target-template sequence alignments generated by different alignment tools used with our MULTICOM protein structure prediction server [35, 36, 37, 38, 39, 40]. CASP9, 10, 11 datasets have 398, 313, and 322 target-template sequence alignments, respectively, which were generated by HHSearch (versions 1.2 and 1.5) [41], HMMer [42], and CSI-BLAST [43] separately under same condition. HHSearch [41] is a profile-profile alignment tool. HMMer [42] is a profile-sequence alignment tool based on profile hidden Markov models. CSI-BLAST [43] is a tool for iterative search of homologues with position-specific scoring matrices. The pairwise alignments between a target and each of multiple templates produced by these tools were combined into multiple sequence alignments [35, 36, 37, 38, 39] in order to use multiple templates if exist. At the end, 299 (75.13%) CASP9 alignments, 243 (77.64%) CASP10 alignments, and 259 (80.43%) CASP11 alignments contain multiple templates. Figure 2.1 shows the distribution of sequence identity in the sequence alignments. Most top template sequences have sequence identity of ~20% with the target sequence, and the average sequence identity is 34.23%. The datasets are sufficiently large and contain diverse types of sequence alignments and targets, making them good datasets to objectively benchmark our method. The sequence alignments, template structures, and predicted models in this study are available at http://sysbio.rnet.missouri.edu/multicom_toolbox/mtmg.html.

We ran MTMG and a state-of-the art comparative modeling tool - Modeller on the three CASP datasets to predict 3D structures for the CASP targets in order to compare their performance. The default approach in Modeller was used to generate 25 structural models for each sequence alignment. The model with lowest DOPE score calculated by Modeller was used for benchmark. MTMG was also run on each single template to generate single template-based models in order to study if and how multiple templates may improve modeling performance. The predicted models were superposed with true structures to calculate GDT-TS score, TM-score, GDT-HA score, and RMSD using the TM-score program [32].

2.3.3 Multiple templates versus single template

We compared the models predicted by MTMG on multiple templates and on the first single template selected by each alignment tool. Table 2.1 reports the average GDT-TS score, TM-score, GDT-HA score, and RMSD of the models based on the first single template and on multiple templates on CASP9, CASP10, and CASP11 targets. The results show that using multiple templates improved GDT-TS score (or TM-score) by 6.37%, 3.65%, and 2.96% (or 5.19%, 3.15%, and 2.42%) over using first single template on the three data sets separately. The average RMSD was also obviously improved by using multiple templates.

Dataset	Template	GDT-TS	TM	GDT-HA	RMSD
CASP11	Single	0.3906	0.4468	0.2739	16.16
	Multiple	0.4155	0.4700	0.2951	14.80
CASP10	Single	0.5366	0.5749	0.4111	11.09
	Multiple	0.5562	0.5930	0.4314	9.90
CASP9	Single	0.5370	0.5827	0.3851	10.61
	Multiple	0.5529	0.5968	0.3985	9.83

Table 2.1: The average GDT-TS score, TM-score, GDT-HA score, and RMSD of the models predicted by MTMG using the first single templates and multiple templates on CASP9, CASP10, and CASP11 targets.

Table 2.2 reports the p-values of t-test [44] on GDT-TS score and TM-score for comparison between single template and multiple templates on CASP9, CASP10, and CASP11 targets. All the p-values are < 0.05, indicating that using multiple templates can significantly improve the global quality of the predicted models in terms of GDT-TS score and TM-score over using single template, which is the generally the alignment with the lowest e-value calculated by a sequence alignment tool.

Dataset	p-value of GDT-TS score	p-value of TM-score
CASP11	2.355e-13	3.356e-12
CASP10	2.423e-07	1.087e-06
CASP9	2.227e-08	2.745e-07

Table 2.2: The p-values of t-test on GDT-TS score and TM-score for the comparisons between using first single templates and using multiple templates on CASP9, CASP10, and CASP11 targets.

We investigated improvements or losses of GDT-TS score, TM-score, GDT-HA score, and RMSD on individual targets. Figure 2.2 shows scatter plots of GDT-TS scores, TM-scores, GDT-HA scores, and RMSDs between the single-template models and the multiple-template models on CASP11 targets. From the figure, using multiple templates improves the predicted models on all the scores. Since the first single template is considered the most relevant (or significant) template selected by each sequence alignment tool, the results suggest using multiple template consistently improve the quality of comparative modeling over using the top one single template selected by an alignment tool. The results are consistent with the previous studies [21, 22, 23].

We chose 51 CASP11 targets (73 domains), which were covered by at least three templates, in order to compare the multi-template models with all possible single-template models, i.e., the models generated by every single template in the alignments. These targets have between 3 and 41 templates. The minimum, maximum, median, quartile at 25 percentile, and quartile at 75 percentile of GDT-TS score of single-template models were calculated. Figure 2.3 shows the GDT-TS score of the predicted models on 73 CASP11 domains using single templates and multiple templates. The multi-template models have higher GDT-TS score than median of the GDT-TS scores of the single template models on 72 domains. The average improvement of GDT-TS score is 0.1188. The results illustrate that using multiple templates can improve the quality of the predicted models substantially. Furthermore, the multi-template models have higher GDT-TS score than the best models built from the best possible single template on 30 domains. The average difference of GDT-TS score between multi-template models and best-template models is almost zero (i.e., -0.0065) and the most significant improvement is 0.1448, indicating that using multiple templates yields the similar performance with using the best single template. Since it is impossible to select the best template for each target without knowing the true structures of the target most time, using multiple templates is the practical way to achieve the best potentials within template structures.

2.3.4 MTMG versus Modeller

We compared MTMG with Modeler on CASP9, CASP10, and CASP11 targets. Table 2.3 reports the average GDT-TS score, TM-score, GDT-HA score, and RMSD of the models predicted by MTMG and Modeller on CASP9, CASP10, and CASP11 targets, respectively. The average GDT-TS score and TM-score on CASP11 targets and the average TM-score on CASP10 targets of MTMG are slightly higher than that of Modeller, while the average GDT-TS score on CASP10 targets and the average GDT-TS score and TM-score on CASP9 targets of MTMG were slightly lower than those of Modeller. One the super dataset that combine all CASP9, CASP10, and CASP11 datasets together, the average GDT-TS score and TM-score of MTMG is 0.5121, 0.5570, which is similar to 0.5136 and 0.5543 of Modeller. Overall, the performance of

MTMG is comparable to Modeller in terms of GDT-TS score and TM-score. However, in terms of RMSD, MTMG performed better than Modeller. According to Table 2.3, the average RMSD of MTMG was 5.96Å, 13.29Å, and 3.91Å lower than that of Modeller on the three datasets, respectively. The reason of the improvement is that MTMG models long gaps in target proteins that are not covered by templates better than Modeller. MTMG and Modeller use similar approaches to model the unaligned regions which no template covers. Both of them loop the unaligned regions out into space. MTMG uses the spatial restraints to sample conformations for long gaps and chooses the angles at random. Modeller chooses always same angle and usually generates an unfolded stick for a long gap. Modeller's method directly shows where no alignment information exists. In the contrast, our method tries to make a reasonably folded conformation for the long gap. This may be a reason that the RMSD of the unaligned regions in MTMG models is averagely lower than that in Modeller models.

Dataset	Method	GDT-TS	TM	GDT-HA	RMSD
CASP11	Modeller	0.4150	0.4641	0.2967	20.76
	MTMG	0.4155	0.4700	0.2951	14.80
CASP10	Modeller	0.5584	0.5889	0.4361	23.19
	MTMG	0.5562	0.5930	0.4314	9.90
CASP9	Modeller	0.5569	0.5992	0.4034	13.74
	MTMG	0.5529	0.5968	0.3985	9.83

Table 2.3: The average GDT-TS score, TM-score, GDT-HA score, and RMSD of the models predicted by MTMG and Modeller on CASP9, CASP10, and CASP11 targets.

We compared the improvements or losses of GDT-TS score, TM-score, GDT-HA score, and RMSD of MTMG and Modeller on the individual targets of the three datasets. Figure 2.4 shows the scatter plots of GDT-TS score, TM-score, GDT-HA score, and RMSD between the MTMG models and the Modeller models on CASP11 targets. The figure shows that the performance of MTMG is comparable to that of Modeller in terms of GDT-TS score, TM-score and GDT-HA score, while RMSD of MTMG is significantly lower than that of Modeller.

We also compared our method with the probabilistic multi-template protein homology modeling method [20]. The probabilistic multi-template method relies on HHSearch's output (.hhr) to generate spatial restraints and calls Modeller to build 3D models. In the contrast, our method is an independent method and doesn't rely on outputs of any specific tools and any model generation tools. Table 2.4 shows average GDT-TS score, TM-score, GDT-HA score, and RMSD of the models predicted by MTMG and the probabilistic multi-template method based on HHSearch's outputs on CASP9, CASP10, and CASP11 targets. From the table, the performance of the probabilistic multi-template method is better than that of our method on GDT-TS score, TM-score, and GDT-HA score. However, our method improved RMSD over the method. We didn't do more comparisons between our method and the probabilistic multi-template method because they accept different kinds of input files.

Dataset	Method	GDT-TS	TM	GDT-HA	RMSD
CASP11	MTMG	0.4256	0.4801	0.3010	11.62
	the probabilistic multi-template method	0.4322	0.4797	0.3112	14.42
CASP10	MTMG	0.5478	0.5897	0.4146	9.57
	the probabilistic multi-template method	0.5604	0.5957	0.4281	13.96
CASP9	MTMG	0.5536	0.5953	0.3970	9.20
	the probabilistic multi-template method	0.5669	0.6062	0.4128	12.45

Table 2.4: The average GDT-TS score, TM-score, GDT-HA score, and RMSD of the models predicted by MTMG and the probabilistic multi-template protein homology modeling method based on HHSearch sequence alignments on CASP9, 10 and 11 targets.

We compared MTMG with Modeller on CASP11 targets with different template coverage in order to elucidate how they performed differently. Figure 2.5a shows the comparison of GDT-TS score between the MTMG models and the Modeller models with different template coverage. X-axis represents template coverage, and y-axis represents the average GDT-TS score. According to the results, MTMG performed better than Modeller on targets with $<$ 0.7 (e.g. 70%) template coverage, but slightly worse than Modeller on targets with $>=$ 0.7 template coverage. The improvement on targets with lower template coverage by MTMG was partially due to its capability of sampling the conformation of long gaps.

We checked how the number of templates might affect the performance of MTMG and Modeller. Figure 2.5b compares the GDT-TS score between the MTMG models and the Modeller models constructed from different numbers of templates. X-axis represents the number of templates, and y-axis represents the average GDT-TS score. The results show that, MTMG performed better than Modeller on targets covered by $<$ 10 templates, while it performed worse than Modeller on targets covered by $>=$ 10 templates.

We further compared the performance of MTMG with Modeller on CASP11 targets containing different numbers of domains. Figure 2.5c reports the GDT-TS score of the MTMG models and the Modeller models for targets with different numbers of domains. X-axis represents the number of domains, and y-axis represents the average GDT-TS score. MTMG performed better than Modeller on targets containing multiple domains. The results suggest that the domain division and combination protocol used by MTMG can improve the quality of modeling multi-domain proteins.

We classified CASP9, 10 and 11 targets into four SCOP protein categories: all-alpha, all-beta, alpha/beta, and alpha+beta. Table 2.5 shows average GDT-TS score of MTMG and Modeller models on four protein categories on CASP targets. From the table, the class of the proteins doesn't have an obvious impact on the quality and improvement of MTMG models.

Dataset	Category	Modeller	MTMG
CASP11	all-alpha	0.3114	0.3195
	all-beta	0.4388	0.4429
	alpha/beta	0.6532	0.6392
	alpha+beta	0.3221	0.3255
CASP10	all-alpha	0.5907	0.5943
	all-beta	0.4526	0.4542
	alpha/beta	0.6670	0.6565
	alpha+beta	0.5894	0.5836
CASP9	all-alpha	0.5109	0.5152
	all-beta	0.4522	0.4496
	alpha/beta	0.6884	0.6777
	alpha+beta	0.5992	0.5923

Table 2.5: The average GDT-TS score of the models predicted by MTMG and Modeller on four protein categories on CASP9, 10, and 11 targets.

We divided CASP9, 10 and 11 targets into easy, medium, and hard targets according to CASP official classification. The average GDT-TS score and RMSD were calculated for different kinds of targets as shown in Table 2.6. From the table, RMSD was improved by our method on any kinds of targets. GDT-TS score was improved by our method on medium and hard targets. For easy targets, the performance of our method on GDT-TS score was a little bit worse than that of Modeller.

We also analyzed GDT-TS score of MTMG and Modeller models on different protein lengths for CASP9, 10, and 11 targets. The analysis results are shown on Figure 2.6. Red points donate GDT-TS scores of MTMG models, and blue points donate GDT-TS scores of Modeller models. The figure shows that GDT-TS scores

Dataset	Classification	GDT-TS		RMSD	
		Modeller	MTMG	Modeller	MTMG
CASP11	TBM	0.5418	0.5385	11.04	9.76
	TBM-hard	0.1921	0.1977	35.56	23.6
	FM	0.1476	0.1578	41.23	24.81
CASP10	TBM	0.5824	0.5792	11.67	9.24
	FM/TBM	0.3181	0.3186	40.84	14.57
	FM	0.3099	0.3195	47.52	17.36
CASP9	TBM	0.6382	0.6338	8.85	6.86
	FM/TBM	0.1850	0.1865	27.29	21.37
	FM	0.1844	0.1861	36.47	23.53

Table 2.6: The average GDT-TS score and RMSD of the models predicted by MTMG and Modeller for different kinds of CASP targets.

of MTMG and Modeller models are mostly similar on different protein lengths. It implies that the length of the proteins doesn't have a critical impact on our method.

In addition to comparing MTMG and Modeller in terms of global backbone quality scores such as GDT-TS score, TM-score, and RMSD, we compare them in terms of MolProbity score that measures the "realistic" level of models. MolProbity is a knowledge based metrics that evaluates the physical reasonableness of molecular models [45]. The models generated by our method have higher average MolProbity scores (i.e., worse local quality) than Modeller. For example, the average MolProbity score of the MTMG models is 3.51, which is higher than 3.02 of the Modeller models on CASP11 targets. The problem may be caused by the way used by MTMG to convert the reconstructed Cα trace into a full backbone. A solution to improve the MolProbity score is to use ModRefiner [46] to generate main-chain and side-chain atoms from Cα trace instead of using Pulchra. According to our experiment on CASP11 targets, with a very slight decrease in GDT-TS score by 0.008, the average MolProbity score of MTMG models could be improved to 2.90 by ModRefiner, which is better than the average MolProbity score of the Modeller models. Therefore, if necessary, users can run ModRefiner on MTMG's models to generate full-atom models with good MolProbity scores.

2.3.5 The impact of simulated annealing protocol

MTMG uses simulated annealing to iteratively generate new models with sampled points. We investigate how it can improve the quality of models. Figure 2.7 shows the changes of TM-score (a) and the number of clashes (b) of two CASP11 targets with respect to iterations during simulated annealing. The plots show that TM-score stochastically went up and down with an overall upward trend during simulated annealing. Even though the final model was not the best one, but it was close to the best one and better than the initial model. Moreover, the number of clashes rather consistently decreased during simulated annealing.

2.3.6 Several case studies

We studied several cases on which MTMG performed better than Modeller to demonstrate how MTMG improved the quality of modeling. Figure 2.8a shows the structural superposition between the native structure (blue) of target T0841 and the models predicted by Modeller (gold) and MTMG (purple). The GDT-TS scores of the Modeller model and the MTMG model were 0.8524 and 0.9058 respectively. The target protein has only one domain, covered by 78 significant templates with e-values of alignment equal to 0. The structural inconsistency within 78 templates may have reduced the quality of Modeller models. After removing inconsistent templates by structural superposition, MTMG chose 10 templates to construct models leading to better quality.

Figure 2.8b illustrates the structural superposition between the native structure (blue) of target T0847 and the models predicted by Modeller (gold) and MTMG (purple). The GDT-TS scores of the Modeller model and the MTMG model were 0.5258 and 0.6553, respectively. MTMG improved GDT-TS score by 0.1295. The region (residues 148 - 168) of the MTMG model circled by red is superposed with the native structure much better than that of the Modeller model circled by red. The target protein is covered by two templates 1BYRA and 4GGKA, which cover 10 common residues (residues 129 - 138) of the target. The circled region is covered by 4GGKA. MTMG superposed 4GGKA against 1BYRA in order to align them to the correct location. The result shows that the MTMG's process of template superposition successfully aligns the templates together to improve the quality of modeling.

Figure 2.8c compares the native structure of domain T0845-D2 (blue) with the Modeller model (gold) and the MTMG model (purple). The CASP11 target T0845 has two domains: T0845-D1 (residues 23 - 119) and T0845-D2 (residues 120 - 448). In the sequence alignment, the target protein is covered by two templates 3TC9A and 3DSMA. 3TC9A covers residues 34 - 112 and 3DSMA covers residues 122 - 448. But the two templates do not cover any common residues. So, MTMG divided the target protein into two domains: D1 (residues 1 - 119) and D2 (residues 120 - 448). The two domains were modeled separately, and the predicted models of the two domains were combined into a full-length model using the moving and rotating algorithm. The GDT-TS scores of the Modeller model and the MTMG model were 0.3574 and 0.4985, respectively. MTMG improved GDT-TS score by 0.1411 on domain T0845-D2. For domain T0845-D1, the GDT-TS scores of the Modeller model and the MTMG model were 0.3262 and 0.3582 separately. MTMG improved GDT-TS score by 0.032. In this case, MTMG's domain division and combination protocol improved the quality of modeling.

2.3.7 Running time

We investigated the running time of our method in the experiment. MTMG was run on single CPU with the x86_64 Red Hat Linux system. Figure 2.9 shows the number of targets in the different ranges of running time on CASP9, CASP10, and CASP11 targets. The running time of 92.83% of targets is <= 10 minutes, which is reasonably fast. The minimum running time on three datasets is 1 second. The average running time on CASP9, CASP10, and CASP11 targets is 2'22", 2'28", and 3' separately. The maximum running time on these datasets is 44'38", 56'58", and 56'44" separately. The speed is related to template similarity, target length, the number of templates, and the number of gaps. The minimum running time occurred on targets with single template or structurally very similar templates, and good template coverage. The maximum running time occurred on long targets with many unfixed residues and/or gaps. We also tested the speed of Modeller on same condition. Modeller usually spent a few seconds to several minutes to build a structural model. Although our method is fast, it is a little bit slower than Modeller on average.

2.4 Discussion

In this study, we designed and implemented a new stochastic point cloud sampling method for multi-template protein model generation (MTMG) in comparative modeling. The stochastic sampling and simulated annealing protocol in MTMG has the capability to improve the global quality and reduce atom clashes in models. Some new techniques are developed to improve modeling, including the template superposition and weighting for removing structural inconsistency and considering the relevance of templates, domain division and combination for integrating overlapped templates, and moving and rotation algorithm for loop (gap) modeling. Our extensive experiment on three CASP datasets clearly demonstrates that using multiple templates significantly improves the performance of comparative modeling over using first / average single templates, and the performance of using multiple templates is comparable to the idea case of using the best possible templates available. On the same benchmark, the performance of MTMG is comparable to that of a state-of-the-art method Modeller. The difference in the performance of MTMG and Modeller is related to the template coverage, the number of templates, and the number of domains of a target protein. Overall, MTMG is a new, complementary, and useful addition to the important, yet under-developed tool set for protein comparative modeling.

2.5 Methods

2.5.1 The stochastic point cloud sampling method (SPC) for sampling conformations from multiple templates

Figure 2.10 illustrates the workflow of the stochastic point cloud (SPC) sampling method for sampling conformations for a target protein covered by multiple templates. For a target protein covered only by a single template, the backbone of the template structure is copied directly to the predicted model without invoking the point cloud method. Otherwise, it works as follows.

The SPC method extracts a set of coordinates (x, y, z coordinates) of Cα atoms for residues in a target protein from the template structures according to target-template alignments if exists. For a Cα atom, it calculates the weighted average coordinates (i.e. the central coordinates) of the Cα atoms in multiple templates, the distance between the position of a Cα atom in each template and the weighted average coordinates, and the weighted average distance between Cα atoms in multiple templates. The set of coordinates of Cα extracted from the template structures define a weighted point cloud centered at the weighted average coordinates for each residue. The weighted average coordinates are calculated using equation (2.1):

$$p\_avg_i = \frac{1}{sw_i} \sum_{j=1}^{n_i} (p_{ij} * w_j). \tag{2.1}$$

where p_avg$_i$ is the weighted average coordinates of the ith residue, sw$_i$ is sum of weights of templates covering the ith residue, n$_i$ is the number of templates covering the ith residue, p$_{ij}$ is coordinates (x$_{ij}$, y$_{ij}$, z$_{ij}$) of Cα of the ith residue in the jth template, and w$_j$ is the weight of the jth template.

The weighted average distance is calculated using equation (2.2):

$$wad_i = \frac{1}{sw_i} \sum_{j=1}^{n_i} (d_{ij} * w_j). \tag{2.2}$$

where wad_i is the weighted average distance of the i^{th} residue, sw_i is sum of weights of templates covering the i^{th} residue, n_i is the number of templates covering the i^{th} residue, d_{ij} is the distance between p_avg_i (weighted average coordinates) and point_{ij}, and w_j is the weight of the j^{th} template.

The weighted average coordinates of Cα atoms are used as the initial model to be optimized. A global energy score for the initial model is calculated by the RW potential function [30] and is denoted as E_{old}. RW is a pairwise distance-dependent, atomic statistical potential function [30].

For a residue, if its weighted average distance is > 0.5Å, the coordinates in the point cloud are considered significantly varied. Such a residue is called an unfixed residue whose conformation is largely uncertain or different in multiple templates and needs to be resampled. SPC samples the positions for unfixed residues using a three-dimensional multivariate normal distribution in order to find a better position to replace the old one. The probability density function of the d-dimensional multivariate normal distribution is given using equation (2.3):

$$y = f(x, \mu, \Sigma) = \frac{1}{\sqrt{\left|\Sigma\right| (2\pi)^d}} e^{-\frac{1}{2}(x-\mu)'\Sigma^{-1}(x-\mu)}. \tag{2.3}$$

where μ represents the 1-by-d mean vector (i.e., weighted average coordinates), Σ represents the d-by-d covariance matrix (i.e., weighted average distance), and x represents a 1-by-d random variable (sampled point or position). Σ is calculated using equation (2.4):

$$\Sigma = \begin{bmatrix} wad & 0 & 0 \\ 0 & wad & 0 \\ 0 & 0 & wad \end{bmatrix} \tag{2.4}$$

where wad represents weighted average distance. The diagonal elements of Σ contain the variances for each variable (i.e. each coordinate), which is set approximately by the weighted average distance by default. The off-diagonal elements of Σ contain the covariance between variables, which are set to 0 assuming there is no covariance between variables.

The multivariate normal distribution provides an effective, density-based clustering method for sampling points from a 3D space defined by the point cloud of the Cα atom of a residue [47]. SPC uses the function mvrnorm in R package 'MASS' [48]. The function is called to sample 100 points for each unfixed residue in each iteration.

The sampled points are evaluated before being accepted or rejected (see Figure 2.11). One sampled point is rejected if it causes a broken chain (the distance between two adjacent Cα atoms $>$ 4.5Å) or atom-atom clashes (the distance between two Cα atoms $<$ 3.5Å), otherwise is accepted. If a sampled point is accepted or all the sampled points are rejected within 100 tries, SPC will move to the next unfixed residue to sample its positions until the positions of all the unfixed residues are resampled. The accepted sampled points replace the current coordinates of the respective unfixed residues in the current model to generate a new model. A global energy score is calculated by RW [30] for the new model as E_{new}.

After a new model is sampled, SPC uses simulated annealing to decide whether the new model is accepted (or kept). Simulated annealing is a stochastic optimization technique to minimize or maximize an objective function [49, 50], i.e. to find the optimal model with minimum energy. The initial temperature is the number of iterations, which equals to 1,000 divided by the number of the unfixed residues, and is constrained in between 20 and 100 by default. If a number between 1 and 500 is given while running MTMG, the number of iterations equals to the number of unfixed residues multiplying the given number. The temperature decreases from iteration by iteration. If E$_{new}$ is less than E$_{old}$, the new model is accepted, E$_{old}$ is set to E$_{new}$, and the temperature (T) decreases by 1. If E$_{new}$ is greater than or equals to E$_{old}$, a probability of accepting the new model is calculated as e$^{\frac{E_{old}-E_{new}}{T}}$. If the probability is $>$ 0.5, the new model is accepted, and otherwise it is rejected. And the temperature decreases by d, which is calculated using equation (2.5):

$$d = (E_{new} - E_{old}) * 0.2 * (1 + \frac{1}{N})^j \quad (j = 1, ..., N). \tag{2.5}$$

where N is the number of iterations. E$_{new}$ - E$_{old}$ makes that d is a positive number because E$_{new}$ is larger than E$_{old}$. We considered 5 as a basic difference between E$_{new}$ and E$_{old}$, so we multiplied it with 0.2 in order to make it around 1. $(1 + \frac{1}{N})^j$ is getting bigger during iterations, which is consistent with simulation annealing.

The sampling process stops when the maximum number of iterations is reached or temperature drops to 0 or below. The last accepted model by the end of sampling is the final model predicted for a target.

The modeling method discussed above can be used directly to build models in most cases when a target is globally covered by at least one template or by multiple overlapped templates. For a target whose regions are only covered locally by different templates, we added one step of domain division and combination to join the conformations of regions covered by different templates into a full-length model. Figure 2.12a shows a target whose two domains are covered by five templates without overlapped linkers to join them together. The left region (domain 1) of the target is covered by templates T1 and T2, and the right region (domain 2) is covered by T3, T4, and T5. But the two regions are not covered by any common template. In this case, we divide the target protein into two domains and model them separately using the SPC method discussed above.

After the two domains are modeled, we use a moving rotation algorithm to combine the models of domains into a full-length model by iteratively combining models of two adjacent domains. The algorithm has the following six steps: (1) getting the coordinates of the last residue (A) in the first domain D1 and the coordinates of the first residue (B) in the second domain D2, and calculating the distance between A and B as d_{AB}; (2) if d_{AB} is too small or too big for two adjacent residues ($d_{AB} < 3.5$Å or $d_{AB} > 4.5$Å), moving (translating) D2 to a new location so that d_{AB} is between 3.5Å and 4.5Å; (3) if there are less than 15 atom clashes between D1 and D2, no severe clashes and no broken chains, the algorithm exits, and otherwise continue to the following steps; (4) sampling points (positions) around A in order to find a point C so that the distance between A and C (d_{AC}) is between 3.5Å and 4.5Å and there are no atom clashes between D1 and C; (5) moving (translate) D2 so that B and C are in the same position; and (6) iteratively rotating D2 in order to find its orientation where the combined model has no/least atom clashes. The rotation is implemented by equation (2.6):

$$\begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = \begin{bmatrix} R_{11} & R_{12} & {\scriptstyle R13} \\ R_{21} & R_{22} & R_{23} \\ R_{31} & R_{32} & R_{33} \end{bmatrix} * \begin{bmatrix} x \\ y \\ z \end{bmatrix} \tag{2.6}$$

Given a unit vector u = (u$_x$, u$_y$, u$_z$), where u$_x^2$ + u$_y^2$ + u$_z^2$ = 1, the matrix (R) used for rotation by an angle θ around an axis in the direction of u is calculated using equation (2.7) [51]:

$$R = \begin{bmatrix} cos\theta + u_x^2(1 - cos\theta) & u_x u_y(1 - cos\theta) - u_z sin\theta & u_x u_z(1 - cos\theta) - u_y sin\theta \\ u_y u_x(1 - cos\theta) - u_z sin\theta & cos\theta + u_y^2(1 - cos\theta) & u_y u_z(1 - cos\theta) - u_x sin\theta \\ u_z u_x(1 - cos\theta) - u_y sin\theta & u_z u_y(1 - cos\theta) - u_x sin\theta & cos\theta + u_z^2(1 - cos\theta) \end{bmatrix} \tag{2.7}$$

Different u$_x$, u$_y$, and u$_z$ were drawn from the range of [-1, 1] and they changed by 0.1 for each iteration.

2.5.2 Template weighting and combination

Before using the structures of the templates in model sampling as described above, all the templates are preprocessed in order to make them structurally comparable and consistent as follows. Residues in templates that do not cover the target protein are removed from the sequence alignment and template structures. The remaining residues and atoms are re-indexed according to their sequence alignments with the target protein.

The quality of the model constructed for the target protein depends on the quality and relevance of selected templates, such as sequence similarity between the target and templates and template coverage. Using multiple complementary templates can often reduce modeling variance and increase template coverage leading to better models, but low-quality templates may decrease the quality of models. Therefore, we assign a weight to a template to control its impact on calculating the average coordinates or the variance of point cloud for a residue. The weight of a template is the sum of five terms: average TM-score, template coverage, sequence identity, sequence similarity, and $\frac{1}{e^{e-value}}$, which are described as follows: (1) Average TM-score. Each template is aligned with other templates by the TM-score program [32] and a pairwise TM-score between 0 and 1 is calculated for each pair of templates. The average TM-score [31] is calculated for each template. (2) Template coverage. It is the ratio between the number of residues covered by the template and the length of the target. (3) Sequence identity. It is the ratio between the number of identical residues between the target and the template in target-template alignment and the total number of target residues covered by the template. (4) Sequence similarity. The similarity score is calculated for each pair of residues in the target-template alignment using BLOSUM62 Matrix [52]. If the score is $<$ 0, score is set to e ^{score}; otherwise score is set to score plus 1. The sequence similarity for the template is calculated using equation (2.8):

$$\frac{1}{n}\sum_{j=1}^{n}\frac{score_j}{12}. \tag{2.8}$$

where n is the total number of target residues covered by the template (template coverage length), and score$_j$ is the similarity score of the jth residue (1 <= j <= n). Score$_j$ is divided by 12 in order to scale it to the range [0, 1]. (5) $\frac{1}{e^{e-value}}$. An e-value measuring the significance of an alignment score is extracted from the sequence alignment for each target-template alignment. If an alignment does not provide e-value, an identical e-value (i.e. 0) is assigned to all the templates. $\frac{1}{e^{e-value}}$ ranges between 0 and 1. Lower e-value (i.e. a more significant alignment score) leads to a larger $\frac{1}{e^{e-value}}$.

After the weights for all the templates are calculated, the template with the highest weight is first selected. For each of other candidate templates, if it covers at least 10 continuous target residues that are not covered by any of the selected templates, or if its pairwise TM-score with the template with the highest weight is > 0.7, it is chosen. This step is repeated until all the candidate templates have been checked. For example, in Figure 2.12b, T1 is the template with highest template weight, which is selected first. T2 is selected because the TM-score between T1 and T2 is > 0.7. T3 is chosen because it covers at least 10 continuous uncovered target residues. However, the left region (gray) of T3 may be removed after template superposition if the TM-score between it and any selected template is < 0.7. This process is to filter structurally inconsistent coordinates before averaging them in order to reduce structural noise.

The selected templates are then superposed by the TM-score program [32] in order to make them structurally consistent and to make their averaged coordinates reasonable as follows. We used TM-score because we filtered unaligned residues from the template structures and the residue names in templates were replaced with those of the target protein. So, pre-handled templates were supposed to be generated from the same protein sequence. The template with the highest weight is selected as the center template. All the other templates are superposed with the center template if they share common residues with the center template. If a template does not share common residues with the center template, it is superposed with an already superposed template that shares most common residues with it. Figure 2.12c illustrates the template superposition protocol using five templates. T1 is the center template. T2, T3, and T4 are superposed with T1 because they share common residues with T1. T5 does not share common residues with T1, so it is superposed with T4. The superposed template structures only contain the x, y, z coordinates of Cα atoms. The superposed template structures are used to generate the average coordinates and the point clouds of the target residues.

2.5.3 Modeling gaps

The residues in a target that are not covered by any selected template are gaps (e.g. loops). Our method models the conformations of gaps by iteratively sampling points based on spatial distance restraints. The restraints include: (1) the distance between any pair of Cα atoms should be > 3.5Å and (2) the distance between two adjacent Cα atoms should be < 4.5Å. Terminal gaps or internal gaps are handled differently as follows.

1. Gaps at N terminal or C terminal

```
A = the first covered target residue adjacent to the gap at the N-terminal or the last covered target residue adjacent to the gap at C-terminal;
While there is gap
    Sample points around A;
    Find point B where there is no atom clash and the distance between A and B is in [3.5, 4.5];
    A = B.
```

2. Gap in the middle

```
A = the last covered residue before gap;
B = the first covered residue after gap;
While there is gap
    Calculate the distance between A and B as $d_{AB}$;
    Sample points around A;
    Find point C where there is no or least atom clash, the distance between A and C is in [3.5, 4.5], and the distance between B and C is in [max(3.5, dd*(k-1)), max(4.5, max(3.8*k, dd*k))]. Here k is the number of remaining gaps and dd is $d_{AB}/(k + 1)$;
    A = C.
```

Figure 2.12d demonstrates how the two kinds of gaps are filled.

2.5.4 Packing other main chain atoms and side chain atoms

The model constructed for the target protein using the method described above only contains the coordinates of Cα atoms (i.e. Cα trace). Other main chain atoms (C, N, O) and side chain atoms need to be added. We use Pulchra [53] to add other main chain atoms (C, N, O) and SCWRL4.0 [54] to add side chain atoms according to residue types.

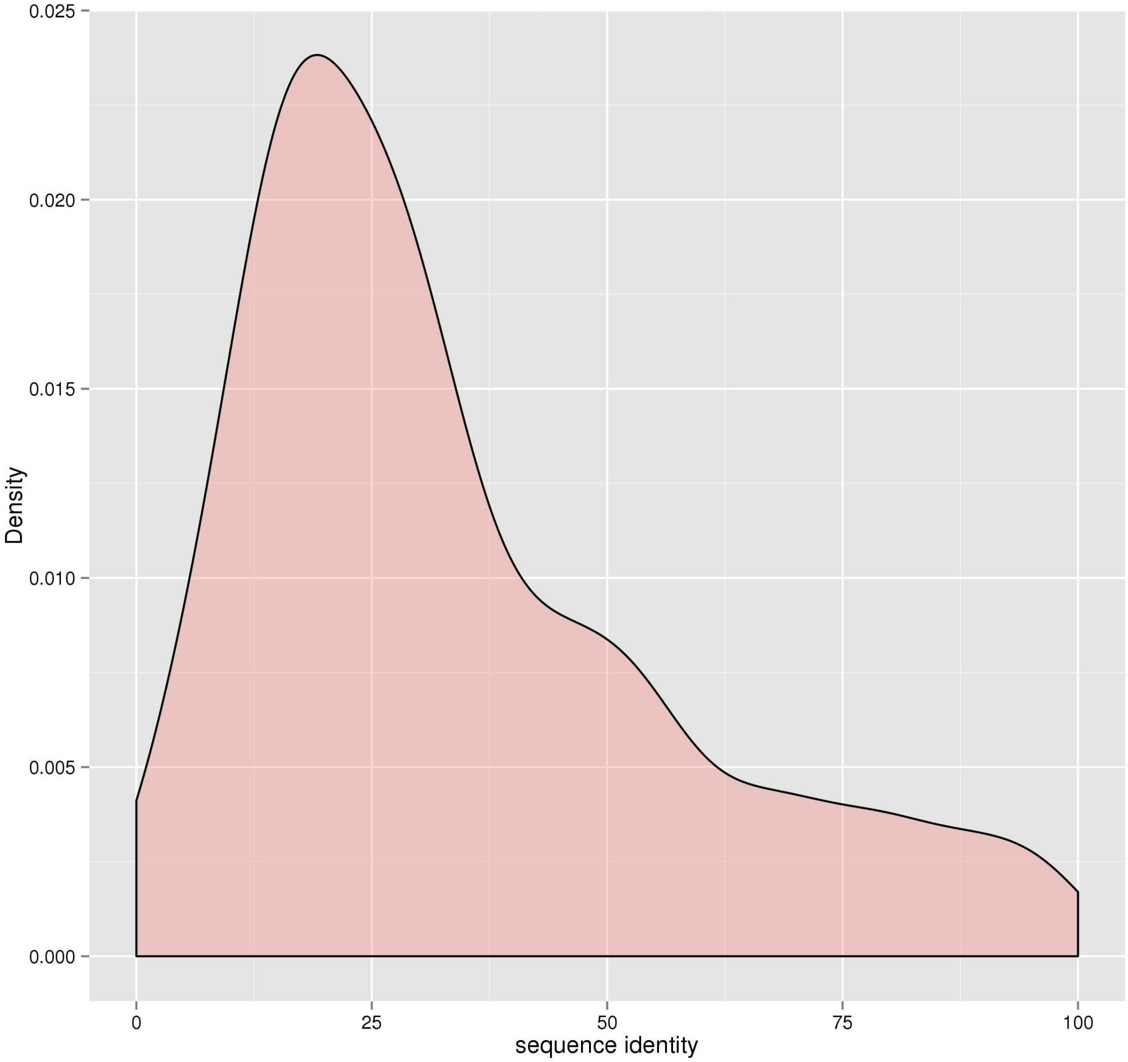


Figure 2.1: The distribution of sequence identity in the sequence alignments.

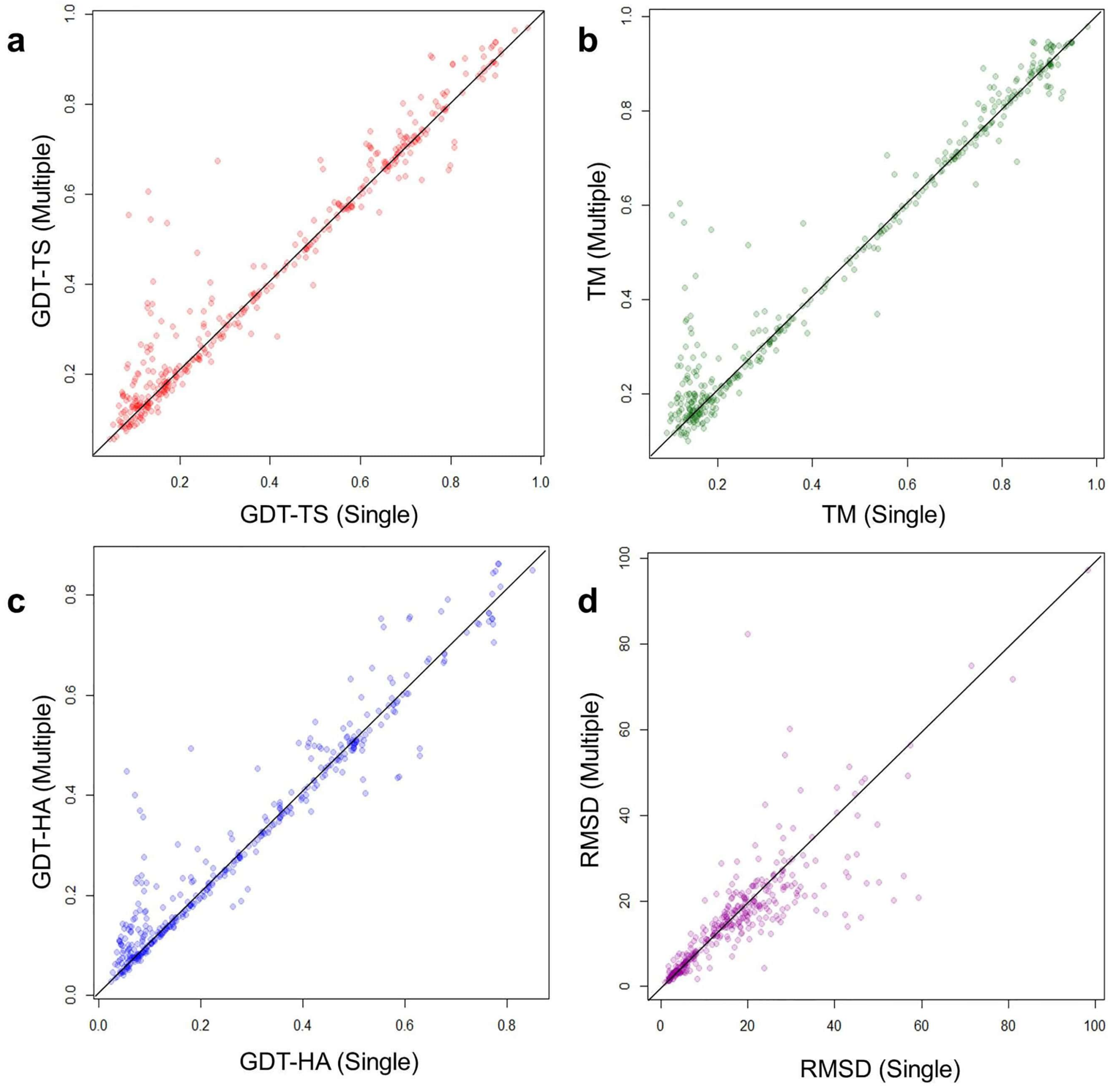


Figure 2.2: The improvements or losses of GDT-TS score, TM-score, GDT-HA score and RMSD of models predicted by MTMG using the first single templates and multiple templates on individual CASP11 targets. The scores of multi-template models are plotted against single-template models. X-axis represents the scores of single-template models and Y-axis represents the scores of multi-template models.

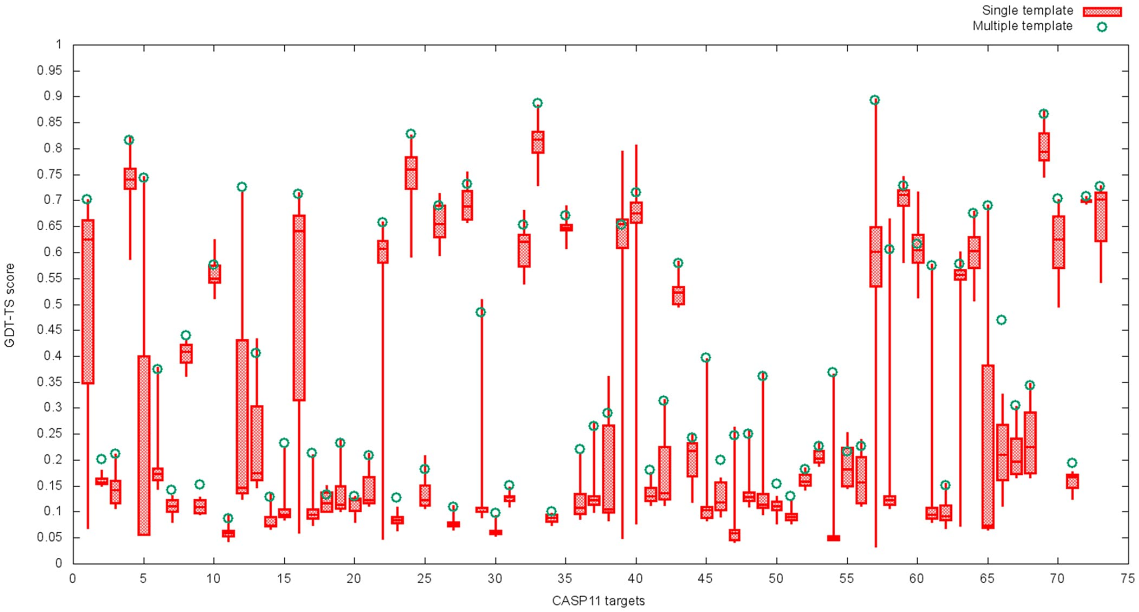


Figure 2.3: The boxplot of GDT-TS scores of the models predicted by MTMG for each of 73 CASP11 domains using each single template and multiple templates. The box plot denotes the maximum, 75% quartile, mean, 25% quartile, and minimum score of the models constructed from each single template for a target. The small green circle denotes the score of the model constructed from multiple templates.

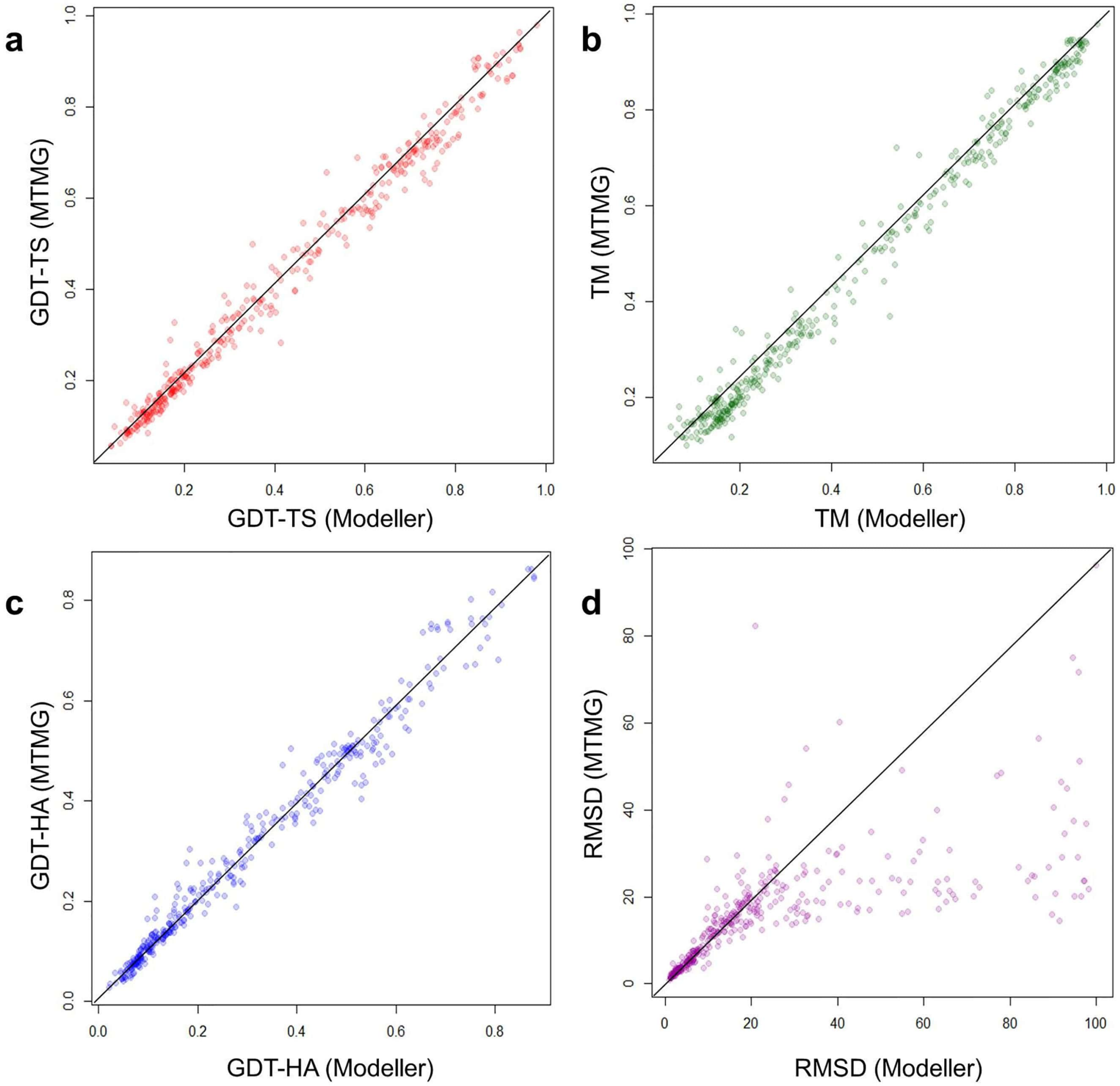


Figure 2.4: The scatter plot of GDT-TS scores, TM-scores, GDT-HA scores and RMSDs of the models predicted by MTMG against those of Modeller on CASP11 targets. The scores of Modeller models are plotted against MTMG models. X-axis represents the scores of Modeller models and Y-axis represents the scores of MTMG models.

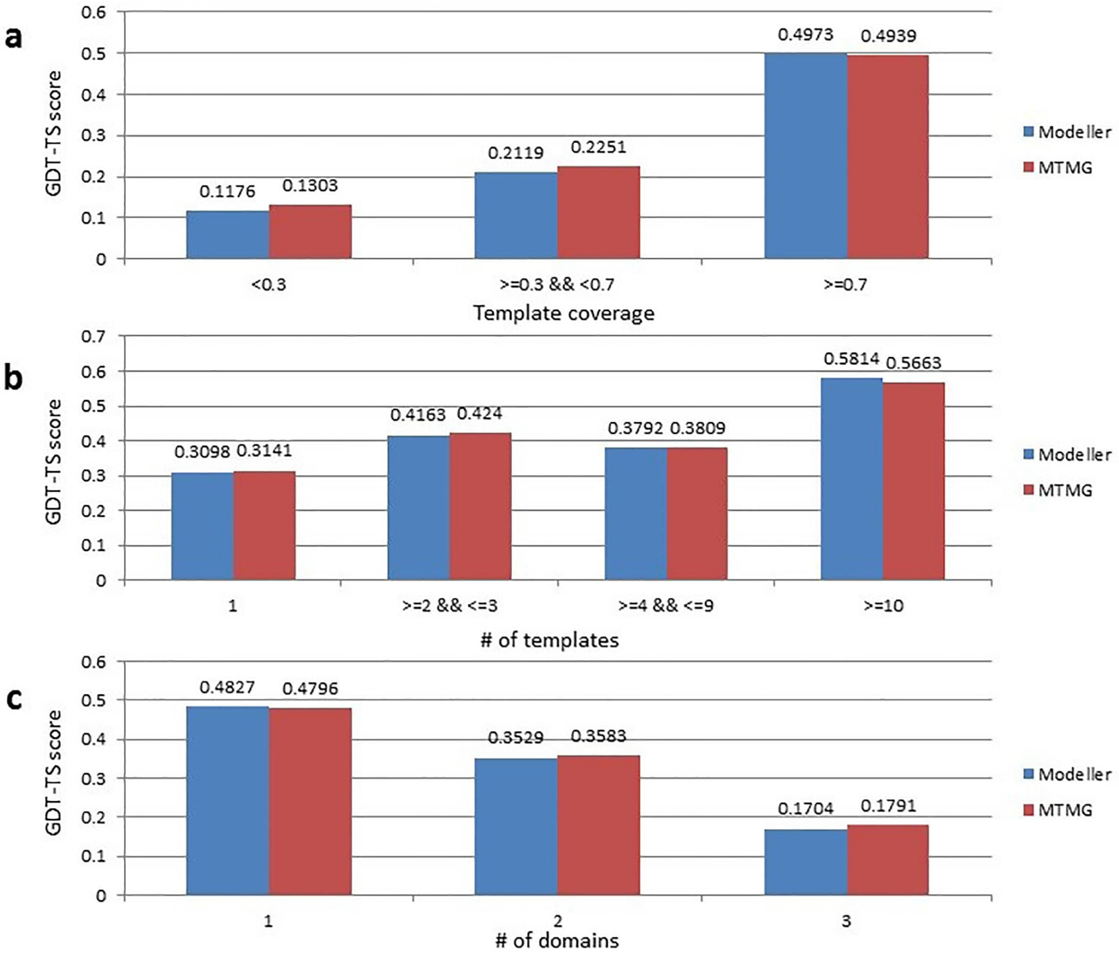


Figure 2.5: Comparison of GDT-TS score between the MTMG models and the Modeller models from three aspects on CASP11 targets. (a) MTMG performed better than Modeller on targets with <0.7 template coverage. (b) MTMG performs better than Modeller on targets covered by <10 templates. (c) MTMG performs better than Modeller on targets containing multiple domains.

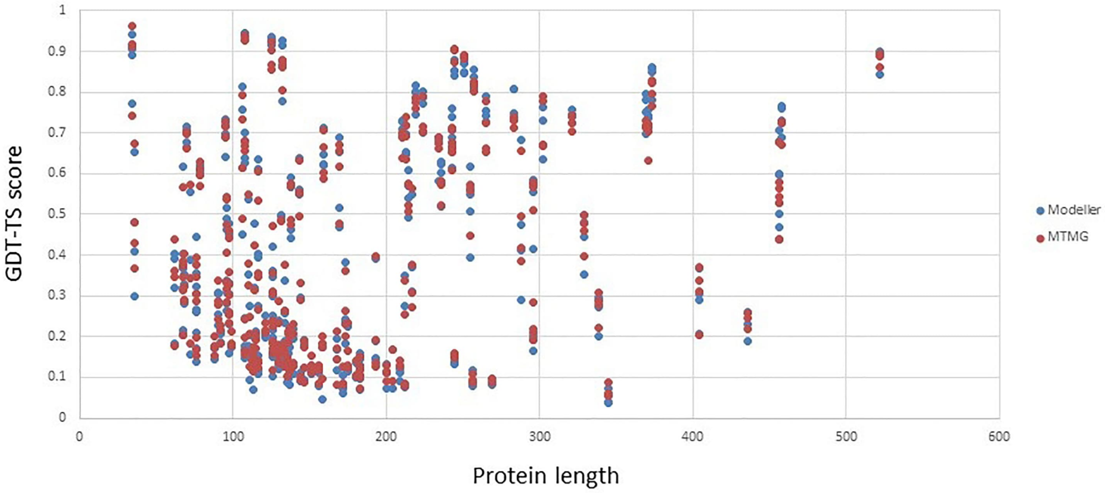


Figure 2.6: The GDT-TS scores of MTMG and Modeller models on different protein lengths. Red points donate GDT-TS scores of MTMG models, and blue points donate GDT-TS scores of Modeller models.

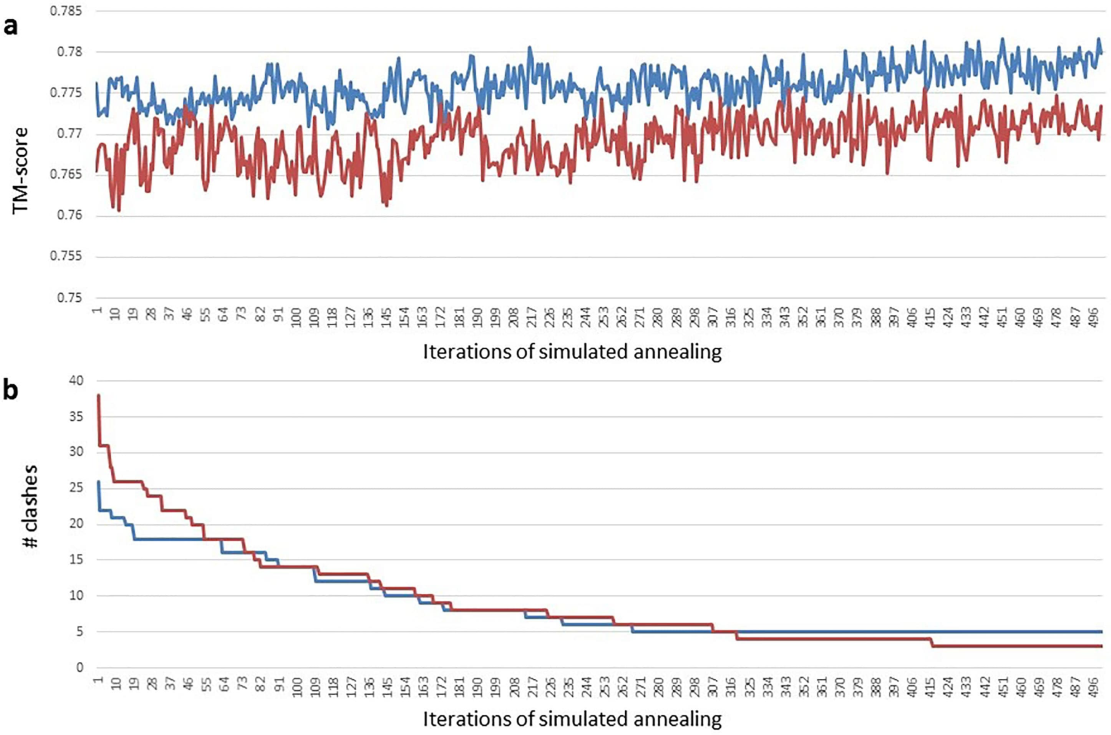


Figure 2.7: Changes of TM-score (a) and the number of atom clashes (b) of the models for two CASP11 targets during the simulated annealing. TM-score stochastically went up and down with an overall upward trend during simulated annealing. Even though the final model was not the best one, but it was close to the best one and better than the initial model. Moreover, the number of clashes rather consistently decreased during simulated annealing.

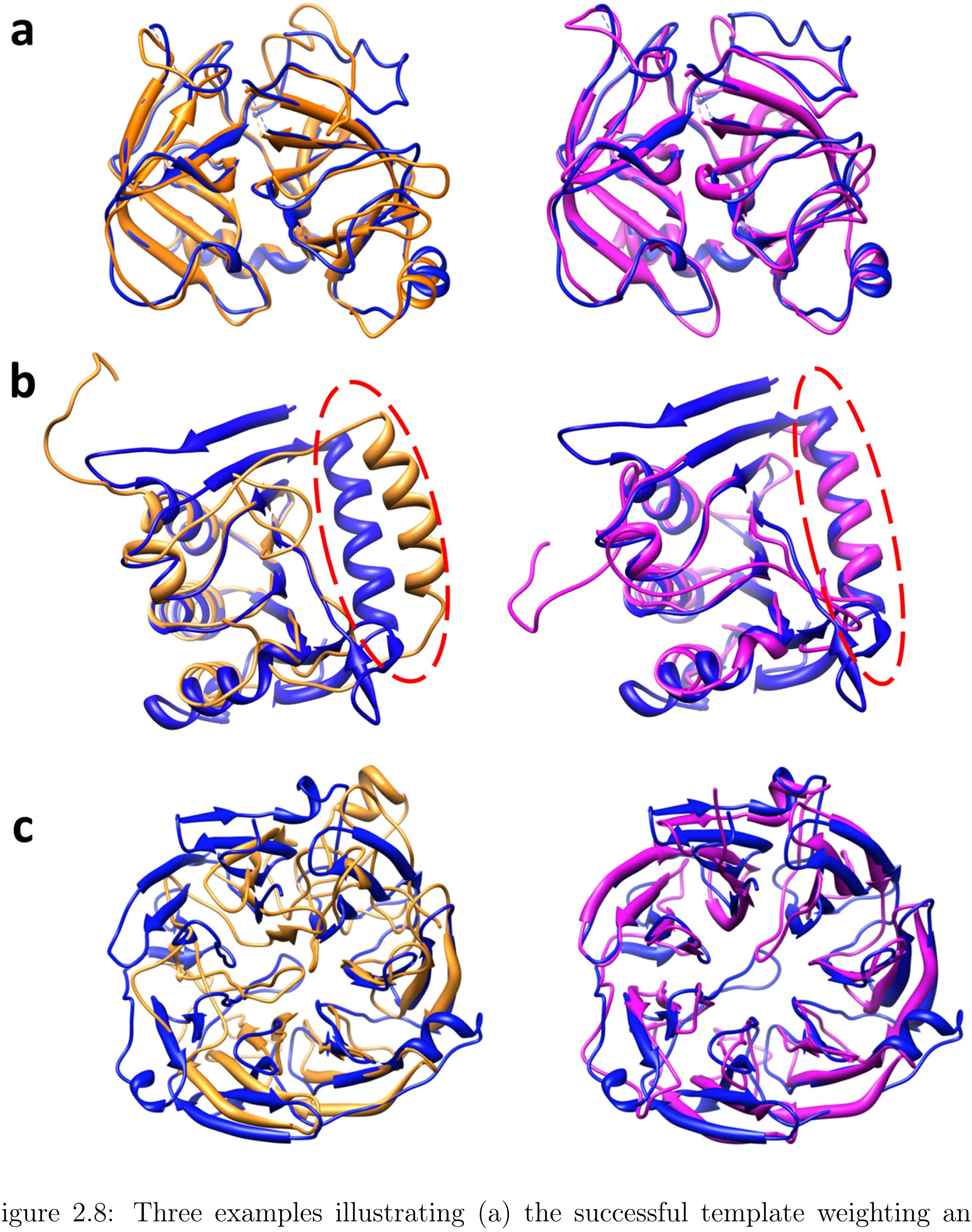


Figure 2.8: Three examples illustrating (a) the successful template weighting and combination, (b) the successful template superposition, and (c) the successful domain division and combination of our method. The models predicted by Modeller (gold) and MTMG (purple) were superposed with the native structure (blue).

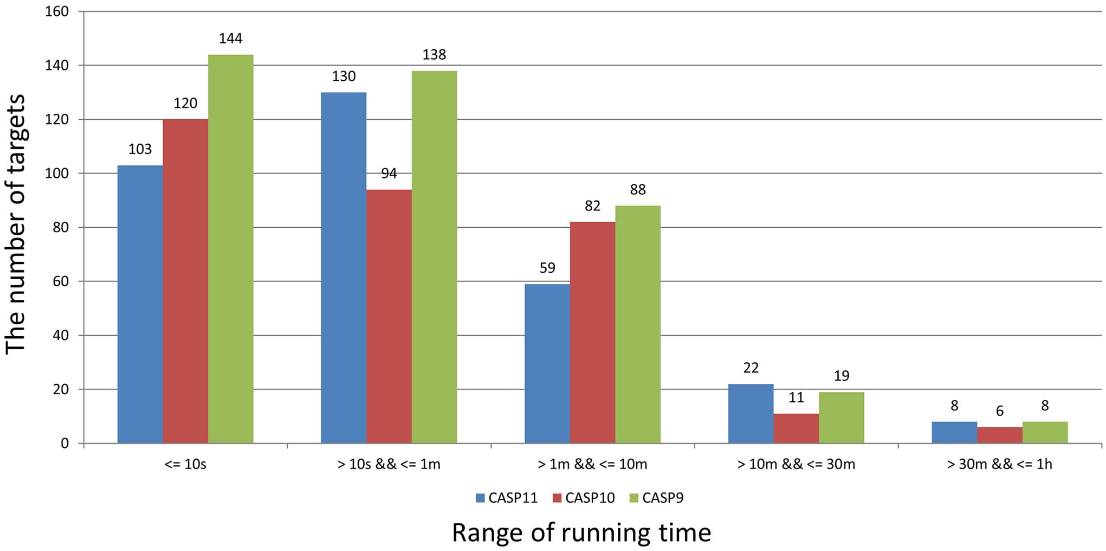


Figure 2.9: The number of targets in different ranges of running time on CASP9, CASP10, and CASP11 targets. 92.83% of targets were modeled by MTMG within 10 minutes, and all the targets were modeled in an hour in the experiment.

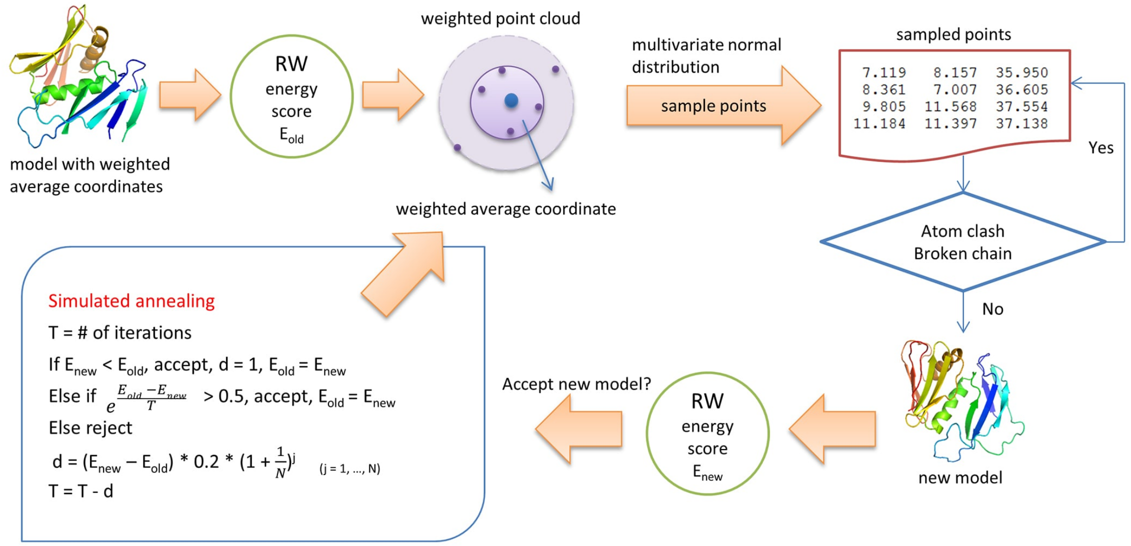

model with weighted average coordinates
RW energy score E_{old}
weighted point cloud
weighted average coordinate
multivariate normal distribution
sample points
sampled points
7.119 8.157 35.950
8.361 7.007 36.605
9.805 11.568 37.554
11.184 11.397 37.138
Yes
Atom clash Broken chain
No
new model
RW energy score E_{new}
Accept new model?
Simulated annealing
T = # of iterations
If $E_{new} < E_{old}$, accept, d = 1, $E_{old} = E_{new}$
Else if $e^{\frac{E_{old}-E_{new}}{T}} > 0.5$, accept, $E_{old} = E_{new}$
Else reject
$d = (E_{new} - E_{old}) * 0.2 * (1 + \frac{1}{N})^j$ (j = 1, ..., N)
T = T - d

Figure 2.10: The workflow of the stochastic point cloud method for sampling conformations. Starting from an initial model comprised of the weighted average coordinates of template structures, its RW energy is calculated as E_{old}, weighted point clouds are constructed for unfixed residues whose conformations are uncertain. New positions are sampled for unfixed residues from the multivariate normal distribution representing the point clouds, the positions with few or no atom clashes or broken chain are accepted to generate a new model. The new model is accepted based on the difference between its energy E_{new} and the old energy E_{old} according to a simulated annealing protocol, and the accepted model is used as the initial model for the next round of modeling, which is repeated until reaching a fixed number of iterations.

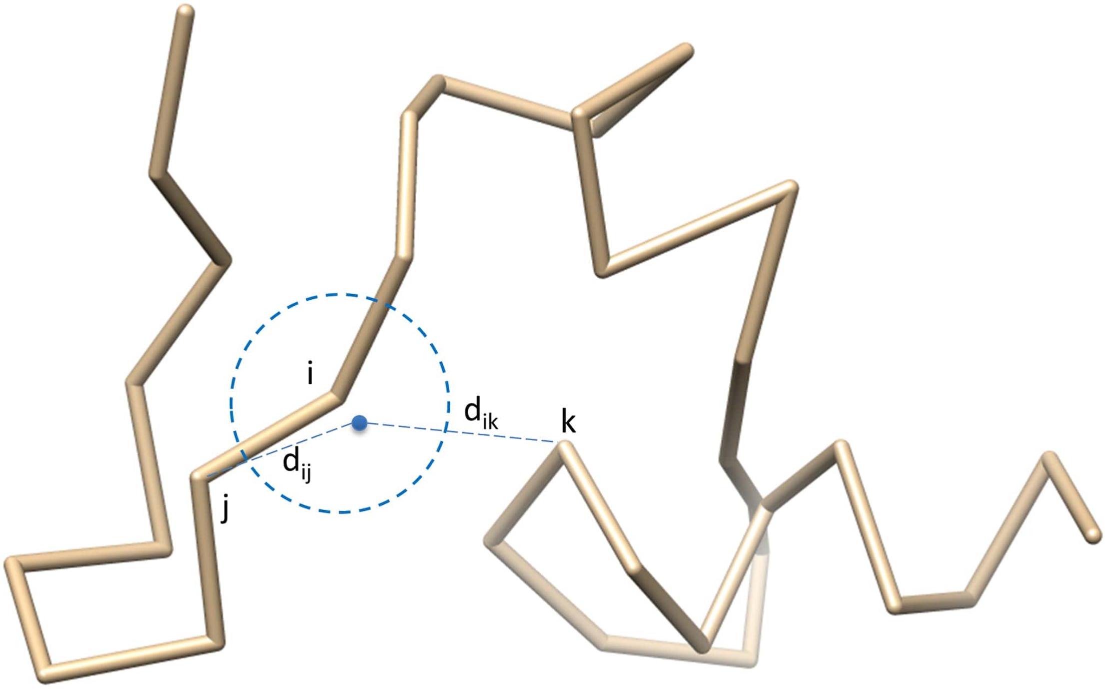


Figure 2.11: Checking the validity of sampled points. The Euclidean distance of the backbone atom Cα is calculated between the sampled point of the ith residue and each of other residues. The sampled point is accepted if it satisfies the spatial restraints without broken chains (i.e. too far away from adjacent atoms: d$_{ij}$ $>$ 4.5Å) and atom clashes (too close to other atoms: d$_{ik}$ $<$ 3.5Å).

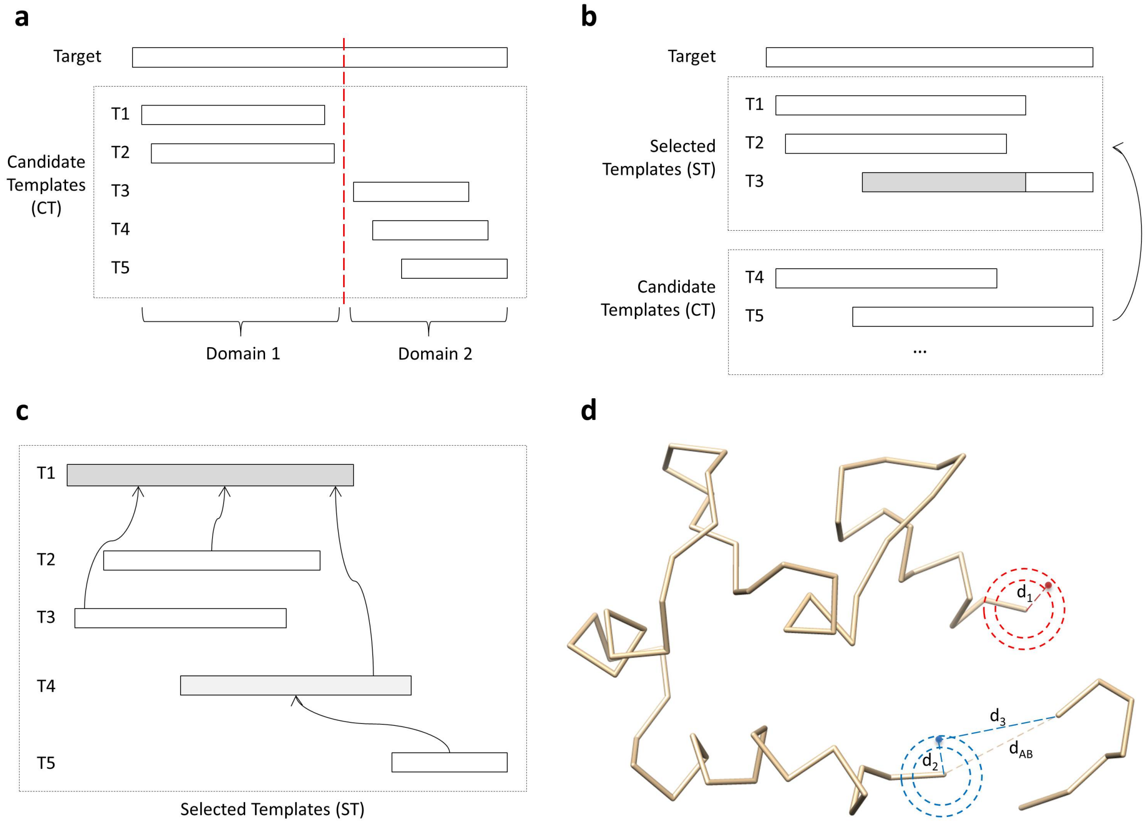


Figure 2.12: (a) Domain division. A target protein covered (aligned with) five templates is divided into two domains because the two regions do not share any common templates. (b) Template combination. The template T1 with the highest template weight is selected first. T2 is selected because the TM-score between T1 and T2 is > 0.7. T3 is chosen because it covers at least 10 continuous uncovered target residues. (c) Template superposition. T1 is the center template. T2, T3, and T4 are superposed with T1 because they share common residues with T1. T5 does not share common residues with T1, so it is superposed with T4. (d) Sampling points for gaps. The radius of the outside circle is 4.5Å, and the radius of the inner circle is 3.5Å. The sampling algorithm randomly samples point between the two circles. In the region circled by red, the gap is at the N-terminal. The distance d_1 between an accepted sampled point and the first covered residue is between 3.5Å and 4.5Å. In the region circled by blue, the three-residue gap is in the middle, and the distance between the two ends of the gap (d_{AB}) is 8.2Å. The distance d_2 between an accepted sampled point and the last covered residue before the gap is between 3.5Å and 4.5Å. The distance d_3 between an accepted sampled point and the first covered residue after the gap is between 4.1Å and 11.4Å.

Chapter 3

A Large-Scale Conformation Sampling and Evaluation Server for Protein Tertiary Structure Prediction and its Assessment in CASP11

3.1 Abstract

With more and more protein sequences produced in the genomic era, predicting protein structures from sequences becomes very important for elucidating the molecular details and functions of these proteins for biomedical research. Traditional template-based protein structure prediction methods tend to focus on identifying the best templates, generating the best alignments, and applying the best energy function to rank models, which often cannot achieve the best performance because of the difficulty of obtaining best templates, alignments, and models. We developed a large-scale conformation sampling and evaluation method and its servers to improve

the reliability and robustness of protein structure prediction. In the first step, our method used a variety of alignment methods to sample relevant and complementary templates and to generate alternative and diverse target-template alignments, used a template and alignment combination protocol to combine alignments, and used template-based and template-free modeling methods to generate a pool of conformations for a target protein. In the second step, it used a large number of protein model quality assessment methods to evaluate and rank the models in the protein model pool, in conjunction with an exception handling strategy to deal with any additional failure in model ranking. The method was implemented as two protein structure prediction servers: MULTICOM-CONSTRUCT and MULTICOM-CLUSTER that participated in the 11th Critical Assessment of Techniques for Protein Structure Prediction (CASP11) in 2014. The two servers were ranked among the best 10 server predictors, demonstrating the effectiveness and robustness of the large-scale conformation sampling and evaluation. The MULTICOM server is available at: http://sysbio.rnet.missouri.edu/multicom_cluster/.

3.2 Background

With the wide application of high-throughput next-generation sequencing technologies, the number of protein sequences is growing exponentially in the genomic era. Since protein functions are determined by protein structures, obtaining the structures of these proteins holds the key of utilizing this huge protein resource for biomedical research, bioengineering, and biotechnology development [1, 2].

Even though protein structures can be determined by experimental techniques such as x-ray crystallography and nuclear magnetic resonance (NMR), they can be only applied to solve the structures of a tiny portion of proteins due to their relatively high cost. Since the tertiary structure of a protein is almost uniquely specified by its amino acid sequence [55], computational methods of predicting protein structures from sequences are not only feasible, but also important to reduce the huge protein sequence-structure gap [3, 56, 57, 58, 59].

Computational protein structure prediction methods can be broadly classified into two categories: template-based modeling (TBM) [4, 5, 6, 7, 8, 9, 10, 11, 38] and template-free modeling (FM) [10, 12, 13]. Template-based modeling is based on the fact that evolutionarily related proteins tend to have similar structures [60] and structures change much slower than sequences [61]. Therefore, in order to predict the structure of a target protein, template-based modeling tries to find a target's homologous protein with known structure and use it as a template, then transfer the structure of the template to the target based on their sequence alignment, and finally adjust the structure to account for the variation from the template sequence to the target sequence [62]. Thus far, template-based modeling is the most widely used and most accurate technique for protein structure prediction. However, it cannot work when no good template is found. In this situation, template-free modeling is needed to build protein structures from scratch or from the combination of small structural fragments. Even though template-based modeling and template-free modeling use very different techniques for protein structure prediction, they are in common in sampling protein conformations in a huge conformation space for a target. The former is just a more focused, targeted sampling based on known, related structural points in the space, whereas the latter is a more unbiased, random sampling to explore a large conformation space.

In order to improve the reliability and robustness of conformation sampling, some recent protein structure prediction methods start to enlarge the sampling space of template-based modeling rather than focusing on one or a few "best" points, and also try to integrate template-based modeling and template-free modeling when no good templates or only partial templates can be found for a target protein [37, 35, 36, 63]. Based on our previous work of integrating multiple templates and alignments [37, 35, 36], we continued to develop and improve the large-scale conformation sampling approach to increase the diversity of template sampling, sequence alignment sampling, and model generation and to complement template-based modeling with template-free modeling in order to create a model pool of good quality. Given a pool of conformations for a target, another innovation is to apply an array of protein model quality assessment methods [39] to evaluate the quality of the models and rank them rather than using only one or a few quality assessment methods as almost all other protein structure prediction methods do. Furthermore, we added a new exception handling protocol to detect the problems in the final model ranking in order to correct the errors slipped through the large-scale model evaluation.

We implemented the large-scale conformation sampling and evaluation approach as two automated servers: MULTICOM-CONSTRUCT and MULTICOM-CLUSTER, which share the same conformation sampling protocol, but differ in the implementation of large-scale model quality assessment. We blindly benchmarked the two servers in the 11th Critical Assessment of Techniques for Protein Structure Prediction (CASP11) in 2014. According to the CASP11 official assessment, the two servers were ranked among the best 10 server predictors for protein tertiary structure prediction and were effective for the targets of a wide-spectrum of difficulty.

3.3 Methods

Figure 3.1 illustrates the large-scale model sampling and evaluation method implemented in our servers (MULTICOM-CONSTRUCT and MULTICOM-CLUSTER). Given a target protein sequence, the method uses sequence-sequence alignment tools or sequence-profile alignment tools (e.g., PSI-BLAST [64], BLAST [64, 65], CS-BLAST [43], CSI-BLAST [43], SAM [66] and HMMer [42]) to search the sequence against a large template database consisting of around 125,000 proteins (a full copy of the PDB database [67] excluding the identical sequences), profile-profile alignment tools (e.g., HHSearch [41], HHSuite [41], HHblits [68], PRC [69], FFAS [70, 71] and COMPASS [72]) to search the sequence against a redundancy-reduced template database consisting of around 39,000 proteins, and locally installed MUSTER [73] and RaptorX [6] to search it against their smaller template databases. The parameters used with these alignment tools are described in Table 3.1.

Each alignment tool identifies a list of templates and generates a list of pairwise target-template alignments. This template identification process corresponds to sampling templates for the target protein in the protein fold space approximated by the template protein databases.

The method uses three different ways to combine homologous templates and alternative alignments. First, it combines each pairwise target-template alignment (i.e. seed alignment) with other pairwise target-template alignments whose e-value is equal to or not much larger than that of the seed alignment. This central-star alignment

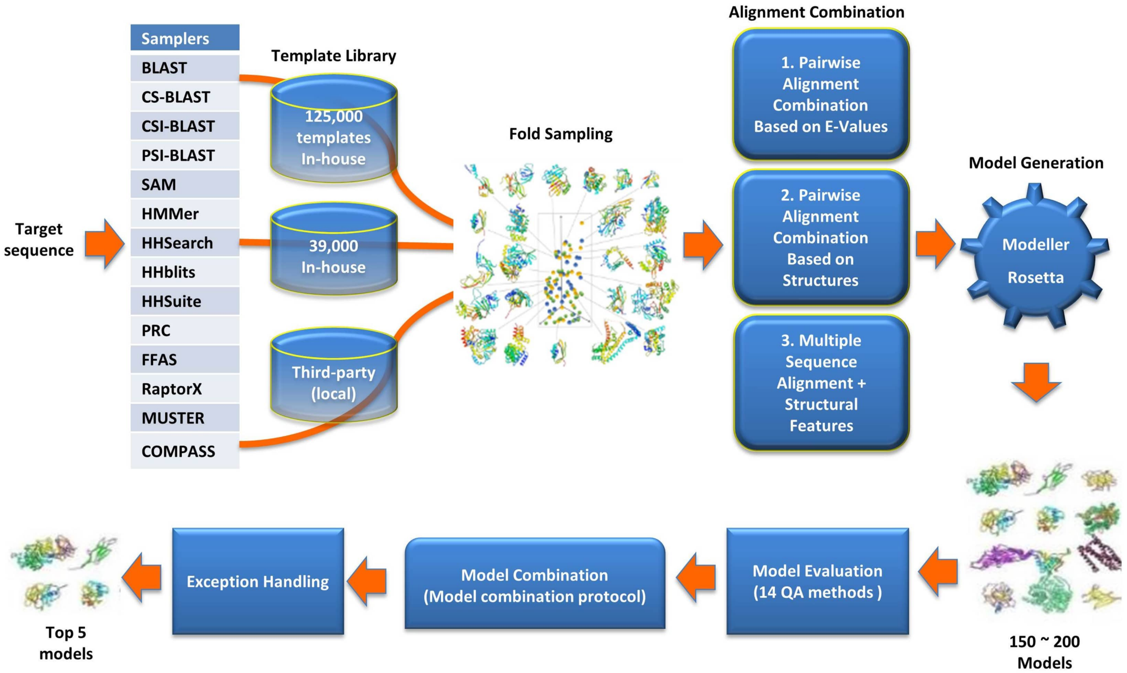


Figure 3.1: The large-scale model sampling and evaluation protocol of the MULTICOM protein structure prediction servers.

algorithm of generating multiple sequence alignments from pairwise alignments is described in details in [29]. The multiple templates in this combination often have, but do not guaranteed to have the similar tertiary structures. Second, it combines each pairwise target-template alignment with other pairwise target-template alignments whose aligned structures are similar according to structural comparison. This approach combines one seed alignment with other pairwise alignments whose templates have similar structures with the template in the seed alignment using the central-star algorithm as in [35]. This approach guaranteed that the structures of the combined templates are consistent. Third, it generates a consensus list of templates ranked by the number of times they are selected by the alignment tools during template identification, and then uses several multiple sequence alignment tools (e.g., MUSCLE [74],

Alignment tool	Version number	Type	Parameters
CSBLAST	2.1.0	Sequence-sequence	-j 5 -e 1 -h 0.0000000001
CSIBLAST	2.1.0	Sequence-sequence	-j 5 -e 1 -h 0.001
BLAST	2.2.17	Sequence-sequence	-F T -j 5 -e 1
PSI-BLAST	2.2.17	Profile-sequence	-j 5 -e 1 -h 0.001
SAM	3.5.i686	Profile-sequence	-calibrate 1 -sw 2 -dpstyle 0 -adpstyle 5
HMMer	3.0b3	Profile-sequence	Default parameters
FFAS	03	Profile-profile	Default parameters
COMPASS	2.42	Profile-profile	-e 1
PRC	1.5.6	Profile-profile	-hits 50 -align prc
MUSTER	3.0	Profile-profile	Default parameters
RAPTOR	1.66	Profile-profile	Number of models is set to 10
HHSearch	1.2	Profile-profile	Default parameters
HHSearch	1.5.0	Profile-profile	Default parameters
HHSearch	1.5.1	Profile-profile	-realign -mact 0
HHSearch in HHSuite	2.0.8	Profile-profile	-realign -mact 0
HHblits	2.2.17	Profile-profile	-oa3m -n $nr_iteration and -realign -mact 0

Table 3.1: The information of different sequence alignment tools used in our method. The information includes names, version numbers, alignment types (e.g. sequence-sequence, profile-sequence, and profile-profile), and parameters of these sequence alignment tools.

MSACompro [75], and MSAProbs [76]) to align the target with the top consensus template proteins in order to generate multiple sequence alignments.

The combined target-template alignments together with template structures are fed into Modeller [24, 25] to generate structural models using comparative modeling. For the targets without reliable templates identified, a template-free modeling tool Rosetta [63, 77] is called to generate dozens of models to complement the template-based models. Targets that contain both easy (template-based) and hard (template-free) domains are often decomposed into different chunks by dividing the target sequence into several sub-sequences (chunks) according to the sequence alignments, where easy domains are covered by homologous templates and hard domains aren't covered by any homologous template. Different modeling protocols (i.e., template-based protocol or template-free protocol) are chosen to predict the structures of each chunk. The conformations of all the chunks will be combined into a full-length model using Modeller [24, 25] by using the structural models of different chunks as the templates for the target protein. In total, about 150 - 200 structural models are generated for a target using the protocol described above.

The pool of predicted models was evaluated by 14 quality assessment (QA) methods (e.g., MULTICOM-NOVEL_QA - a new in-house single QA method, ModFOLDclust2 [78], ProQ2 [79], Pcons [80], APOLLO [81], ModelEvaluator [82], ModelCheck2 - an improved version of ModelEvaluator, QApro - a weighted combination of ModelEvaluator and APOLLO, SELECTpro [83], Dope [84], DFIRE2 [85], OPUS_PSP [86], RWplus [30], RF_CB_SRS_OD [87]). MULTICOM-CONSTRUCT used the consensus (average) ranking of the individual rankings produced by these methods to select top five models as predictions [39]. MULTICOM-CLUSTER selected top five models based on primarily Apollo pairwise similarity score in conjunction with the coverage and identify of template-target alignments, the e-values of alignments used to generate the models, and the types (i.e., template-based, template-free, or the combination of the two) of the models.

No matter how comprehensive the model evaluation process is, some bad models may still be ranked at the top occasionally. In order to solve the problem, for the first time, we designed an exception handling strategy to improve or replace the bad models within top five models in the following six situations: (1) If the top one model is a template-based model and $>=$ 40 residues in its front end or back end are not covered by (i.e. aligned with) any template, the conformation of these uncovered residues will be replaced by the conformation of another model that is covered by a template. (2) If the top one model is a template-based model and $<$ 40% of residues are covered by a template, the model will be replaced by another top-ranked model with $>=$ 40% template coverage. (3) If the top one model is template-based and the coverage of the most significant template is $<$ 30%, all other templates' coverage is $<$ 50% and the highest average pairwise similarity score in the model pool is $<$ 0.2 (i.e. a hard modeling case), the model is replaced by another top-ranked model if available. (4) If a model with $>$ 0.7 target-template sequence identity and $>$ 0.8 template coverage exists in the model pool and the highest average pairwise model similarity is $>$ 0.4 GDT-TS score (i.e. an easy modeling case), the top one model is replaced with the model that has highest target-template sequence identity and $>$ 0.8 template coverage and the highest average pairwise GDT-TS score. (5) If the top ranked model is a combination of models of protein domains and a significant template with $>$ 0.7 coverage is found, the top model may be replaced by a highly ranked model without domain combination. If domain division and combination happens, we check if the top domain-based model is better than the full-length model to decide if domain division and combination has to be reverted. If the e-value of templates in top ranked full-length models (e.g. HHSearch and RaptorX models) is $<$ e-6, the coverage of templates is $>$ 0.7, and the top GDT-TS score between the models is $>$ 0.35, the top full-length model will replace the top domain-based model as new top 1 model, and the domain-based model will be used as no. 5 model. And (6) If all the top five MULTICOM-CONSTRUCT models are ab initio models, no. 4 and no. 5 models are replaced with top two template-based models in order to increase the diversity of the submitted models.

3.4 Results and discussion

3.4.1 Summary of results

The two servers of our method participated in the 11th Critical Assessment of Techniques for Protein Structure Prediction (CASP11) in 2014. According to the CASP11 official assessment at http://www.predictioncenter.org/casp11/zscores_final.cgi (click on server groups), MULTICOM-CONSTRUCT and MULTICOM-CLUSTER were ranked among best 10 methods (no. 6 and no. 7) for protein tertiary structure prediction among 44 server predictors.

We evaluated MULTICOM-CONSTRUCT and MULTICOM-CLUSTER on the 105 CASP11 domains whose experimental structures were released to date. The difficulty of these domains ranges from easy template-based modeling to hard template-free modeling. Our submitted server models for 105 CASP11 domains were superimposed onto the true structures. GDT-TS scores and TM-scores of the models were calculated by the TM-score program [32]. Table 3.2 reports the average GDT-TS scores and TM-scores of top one and best of five models predicted by our servers. The average TM-Scores of the first submitted models and the best of five models are 0.54 and 0.56 respectively for the two servers, which are higher than the commonly accepted threshold of 0.5 for a correct topology. Table 3.3 reports the number of target domains for which our servers submitted models to CASP11 whose TM-Scores are higher than 0.5, a common threshold indicating if a model has correct topology. Our server submitted models with a TM-Score higher than 0.5 for ∼75% of the TBM domains. But TM-Scores of almost all the models submitted for FM domains are lower than 0.5, suggesting that generating or selecting good models for FM targets is still a major challenge.

Predictors	Top One		Best of Top Five	
	GDT-TS	TM-score	GDT-TS	TM-score
MULTICOM-CONSTRUCT	0.48	0.54	0.50	0.56
MULTICOM-CLUSTER	0.49	0.54	0.50	0.56

Table 3.2: Average GDT-TS scores and TM-scores of MULTICOM-CONSTRUCT and MULTICOM-CLUSTER models on 105 CASP11 domains. The numbers represent the average GDT-TS scores and TM-scores of top one and best of top five models predicted by MULTICOM-CONSTRUCT and MULTICOM-CLUSTER on 105 CASP11 domains.

Predictors	TBM domains (75)		FM domains (30)	
	Top 1	Best of 5	Top 1	Best of 5
MULTICOM-CONSTRUCT	57	59	0	0
MULTICOM-CLUSTER	54	57	0	1

Table 3.3: The number of target domains whose models have TM-Scores higher than 0.5. The numbers represent the number of target domains for which MULTICOM-CONSTRUCT and MULTICOM-CLUSTER submitted models to CASP11 whose TM-Scores with native structures are higher than 0.5 on 75 CASP11 TBM domains and 30 FM domains.

3.4.2 The quality of the model pool

We investigated the quality of the pool of conformations for each target generated by our servers in comparison with our submitted models and all the CASP11 models submitted by up to 44 server groups around the world. Figures 3.2 and 3.3 show the comparison of GDT-TS scores of top 1 models of MULTICOM-CONSTRUCT, top 1 models of MULTICOM-CLUSTER, the best models in the MULTICOM model pool, and the best of top 1 models in CASP11 on 75 easy TBM domains and 30 hard FM domains separately. The target domains were sorted by the scores of the best CASP11 models, which are some sort of indicators of the difficult of the target domains. From the figures, the best models in our model pool had the same (higher) GDT-TS scores as (than) the best of top 1 models in CASP11 on 16 TBM domains and 12 FM domains respectively. Figure 3.4 shows the comparison of GDT-TS scores of best of top 5 models of MULTICOM-CONSTRUCT, best of top 5 models of MULTICOM-CLUSTER, the best models in the MULTICOM model pool, and best of top 5 models in CASP11 on 17 CASP11 domains (12 easy TBM domains and 5 hard FM domains) where the best models in our model pool had the same (higher) GDT-TS scores as (than) best of top 5 models in CASP11. Also, the differences in GDT-TS scores between the best models in our model pool and the best models in the CASP11 model pool produced by dozens of protein structure prediction methods in the community are less than 0.02 and 0.05 on 40 and 65 domains separately. The results indicate that our large-scale conformation sampling method can generate good models for a large portion of targets.

Moreover, we evaluated a number of alignment tools used by the MULTICOM servers. Table 3.4 reports how many times each of the independent pairwise alignment tools excluding the alignment combination methods generated alignments leading to the creation of the best models for 75 TBM and 30 FM CASP11 domains. From the table, HHSearch / HHSuite and its variants contributed to the creation of the best models for 41 TBM and 8 FM domains, and performed best among these methods on both TBM and FM domains. However, it is worth noting that the MULTICOM

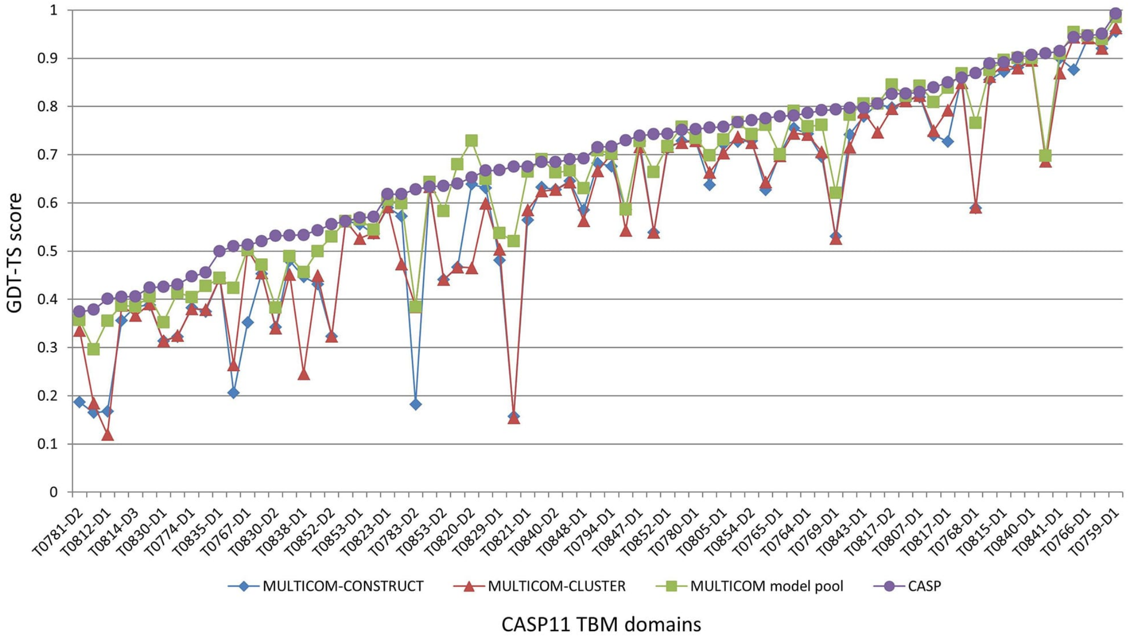


Figure 3.2: Comparison of top 1 models in the MULTICOM servers and CASP11 on 75 TBM domains. The comparison is based on GDT-TS scores of top 1 models of MULTICOM-CONSTRUCT, top 1 models of MULTICOM-CLUSTER, the best models in the MULTICOM model pool, and best of top 1 models in CASP11 on 75 easy TBM CASP11 domains.

servers used several different versions of HHSearch and their combined results were reported here. RaptorX, MUSTER, HHblits and COMPASS also contributed to the generation of the best models on both TBM and FM domains. In addition to these alignment tools, the template-free modeling tool Rosetta generated the best models for 1 TBM domain and 13 FM domains. The alignment and model combination algorithms in the MULTICOM server that combined the output of the independent alignment tools also generated best models for 24 TBM domains and 4 FM domains. The experiment shows that the combination of multiple different alignment tools improves the quality of the best models in the model pool and is an effective way to improve the reliability and robustness of protein structure prediction.

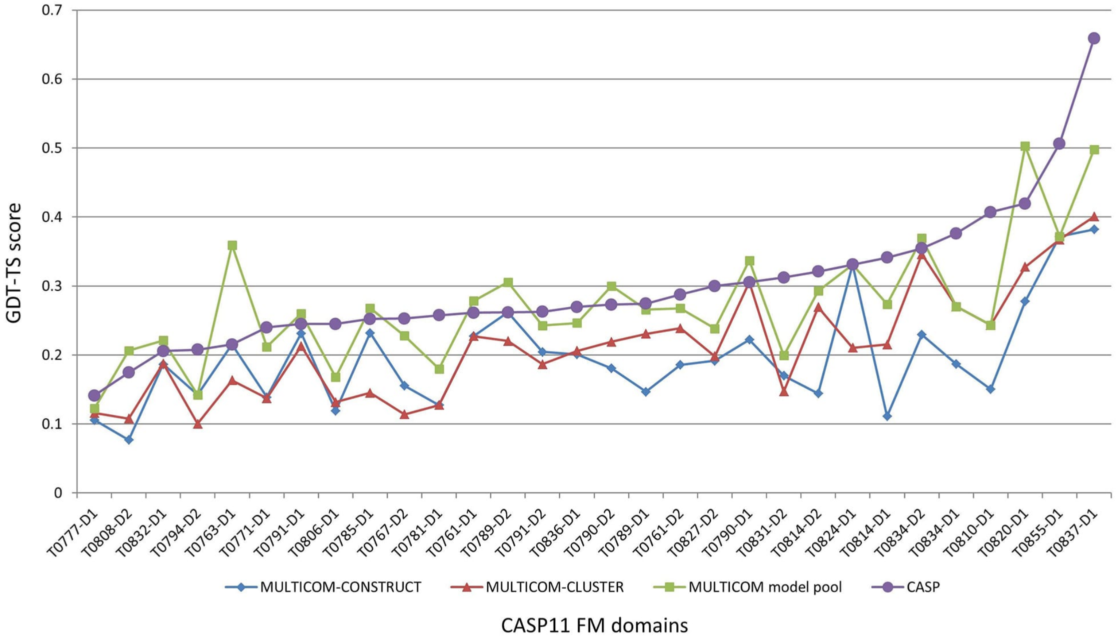


Figure 3.3: Comparison of top 1 models in the MULTICOM servers and CASP11 on 30 FM domains. The comparison is based on GDT-TS scores of top 1 models of MULTICOM-CONSTRUCT, top 1 models of MULTICOM-CLUSTER, the best models in the MULTICOM model pool, and best of top 1 models in CASP11 on 30 hard FM CASP11 domains.

3.4.3 Evaluation of the large-scale model ranking strategy

We compared top 1 models with best of top five models and the overall best models in the model pool for MULTICOM-CONSTRUCT and MULTICOM-CLUSTER on 105 CASP11 domains in order to check the performance of the ranking strategy. Figure 3.5 (A) and (B) illustrates the number of domains in various ranges of differences of GDT-TS scores between best of top 5 models and top 1 models generated by the two servers respectively. The differences of GDT-TS scores are small (i.e. $<$ 0.02) on 80 and 79 domains, and the top 1 models are the best of top five models on 43 and 41 domains for MULTICOM-CONSTRUCT and MULTICOM-CLUSTER

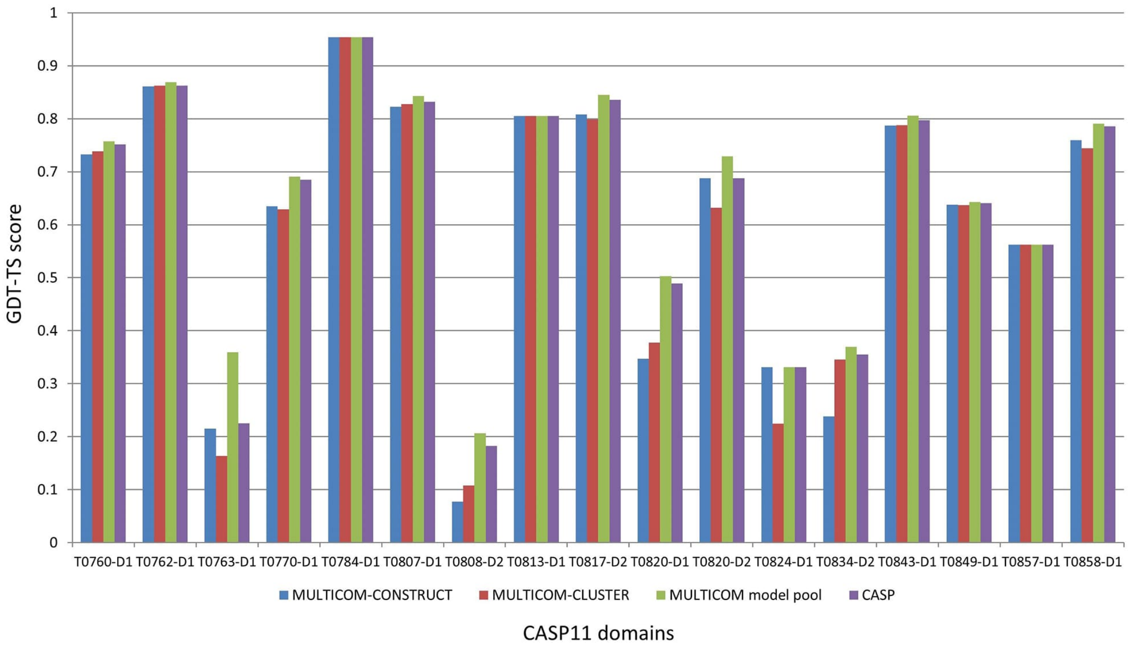


Figure 3.4: Comparison of top 5 models in the MULTICOM servers and CASP11 on 17 domains. The comparison is based on GDT-TS scores of best of top 5 models of MULTICOM-CONSTRUCT, best of top 5 models of MULTICOM-CLUSTER, the best models in the MULTICOM model pool, and best of top 5 models in CASP11 on 17 CASP11 domains where the best models in our model pool had the same (higher) GDT-TS scores as (than) best of top 5 models in CASP11.

separately. So, the ranking strategy worked generally well. However, it failed on some domains. For example, the top 1 model of T0816-D1 selected by MULTICOM-CONSTRUCT had a GDT-TS score 0.47, 0.21 less than 0.68 of the best of top five models. Figure 3.5 (C) and (D) shows scatter plots of GDT-TS scores between top 1 models and the best models in the model pool. The differences of GDT-TS scores are less than 0.02 on 46 and 50 domains, and less than 0.05 on 74 and 79 domains for MULTICOM-CONSTRUCT and MULTICOM-CLUSTER separately. Therefore, the ranking strategy successfully picked up good models on most of the domains.

We also evaluated the performance of 14 model quality assessment methods and

Alignment tool	# of times generating best models		
	TBM & FM domains	TBM domains	FM domains
HHSearch / HHSuite	49	41	8
RaptorX	13	10	3
MUSTER	7	5	2
HHblits	7	6	1
COMPASS	6	5	1
PSI-BLAST	2	2	
BLAST	1	1	
HMMer	1	1	
PRC	1	1	
FFAS	1	1	

Table 3.4: The number of times that each alignment tool contributed to generation of the best models. The numbers represent the number of times that each of the independent pairwise alignment tools excluding the alignment combination methods generated alignments leading to the creation of the best models for 75 TBM and 30 FM CASP11 domains. It is worth noting that the results of several versions of HHSearch used in the MULTICOM server are combined together.

their consensus ranking. The consensus ranking of a model is the average rank of all the rankings predicted by these methods for the model. Table 3.5 reports the number of times when top 1 models selected by an individual quality assessment (QA) method were actually the best of top 1 models identified by all the QA methods, and the number of times when top 1 models selected by an individual method were actually the best models in the MULTICOM model pool. "Avg loss" means the average loss (difference) in GDT-TS scores between the best models and top 1 models ranked by each QA method. A tolerance of marginal difference in scores is applied when calculating these numbers. In the previous CASP QA assessment papers [88, 89, 90], few predictors could identify best models within 0.02 GDT-TS score, and also the average loss of GDT-TS score of best QA methods is larger than 0.02. Therefore, in our experiment, if GDT-TS score of the top 1 model identifies by an individual method was within 0.02 GDT-TS score from the actual best model, the method is considered to successfully identify the best model. Also, we removed targets whose highest GDT-TS score is < 0.35 for the analysis because these models are of poor quality and GDT-TS score is not a good measure for differentiating models of less than 0.35 GDT-TS score. The table shows that the consensus ranking performed better than 14 individual QA methods in terms of selecting the top 1 model. SELECTpro, ModFOLDclust2, APOLLO, Pcons, and ProQ2 performed best among the 14 individual methods. However, the probability of any of these methods selecting the best model is low, indicating that selecting the best model is still more or less a guess game.

3.4.4 Case study

From our analysis, the submitted models of the two servers are the best models among all the CASP11 server models on six CASP11 domains: T0762-D1 (TBM), T0784-D1 (TBM), T0813-D1 (TBM), T0820-D2 (TBM), T0824-D1 (FM), and T0857-D1 (TBM).

Figure 3.6 (A) shows the structural superposition between the native structure of T0762-D1 (blue) and a high-accuracy model (no. 3) predicted by MULTICOM-CLUSTER (gold), which was reconstructed from multiple templates (i.e. 4IB2A, 4EF1A, 4OTEA, and 4K3FA). The model is the best model among all the models submitted to CASP11 for T0762-D1. It has a GDT-TS score 0.86 and RMSD 2.3Å with the native structure. Figure 3.6 (B) illustrates the distributions of GDT-TS scores of the MULTICOM server models (red) and the CASP server models (blue) of T0762-D1. Here, the MULTICOM server models include all the models in the

QA method	Best of top 1	Avg loss	Best in the pool	Avg loss
Consensus ranking	34	0.04	17	0.07
SELECTpro	32	0.05	17	0.08
ModFOLDclust2	30	0.07	18	0.10
APOLLO	30	0.07	16	0.09
Pcons	29	0.07	16	0.10
ProQ2	27	0.05	15	0.07
QApro	18	0.07	8	0.09
ModelCheck2	16	0.16	10	0.18
MULTICOM-NOVEL_QA	11	0.11	4	0.14
DFIRE2	9	0.11	6	0.14
Dope	9	0.11	6	0.14
ModelEvaluator	9	0.13	6	0.16
OPUS_PSP	9	0.11	6	0.14
RF_CB_SRS_OD	9	0.11	6	0.14
RWplus	9	0.11	6	0.14

Table 3.5: Comparison of 14 model quality assessment methods and their consensus ranking. "Best of top 1" means the number of times when top 1 models selected by an individual QA method were actually the best of the top 1 models identified by all the QA methods. "Best in the pool" means the number of times when top 1 models by an individual method were actually the best models in the MULTICOM model pool. "Avg loss" means the average loss of GDT-TS scores between the best models and top 1 models ranked by each QA method.

MULTICOM candidate pool. Density (Y-axis) represents the number of models. The two distributions are similar and most models have GDT-TS scores around 0.8 or above, but the MULTICOM model pool contains the best models. The results show that our method successfully identified homologous templates, generated good alignments, and constructed and picked up high-quality models for this domain.

Figure 3.7 (A) shows the structural superposition between the native structure of T0813-D1 (blue) and the top 1 model of MULTICOM-CONSTRUCT (gold), which was reconstructed from four templates (3KTDA, 2F1KA, 3B1FA, and 3GGOA). The model is the best model among all the models submitted to CASP for T0813-D1.

It has a GDT-TS score of 0.81 with the native structure. Figure 3.7 (B) illustrates the distributions of GDT-TS scores of the MULTICOM server models (red) and the CASP server models (blue). Here, the MULTICOM server models include all the models in the MULTICOM candidate pool. Density (Y-axis) represents the number of models. The distribution of the MULTICOM server models is bimodal, suggesting the models were constructed from both very good templates and some sub-optimal templates. The distribution of the CASP server models is uni-modal with mostly good models and a small number of low-quality models that may be constructed by template-free modeling methods or from bad templates. The MULTICOM server model is the best server model for T0813-D1 in CASP11, indicating that our method generated a pool of good models and selected the best models from the pool for this domain.

We also investigated the cases in which the MULTICOM servers failed due to not generating good models or not being able to select good models from the model pool. The most dramatic failure occurred on T0845-D1, for which the best submitted model (no. 2) by MULTICOM-CLUSTER has a GDT-TS score of 0.29. The GDT-TS score of the best model in the MULTICOM model pool is 0.52, 0.23 point higher than the best submitted model. Figure 3.8 (A) shows the structural superposition between the native structure of T0845-D1 (blue) and the best submitted model by MULTICOM-CLUSTER (gold) and the best model in the MULTICOM model pool (purple). The best model in the pool superimposed much better with the native structure. Figure 3.8 (B) visualizes the distributions of GDT-TS scores of the MULTICOM server models (red) and the CASP server models (blue) of T0845-D1. The majority of the MULTICOM server models except a few ones are of bad quality. The distribution of the CASP server models is bimodal, where a significant portion of models is of good quality. The GDT-TS score 0.71 of the best CASP11 model (TASSER-VMT_TS4) is much better than that of MULTICOM models, suggesting that our servers failed to generate good models. Also, the ranking strategy in our servers was not able to select the few relatively good models in its model pool on this domain. The case suggests that both model generation and model selection in the MULTICOM servers still have a significant room for improvement.

3.4.5 Availability

After being rigorously tested in CASP11, the protein structure prediction web service of MULTICOM-CLUSTER is released for public use at http://sysbio.rnet.missouri.edu/multicom_cluster/. Since MULTICOM-CONSTRUCT is slower than MULTICOM-CLUSTER and has similar accuracy as MULTICOM-CLUSTER, it is not made available for public use.

The experimental data of MULTICOM in CASP11 is available now. Users can access the data by clicking the link "Experimental data (models and alignments) in CASP11" on the home page of the MULTICOM server at http://sysbio.rnet.missouri.edu/multicom_cluster/.

3.5 Conclusions

We developed and implemented a large-scale conformation sampling and evaluation method to improve the reliability and robustness of protein structure prediction, overcoming the problem of failing to obtain the best template, alignment and model in traditional protein structure prediction methods. The approach can naturally integrate multiple templates, multiple alignments, and diverse sampling and evaluation methods into one system to improve model sampling and ranking as demonstrated by the good performance of our method in the CASP11 experiment. Furthermore, our analysis of the quality of conformation pool provides the new insights into the sampling and evaluation of protein models. Overall, the method and its server implementation are useful tools for protein structure predictors and users.

However, despite of the progress enabled by the large-scale sampling and evaluation approach, there are still some major challenges in protein structure prediction, including how to reliably identify weak homologous templates from irrelevant noisy templates, how to enrich the proportion of good alignments, how to distinguish a few good models from a large number of low-quality models, and finally how to generate better template-free models when no homologous template exists. In order to solve these problems, on the one hand more sensitive or complementary data mining methods need to be developed to mine a large number of templates, alignments, and protein models produced by existing methods, on the other hand novel methods for simulating alignments and structural models for hard targets more effectively are required to generate ensembles of protein conformations of better quality.

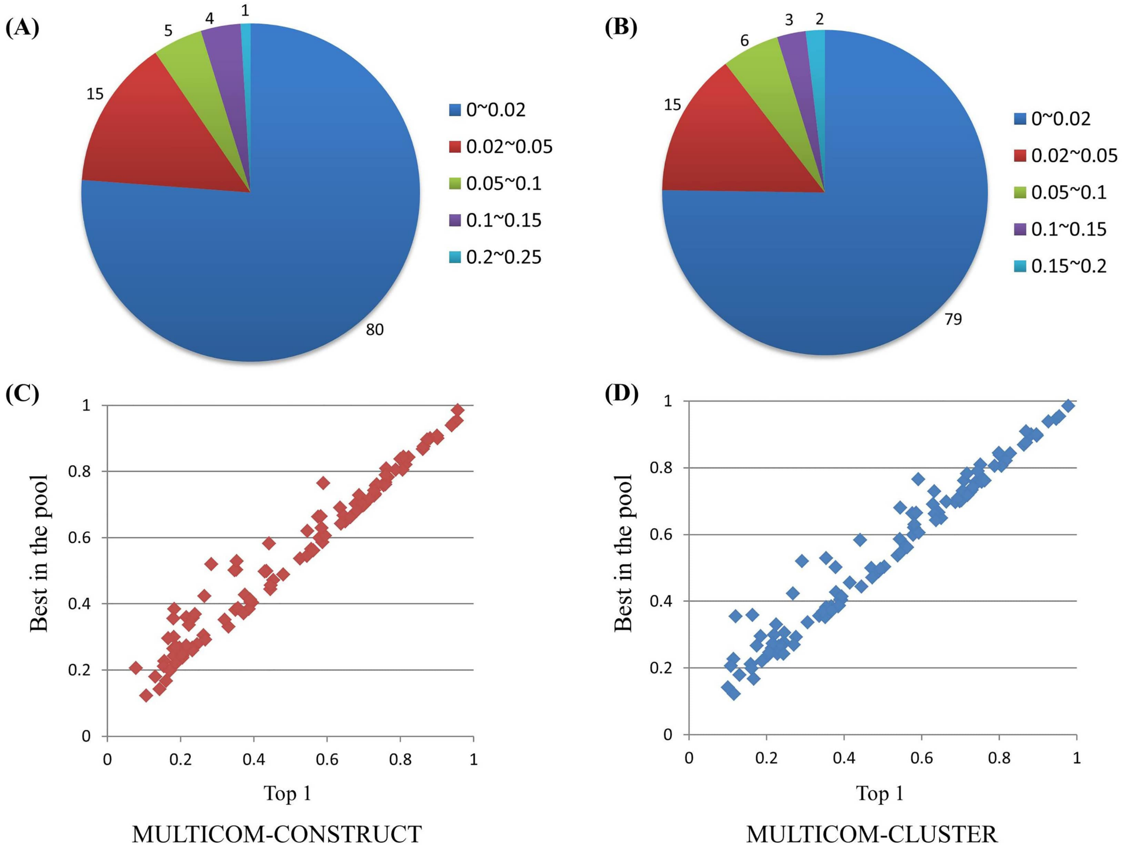


Figure 3.5: Evaluation of the ranking strategy. (A) and (B) illustrate the number of domains in various ranges of differences in GDT-TS scores between best of top 5 models and top 1 models generated by MULTICOM-CONSTRUCT and MULTICOM-CLUSTER respectively on 105 CASP11 domains. (C) and (D) show scatter plots of GDT-TS scores between top 1 models and the best models in the model pool for the two servers separately.

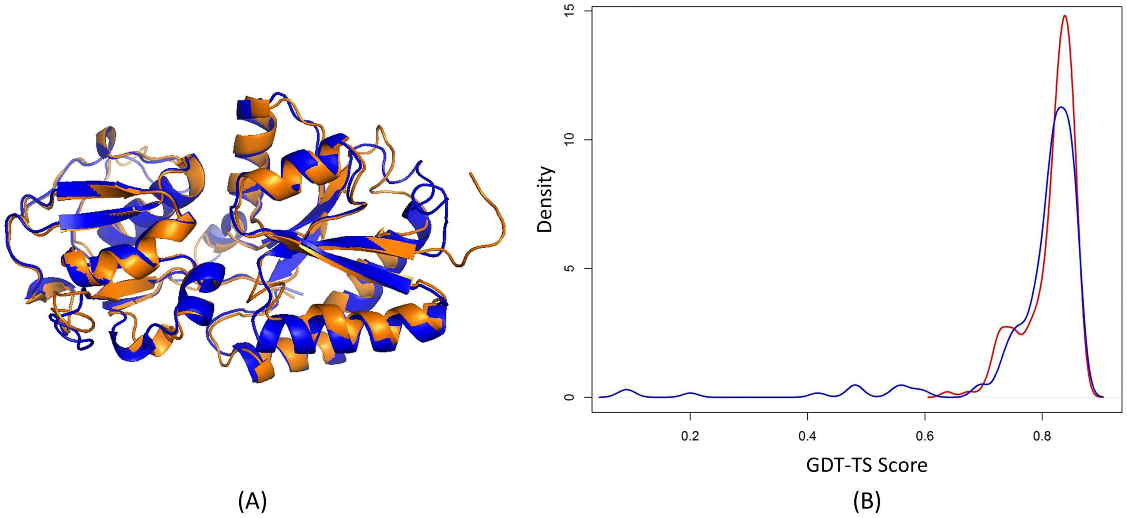


Figure 3.6: One good prediction of MULTICOM-CLUSTER on T0762-D1. (A) Structural superposition between the native structure of T0762-D1 (blue) and the no. 3 model of MULTICOM-CLUSTER (gold). (B) Distribution of GDT-TS scores of the MULTICOM server models (red) and the CASP server models (blue) of T0762-D1.

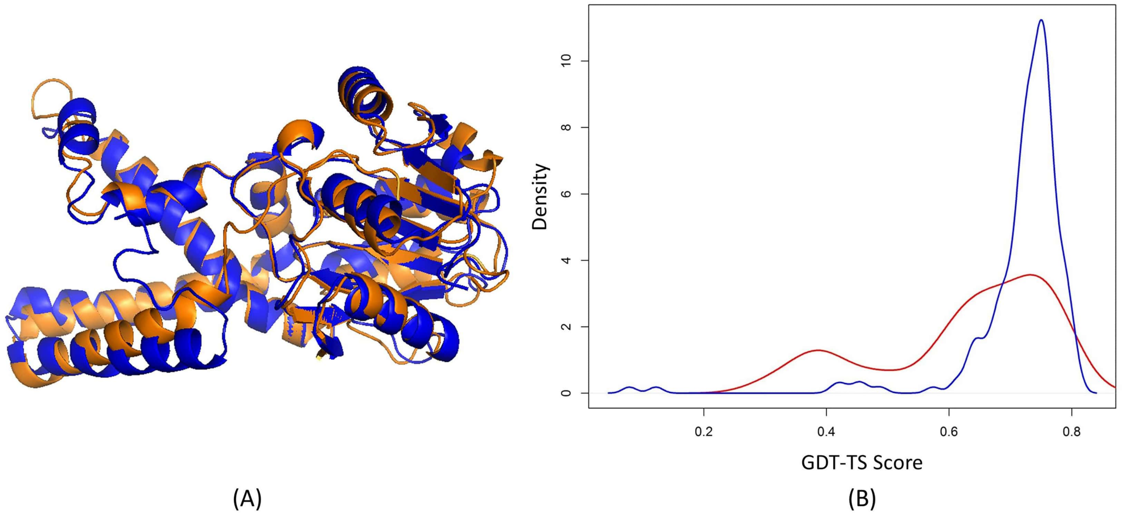


Figure 3.7: One good prediction of MULTICOM-CONSTRUCT on T0813-D1. (A) Structural superposition between the native structure of T0813-D1 (blue) and the top 1 model of MULTICOM-CONSTRUCT (gold). (B) Distribution of GDT-TS scores of the MULTICOM server models (red) and the CASP server models (blue) of T0813-D1.

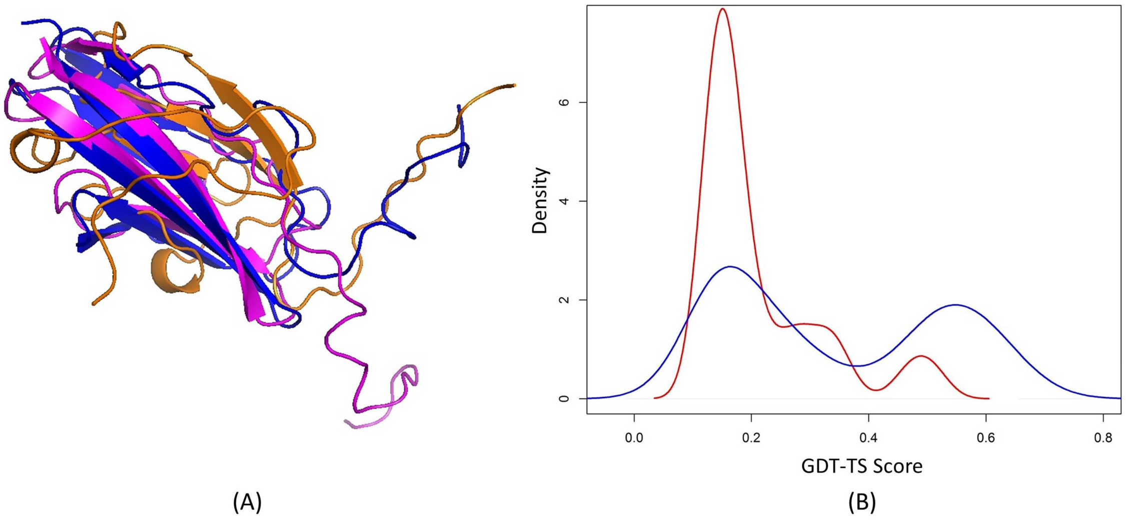


Figure 3.8: One bad prediction of MULTICOM-CLUSTER on T0845-D1. (A) Structural superposition between the native structure of T0845-D1 (blue) and the no. 2 submitted model of MULTICOM-CLUSTER (gold) and the best model in the MULTICOM model pool (purple). (B) Distribution of GDT-TS scores of the MULTICOM server models (red) and the CASP server models (blue) of T0845-D1.

Chapter 4

Designing and Benchmarking the MULTICOM Protein Structure Prediction System

4.1 Abstract

Predicting protein structure from sequence is one of the most significant and challenging problems in bioinformatics. Numerous bioinformatics techniques and tools have been developed to tackle almost every aspect of protein structure prediction ranging from structural feature prediction, template identification and query-template alignment to structure sampling, model quality assessment, and model refinement. How to synergistically select, integrate and improve the strengths of the complementary techniques at each prediction stage and build a high-performance system is becoming a critical issue for constructing a successful, competitive protein structure predictor. Over the past several years, we have constructed a standalone protein

structure prediction system MULTICOM that combines multiple sources of information and complementary methods at all five stages of the protein structure prediction process, including template identification, template combination, model generation, model assessment, and model refinement. The system was blindly tested during the ninth Critical Assessment of Techniques for Protein Structure Prediction (CASP9) in 2010 and yielded the very good performance. In addition to studying the overall performance on the CASP9 benchmark, we thoroughly investigated the performance and contributions of each component at each stage of prediction. Our comprehensive and comparative study not only provides useful and practical insights about how to select, improve, and integrate complementary methods to build a cutting-edge protein structure prediction system, but also identifies a few new sources of information that may help improve the design of a protein structure prediction system. Several components used in the MULTICOM system are available at: http://sysbio.rnet.missouri.edu/multicom_toolbox/.

4.2 Introduction

Predicting protein tertiary structure from sequence is an important and challenging problem in bioinformatics and computational biology [1, 2]. Computational protein structure prediction is useful for protein function study, protein design, protein engineering, drug design, and protein evolution analysis [3, 56]. It is becoming increasingly important in the post genomic era as millions of new protein sequences are produced by numerous DNA sequencing projects each year, leading to an enlarged knowledge gap between sequences and known experimental structures [57].

During the last few decades, numerous techniques were developed by scientists in multiple disciplines, such as biophysics, computational chemistry, computer science, and bioinformatics, to address different aspects of protein structure prediction. These aspects include secondary structure prediction, solvent accessibility prediction, disordered region prediction, domain boundary prediction, template identification, query-template alignment, template-based model generation, template-free model sampling, loop modeling, model/alignment quality assessment, and model refinement. Although not perfect, many of these methods can produce complementary and useful information to inform the final tertiary structure of a query protein [58, 59]. In addition to technological advances, increasing amounts of protein structures have been determined by experimental techniques and provide a rich set of structural data for enhancing protein structure prediction. Thus, it has become an important task to systematically integrate these diverse and complementary methods into a state of the art protein structure prediction system that can mine the enlarging protein sequence and structure databases to accurately and quickly predict the tertiary structure of any query protein [57, 91].

In order to integrate diverse protein structure prediction methods and multiple sources of information into one effective system, we have designed an open, five-layer, component-based protein structure prediction pipeline [37] that corresponds to the five major steps of protein structure prediction: template identification, query-template alignment and combination, model generation, model quality assessment, and model refinement. The components in the pipeline are loosely linked through information flow from one layer to next. The input to the pipeline is a query sequence and the output of the previous step is used as input to next step until the final structural models are produced from the pipeline. The interfaces between components are flexible and well designed, so that different methods developed for each step can be easily plugged into the system. Once the system is constructed under the open architecture, the next challenge is to benchmark the system and optimize a large number of parameters of the components. This system then selectively integrates the sequence and structural information produced by these components to generate final protein conformations of good quality. We blindly tested our current implementation of the system, MULTICOM, during the ninth Critical Assessment of Techniques for Protein Structure Prediction (CASP9, http://predictioncenter.org/casp9/) in 2010. The open system delivered very good performance. After the blind prediction phase of CASP9 ended, we systematically analyzed the intermediate data generated by each component in each prediction step and gained a great deal of experience about how to combine and configure components and integrate multiple sources of information in order to build a high-quality protein structure prediction system. In addition to present a comprehensive benchmark of the components of the MULTICOM system tested in CASP9, this work describes a number of new methodological developments occurred after it was first launched during the CASP8 experiment.

4.3 Methods

4.3.1 Overview of System Architecture, Design, and Implementation

Figure 4.1 illustrates the architecture of the MULTICOM protein structure prediction system [37]. The system consists of five major layers. The template identification layer accepts an input query sequence and searches it against a non-redundant protein sequence database to construct a query sequence profile. This profile is searched against a template library in order to identify a list of template protein structures that may provide conformation information about the structure of the query. A subset of top ranked templates and their sequence alignments with the query protein if available are fed into the template combination layer, which combines the structurally similar templates and the query into query-template alignments. The query-template alignments may contain more than one template which provides complementary information about the query. Then systematically combining of multiple templates generates a number of query-template alignments. The query-template alignments and template structures are fed into model generation tools (model generator) to sample conformations for the query. The regions of the query aligned with templates are sampled by a template-based model generator (e.g. a comparative modeling tool) and the large (>10 residues) unaligned query regions are sampled by a template-free model generator (e.g. a fragment-assembly tool). The model generators usually produce a number of models, which are then evaluated by the model quality assessment layer. The model quality assessment tools assign a global quality score to each model measuring its overall quality (e.g. overall similarity between the model and the known native structure) and a local quality score to each residue predicting its deviation compared with native structure. Finally, the models and their predicted quality scores are fed into the last model refinement layer in order to further improve their quality. In this layer, multiple models with similar conformations may be combined (e.g. averaged) and the low-quality regions of some models may be refined by stochastic simulations. At the end, the models with the best predicted qualities are released from the system as the final predictions.

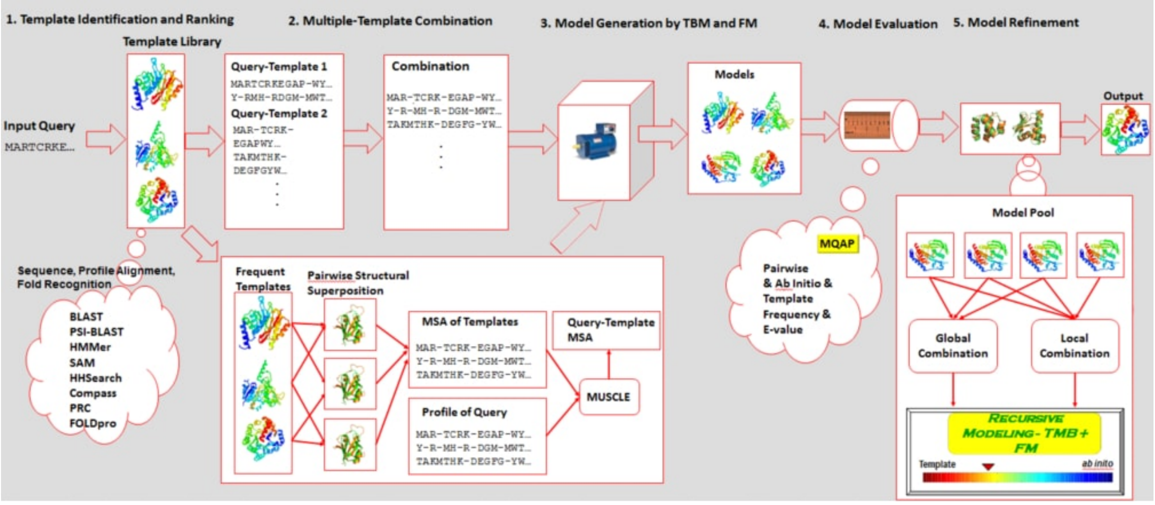


Figure 4.1: The five-layer architecture of the MULTICOM protein structure prediction system. TBM stands for template-based modeling and FM template-free modeling.

The open architecture of the protein structure prediction system makes it easy to plug in complementary methods as components and integrate multiple sources of information (e.g. template conformations) drawn from the template and sequence library / databases in order to produce high quality models. The subsections below present the implementation of the MULTICOM system emphasizing the new developments occurred since its first version [37] and the components that were thoroughly assessed in this work.

4.3.2 Template structure and sequence library

In order to support template-based structural prediction, a template library is constructed from the known experimental structures in the Protein Data Bank [67]. The template library includes template sequence, template structure (i.e. atom coordinates), secondary structure and solvent accessibility derived from the structure by DSSP [92, 93], and template sequential profiles. The template profiles are constructed from the multiple sequence alignment of the template sequence and its homologous sequences found by PSI-BLAST [64] when searching the template sequence against the Non-Redundant protein sequence database. The e-value cut off and the number of iterations of PSI-BLAST search range from 0.001 - 0.1 and 3 - 8, respectively, depending on the difficulty of the query. Different profiles such as HHSearch [94] hidden Markov model, COMPASS [72] profile, PRC [69] hidden Markov model, and PSI-BLAST [64] PSSM are created in order to facilitate a variety of profile-profile alignments. The HHSearch profiles also include the secondary structure information of the template proteins. Two lists of template sequences are created. The big list (LIB-A) essentially includes all the proteins ($\sim$60,000) in the PDB excluding identical proteins and short proteins ($<$30 residues) before the CASP9 experiment started. The small list (LIB-B) is a redundancy reduced list filtered at 90% sequence identify, which includes $\sim$20,000 proteins. In order to keep the library updated, the new protein structures released by the PDB are retrieved and incorporated into the library every week. Similarly, the non-redundant sequence database is updated weekly from the NCBI's web site.

4.3.3 Template identification

A query sequence is first searched against the Non-Redundant protein sequence database by PSI-BLAST [64] in order to find its homologous sequences. Query profiles (i.e., PSI-BLAST [64] PSSM, HHSearch [94] HMM, SAM [66] HMM, HMMER [42] HMM, PRC [69] HMM, and COMPASS [72] profile) are constructed from the query and its homologous sequences. Because the template identification is often sensitive to profile content, three kinds of HHSearch profiles are constructed for the query using the small, large, and filtered NR database. One special addition to the HHSearch profiles is that they include the secondary structure of the query protein predicted by either SCRATCH [95] or PSI-PRED [96]. In order to identify a list of template structures potentially relevant to the structure of a query protein, the sequence and its profile are searched against the template sequences and profiles. Specifically, the query sequence is searched against LIB-A using BLAST [64, 65] and CSI-BLAST [43]. The query PSSM, SAM, and HMMER profiles are searched against LIB-A by PSI-BLAST, SAM, and HMMER. The query HHSearch, PRC, and COMPASS profiles are searched against the profiles in LIB-B by HHSearch, PRC, and COMPASS. These searches are carried out by multiple threads in parallel. Each search may return a list of templates with e-values below a pre-defined threshold (e.g., 1 for hard targets and 0.001 for easy targets) and the local alignment between the query and templates is also generated. The top ranked template hits ranked by the e-values of the query-template alignments are retained for each method and the query-template alignments from the top hits identified by each method are stored in separate lists for later combination. Furthermore, the system counts the number of times a template was found by each alignment method and generates a consensus list of the top ranked (e.g. top 10) templates ranked solely by the frequency counts. The consensus template selection is a new addition to the MULTICOM system. CSI-BLAST, PRC, HMMER, and SAM are new alignment methods added into the system. It is worth noting that more sequence and profile alignment methods could be easily plugged into this layer, which often improves the performance of the system as multiple search tools often contribute complementary information or reinforce weak signals.

4.3.4 Multiple template combination

A template structure directly suggests a conformation that is supposed to be near the native conformation of the query protein being searched. This drastically reduces the search space. Multiple structurally similar templates may provide an ensemble of conformations that better confine the native structure of the query protein [29]. The multiple template combination layer is designed to integrate the structural information from multiple templates at the alignment level in order to reduce noise. Currently three multiple template combination methods are implemented. The first is the structure-alignment-guided, central-star, top-down approach combination method to integrate every list of query-template alignments directly generated by each search tool. The method first selects a top ranked query-template as a seed. Using the common query sequence as an anchor, it combines other template-query alignments ranked lower in the list with the seed if their e-values are close to the seed alignment and their aligned regions are structurally consistent with previously combined query-template alignments. The structural similarity of two query-template alignments is checked by comparing the structure of two templates which align to the same regions of the query (as determined by TM-align [97]). Two regions that could be structurally aligned with a high structural similarity score (i.e. GDT-TS score [31] > 0.75) are considered to be structurally consistent. The structural consistency check ensures the structural consistency of combined templates and improved model quality by avoiding or reducing atom clashes that result from the combination of structurally inconsistent templates. The second approach called "structure-alignment-driven profile alignment" is applied to the consensus list of templates that do not include query-template alignment information. The method can also generate structurally consistent alignments between a query and multiple templates. For each template in the list, the method first aligns its structure with that of each of the remaining templates using TM-align [97]. Each pairwise template-template structure alignment is converted into a pairwise sequence alignment by retaining only structurally aligned residues in the template. These pairwise sequence alignments between the common template and other templates in the list are combined into a multiple sequence alignment using the common template as an anchor. Because only those regions of the other templates that aligned well to the anchor template are kept, the multiple sequence alignment involving multiple templates is structurally consistent. The multiple sequence alignment (resp. HHSearch [94] profile) of these templates is then aligned with the multiple sequence alignment (resp. HHSearch profile) of the query to generate an alignment between the query and all the templates using the multiple sequence alignment tool MUSCLE (resp. HHSearch). The third approach is a hybrid alignment combination approach that gradually combines the alignments of a query-template pair generated by three different alignment methods: PSI-BLAST [64], HHSearch [94], and SPEM [98]. More specifically, this approach works by taking the PSI-BLAST alignment method first and then adding the HHSearch alignment for query regions not covered by PSI-BLAST alignment if available. Finally the SPEM global alignments are included for the rest of the uncovered query regions if available. The hybrid approach tries supplement the shorter, but likely more confident local alignments (e.g. PSI-BLAST) with longer, but perhaps less accurate global alignments (e.g. SPEM). Through the second and third methods, a list of combined query-template alignments is generated for the consensus template list. The two structure-alignment guided template combination methods that ensure the structural consistency among multiple templates and the hybrid combination method are the new development in the MULTICOM system.

4.3.5 Model generation

Each combined query-template alignment and the associated template structures are fed into model generators to sample conformations for the query protein. If one or more templates are found to cover the entire query protein leaving no unaligned region or very short unaligned regions (< 10 residues) the template-based modeling tool (Modeller 9v7 [25]) is used to generate a number of conformations (e.g. 10) for one set of input alignment and template structures. The model best fitting the restraints extracted from template structures is selected as the output model for the set of inputs. As such, a list of models will be generated for the list of input alignments and template structures. About 30-40% of the time, no homologous templates or only a template covering a part of the query protein is found, so a recursive protein modeling protocol [63] is used to integrate template-based modeling method and template-free modeling method to construct conformations that cover the entire query protein. Under this protocol, the certain regions of the query that align well with templates are first constructed by a comparative modeling tool - Modeller [25]. While keeping the conformations of template-based regions fixed and as restraints, a variant of the fragment-assembly tool (i.e. Rosetta [77]) is used to sample the conformations for the uncertain / unaligned regions. This method took the internal core region modeled by template-based modeling into consideration when calculating the energy while keeping the core rigid. This approach can integrate template-based and template-free modeling at a percentage from 0% to 100% depending on the amount of template information available. The conformations of certain and uncertain regions are then composed into a full model using Modeller. In the end, the model generation layer will produce a pool of candidate models (e.g. a few hundred) for the query protein. In this layer, the method of combining the template-based model and template-free model is a new addition.

4.3.6 Model quality assessment

The model quality assessment layer evaluates the quality of each model in the pool in order to select more accurate models. There are two kinds of model quality assessment (or model selection) methods, which can be referred to as the white box approach and the black box approach. The white box approach uses the information applied in generating a model to evaluate its quality. A typical method of the white box approach is an alignment-based model selection method [99] which uses the level of the similarity between query-template alignments (e.g. e-value of alignment score, sequence identity) to rank models generated from the alignments. The method of the black box approach uses the features extracted from the 3D shape of a model to assess its quality without exploiting any specific information about how the model is generated. In comparison to the scarcity of the white box methods, a variety of the black box model selection methods (e.g., energy-based methods [83, 100, 101], machine learning methods [82, 102, 103], and consensus methods [104, 105, 106, 107]) have been developed since the information related to how a model is generated is often not available. However, if there is such information, the white box approach tends to provide new insights into the quality of a model that might not be captured by the black-box methods.

Because there is no white-box model quality assessment method publicly available, we developed a support vector machine (SVM [108]) method to predict the quality score of a model based on the features extracted from the query-template pairwise sequence alignment employed to generate the model. The input features provided to the SVM predictor include the logarithm of e-value of the given query-template alignment, the percent of identical residue pairs in aligned positions, the percent of residues of the query that are aligned with a residue in the template, and the average of BLOSUM scores of all aligned residue pairs. From the input feature of a query-template alignment, the SVM predictor aims to predict the GDT-TS score of the model generated from the alignment. The input feature vectors in the training data set were extracted from 245 pairwise protein sequence alignments generated for 50 CASP9 targets by PSI-BLAST [64] and the output score of each input feature vector was the real GDT-TS score of its corresponding model calculated by the TM-score program [32]. This data were used to train a SVM regression predictor equipped with a Gaussian radial basis kernel (RBF) to predict the GDT-TS scores of models from the input features. The three parameters of the Gaussian radial basis kernel (RBF) to be tuned were the epsilon width of the regression tube (w), the margin-error tradeoff parameter (c), and the gamma of the RBF kernel (g). The root mean square error (RMSE) and the absolute mean error (ABS) between predicted and real GDT-TS scores were calculated for each set of parameter values to evaluate its performance. A five-fold leave-one-out cross validation (LOOCV) protocol was used to select the best parameter values of c from 2.0, 1.0, 0.5, 0.1, 0.05, 0.01, w from 0.5, 0.2, 0.1, 0.05, 0.02, and 0.01, and g from 0.5, 0.3, 0.2, 0.1, 0.05, 0.01, 0.005, and 0.001 according to the ABS and RMSE on all the five folds. The global average RMSE and ABS of the SVM trained with the best parameter values on the five-fold training data set were 0.083 and 0.061, respectively. The trained SVM predictor was applied to predict the GDT-TS scores of models of 46 CASP9 targets not used in training from the input features extracted from the corresponding PSI-BLAST alignments.

As model assessment is very challenging and none of the current methods can consistently select the best model, three model quality assessment methods (single-model approach, model pairwise comparison approach (APOLLO) [81], and a hybrid approach [37, 109]) are employed to assess the quality of the models in this layer. The single-model method (i.e. ModelEvaluator [82]) assigns an absolute quality score (e.g. GDT-TS score, the expected similarity between the model and the native structure) to each model by comparing the secondary structure, solvent accessibility, contact map, and beta-sheet topology of the model with that predicted from the query sequence [95, 110, 111]. This method is generally effective at discriminating good models from poor models. The pairwise comparison method (APOLLO) compares a model against all other models using a structure alignment tool (e.g. TM-score [32]) and calculates their similarity in terms of GDT-TS score, TM-score, and MaxSub score. The average similarity between a model and all other models is used as the predicted quality of the model. Note that the accuracy of the pairwise comparison method is input dependent (i.e. it works well only if the size of the model pool is large enough and the largest group of similar models in the pool are of good quality). The hybrid method is a compromise between the single-model method and the pairwise-comparison method. It first ranks the models by the quality scores predicted by

ModelEvaluator. The top several (e.g. 5) models are selected as reference models, against which each model is compared to. The average similarity between a model and the reference models is used as the quality score of the model. Furthermore, the average distance between a residue in a model and its counterpart in the reference models is used as the local quality of the residue (i.e. its deviation from the native structure). In addition to the three methods above, three simple scoring metrics were also tested, which included secondary structure scoring, secondary structure segment scoring, and solvent accessibility scoring. The secondary structure ranking method uses the percent of the secondary structures predicted from the sequence of a target that agree with those extracted from a model of the target to rank models. Higher a secondary structure agreement score, higher ranked a model. The idea of secondary structure segment score ranking is similar to the secondary structure ranking except the percent of agreement between secondary structure segments rather than between secondary structures of individual residues is used. Similarly, the solvent accessibility score ranking method uses the percent of the solvent accessibilities predicted from the sequence of a target that agree with those extracted from a model of the target to rank models. Higher a solvent accessibility alignment score, higher ranked a model. At the end of this layer, all models in the pool have been ranked by the quality scores predicted by these three scores. In this layer, the alignment-based model evaluation and the pairwise model evaluation are new developments in the system.

4.3.7 Model refinement

This last layer of the system uses a top-down local-global model combination approach to combine the top ranked models with other models that were globally very similar to it (e.g., pairwise GDT-TS score > 0.7) or combines very similar local regions of other models if no globally similar models were found. The model combination is essentially a model averaging process which in many cases can produce a model better than the top ranked model or even the best model in the pool. In addition to model combination, some regions of models are also refined according to the local quality. The poorly predicted local regions (e.g. tail regions) are resampled by a modified fragment-assembly method (a Rosetta variant), which keeps the other regions fixed and uses them as restraints to constrain the free modeling of the local regions. However, since some poorly predicted local regions are actually disordered regions, refinement on these regions cannot improve the global quality of the model. Finally the top refined models are released from the system as the final predictions.

According to the description of the five steps above, many database search / alignment tools are used in the MULTICOM protein structure prediction system. BLAST [64, 65] (Basic Local Alignment Search Tool) is a tool for finding local similarity between sequences. PSI-BLAST [64] (Position-Specific Iterative Basic Local Alignment Search Tool) is a tool for detecting distant relationships between proteins. COMPASS [72] is a tool for comparison of multiple protein alignments with assessment of statistical significance. HHSearch (version 1.2 and 1.5) [94] is a tool for detecting remote homologues of proteins and generating high quality alignments for homology modeling and function prediction. HMMER [42] is a tool for searching sequence databases for homologs of protein sequences and for finding protein sequence alignments using probabilistic models (profile HMMs). PRC [69] is a stand-alone tool for aligning and scoring two profile hidden Markov models. CS-BLAST [43] is an extension to standard NCBI BLAST that allows an increase in sensitivity by a factor of more than two at the same speed. CSI-BLAST [43] is an extension of CS-BLAST for iterative search with position-specific scoring matrices, two search iterations of which are more sensitive than five search iterations of PSI-BLAST. PSI-BLAST-multi is a top-down PSI-BLAST alignment combination approach to protein structure prediction and its assessments. SAM [66] (Sequence Alignment and Modeling system) is a profile HMM and sequence alignment tool. The alignments of all these tools except for BLAST and PSI-BLAST were combined into one-query and multiple-template alignment by the structure-alignment-guided, central-star, top-down approach for model generation. Individual BLAST and PSI-BLAST alignments were used for model generation. The consensus templates found by these alignment tools were used to generate query-template alignments by the structure-alignment-driven profile alignment approach. CENTER stands for one-query and multiple-template alignment by MUSCLE, while STAR stands for one-query and multiple-template alignment by HHSearch. CONSTRUCT denotes the hybrid query-template alignment derived from the PSI-BLAST, HHSearch and SPEM. The performance of these individual methods and their combination were discussed in the results and discussions section.

4.4 Results and discussions

4.4.1 System Testing, Integration, and Environment

As shown above, a sophisticated protein structure prediction system can be rather complicated and many choices and decisions must be made in each layer of the system. Thus integrating the components into one system that performs better than the simple sum of all the components is as critical as assembling computer components into a high-performance computer system. In order to objectively measure the performance of our integrated system, we blindly tested it in the 9th Critical Assessment of Techniques for Protein Structure Prediction (CASP9, http://predictioncenter.org/casp9/) in 2010. CASP9 released 129 protein targets whose structures were not available to the community. After some of the targets were canceled due to prematurely leaked information or difficulties in experimentally determining the structure, 107 official targets are available to assess the performance of the system. The set is sufficiently large and contained diverse types of protein topologies at different levels of difficulty, making it an ideal dataset to objectively benchmark the MULTICOM system. Four variants of the MULTICOM system participated in the CASP9 as four automated server predictors: MULTICOM-CLUSTER, MULTICOM-REFINE, MULTICOM-NOVEL, and MULTICOM-CONSTRUCT. The MULTICOM servers generated a large amount of intermediate data in each step of predictions. The raw data was analyzed in this work to study and compare the performance of the components of each layer during the CASP9 experiment. The analysis provided useful information for tuning the parameters of the components and the entire system.

The entire MULTICOM system was installed and run on a workstation with 8 cores, 8G of memory and a 1 TB hard disk during the CASP9 experiment. Essentially, the system can be installed and run on a modern PC. Generally, on the workstation, the system can make predictions for a query protein within a timeframe ranging from half an hour to several hours, depending on the length and the difficulty of the target. Prediction times for average-length template-based targets are shorter than average-length template-free targets because template-based targets do not require invoking the more time-consuming template-free modeling tools.

In order to investigate its design and performance, we evaluated the first four steps of the MULTICOM protein structure prediction system by comparing the templates, alignments, and models generated by all kinds of database search/alignment tools, comparing different model generation methods and comparing different model quality assessment tools.

4.4.2 Comparison of template identification methods

In order to evaluate all database search/alignment tools in the first step (i.e., template identification) we compared these tools from different aspects based on the templates identified by each of them. Firstly, the top 5 templates identified by two database search/alignment tools HHSearch [94] and PSI-BLAST-single for 107 CASP9 targets were aligned with the query's true structure, and their TM-scores were calculated using the TM-align program [97] in order to assess the performance of these two tools in template identification. TM-score [32] is a score in the range [0, 1] measuring the similarity between two protein structures, which is largely independent of protein length. Here, HHSearch and PSI-BLAST were compared because they are two typical profile-profile and profile-sequence alignment methods. Figure 4.2 illustrates the highest TM-scores of the top 5 templates identified by HHSearch and PSI-BLAST-single for 107 targets. HHSearch and PSI-BLAST-single identified the templates of the same quality for 25 targets. HHSearch obtained better templates for 60 targets, while PSI-BLAST-single recognized 22 better templates. It is consistent with previous observations that profile-profile alignment methods are more sensitive in recognizing templates than profile-sequence alignment methods. However, profile-sequence alignment can complement profile-profile alignment methods by identifying better templates in some cases.

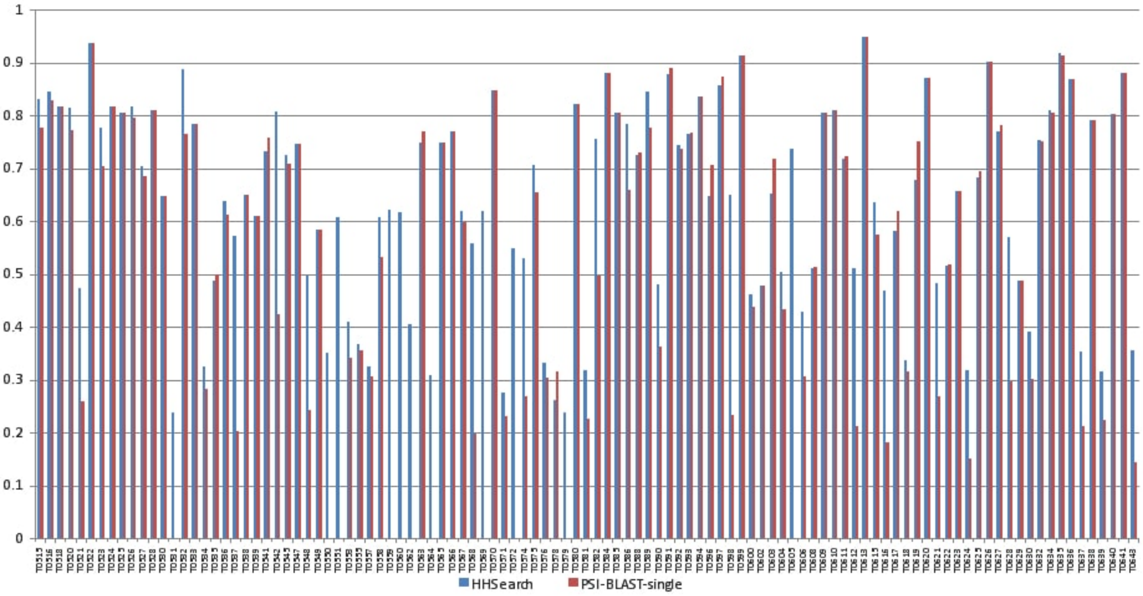


Figure 4.2: The highest TM-scores of the top 5 templates searched by HHSearch and PSI-BLAST-single for 107 CASP9 targets. Y axis represents TM-scores. X axis denotes the index of each target.

Then we evaluated all of the tools from another aspect by aligning the top 5 templates selected by them with the query's true structure for 107 CASP9 targets. Their similarities (i.e. TM-scores) were calculated using the TM-align program [97]. CONSTRUCT is a consensus template identification method that ranks templates based on the frequency of their selection by the other methods. PSI-BLAST-multi used PSI-BLAST to search a query against the NR database to build a PSSM profile and then searched the profile against the template library to select template structures. One difference between PSI-BLAST-multi and PSI-BLAST-single is that the latter searched the NR database for more iterations to include more remote homologous sequences into profile building. Another difference is that PSI-BLAST-multi combined the alignments between one query and multiple templates while PSI-BLAST-single only used one query-template alignment for model building. Figure 4.3 illustrates the total TM-scores (the addition of all TM-scores) of the top 1 template and the best template with the highest TM-score among the top 5 templates for each tool for 107 CASP9 targets. In both cases, two HHSearch-based profile-profile alignment methods (HHSearch and SS) delivered the best results, followed by the consensus methods (Center, Star, and SAM). Figure 4.4 illustrates the common and different sub-set of targets for which some good templates (TM-score $>$ 0.5) were identified when using HHSearch, CENTER, BLAST, and PSI-BLAST-single and demonstrated that these methods might identify a complementary set of templates.

Table 4.1 shows the specificity and sensitivity for the top 1 template and the best template among the top 5 templates for each tool and the number of targets that have templates identified for each tool. It shows that HHSearch, SS, CONSTRUCT, CENTER, and STAR found at least one template for each target of 107 targets. The templates found for around two thirds of the targets were good (TM-score $>$ 0.5). Although it only identified templates for 71 targets, PSI-BLAST-multi got the best specificity for the top 1 model and the best model, which means that the templates searched by PSI-BLAST-multi for more than 80% targets were good templates (TM-score $>$ 0.5) (see Table 4.1).

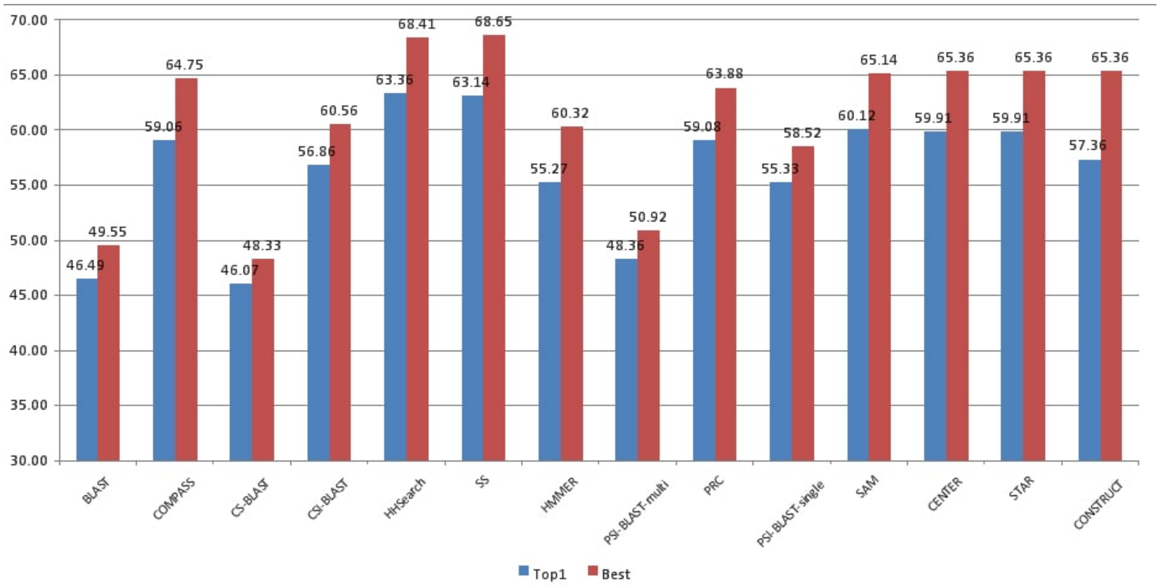


Figure 4.3: The total TM-scores of the top 1 template and the best template of each tool for 107 CASP9 targets. HHSearch is HHSearch version 1.2 and SS is HHSearch version1.5. PSI-BLAST-multi is the multi-template combination of PSI-BLAST alignment, which had higher total GDT-TS score than the single-template PSI-BLAST alignment approach. Here, the total TM-Score of the top-one templates is the sum of the TM-Scores of the no. 1 template identified for 107 CASP9 targets by a method by comparing the structure of each top-one template with the native structure. Similarly, the total TM-Score of the best templates is the sum of the TM-Scores of the best template identified for 107 CASP9 targets by a method by comparing the structure of the best template with the native structure.

4.4.3 Impact of alternative templates and alignments, alternative methods, structural consistency checking, and multiple-template combination on model accuracy

In order to explore the impact of multiple-template combination of all of the tools (BLAST [64, 65], CS-BLAST [43], CSI-BLAST [43], HHSearch [94] with different profiles, PRC [69], COMPASS [72], HMMER [42], SAM [66], PSI-BLAST-single, PSI-BLAST-multi, CONSTRUCT, CENTER, and STAR), the top 5 models generated by these tools for 107 CASP9 targets were superimposed onto the query's true structure,

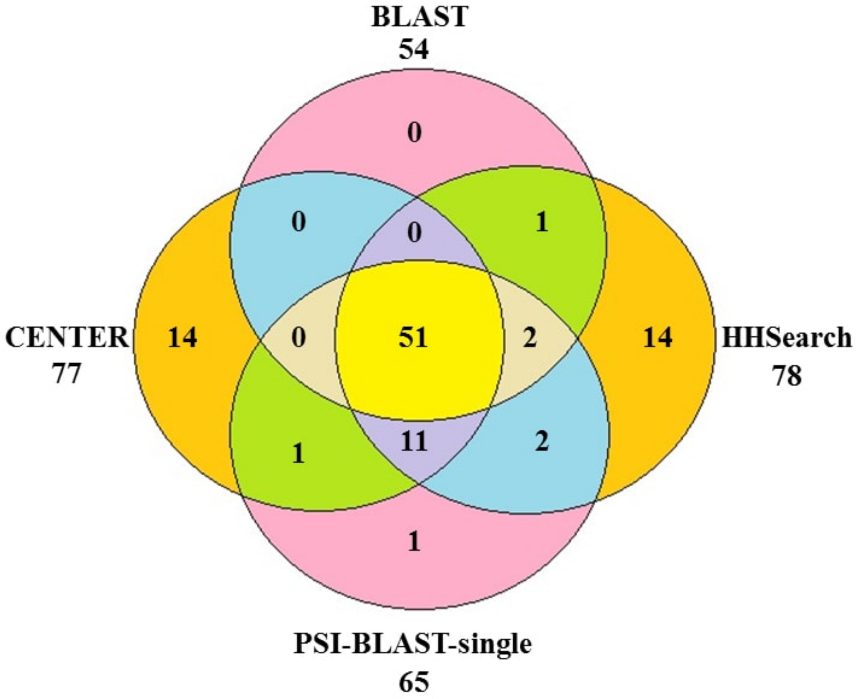


Figure 4.4: The common and different sub-set of targets for which some good templates (TM-score > 0.5) were identified.

and their GDT-TS scores were calculated by the TM-score program. GDT-TS (Global Distance Test) score is the average percent of residues in the model whose position is within 1, 2, 4, 8 Angstrom with that of their counterparts in the experiment structures after superposition [31]. Figure 4.5 reports the total GDT-TS scores of the top 1 models of each individual method and the total GDT-TS score of the top 1 models among all the models of all the methods. Figure 4.6 reports the total GDT-TS scores of the best models with highest GDT-TS score of each individual method and the total GDT-TS score of the best model with the highest GDT-TS score among all models of all the methods. As shown in Figures 4.5 and 4.6, the score of HHSearch 1.5 (i.e. SS) on a filter profile is slightly higher than the ones of other tools, which reveals this method generated better target-template alignments. However, the total score of the method was still a few percent lower than the total score of top ranked or the best models generated from the target-template alignments of all the methods, suggesting that pooling models generated from alternative target-template alignments produced by the different methods improved model quality.

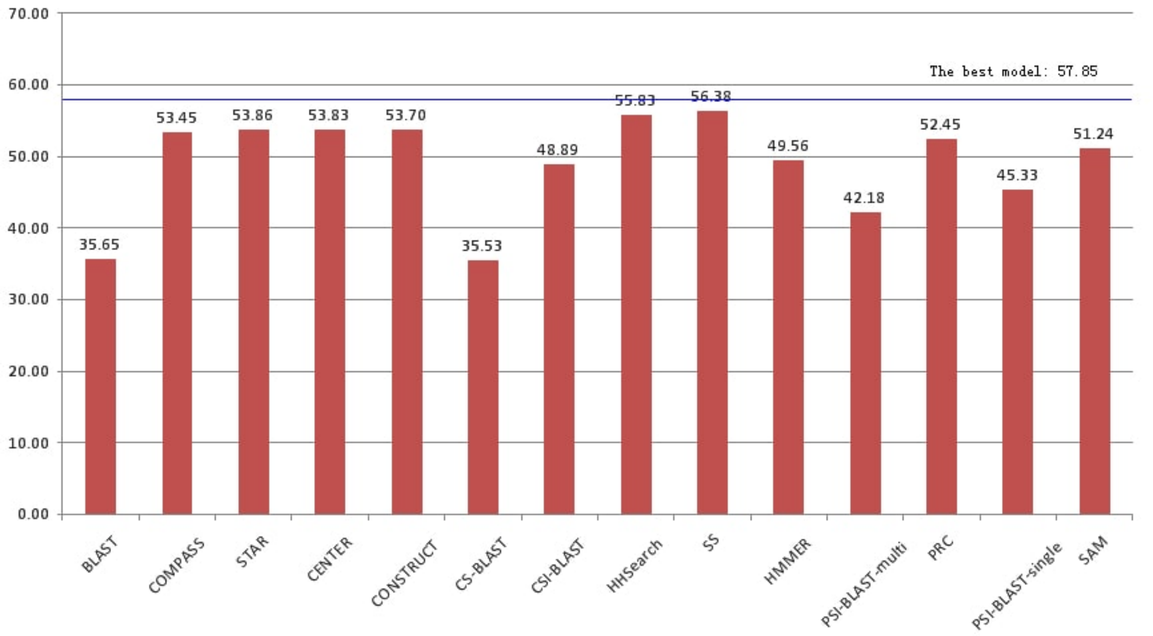


Figure 4.5: The total GDT-TS scores of the top 1 ranked model of each individual method and the top 1 ranked models of all of the methods for 107 CASP9 targets. The vertical bars represent the total scores of individual methods. The blue line denotes the total score of top 1 model of all the methods.

Table 4.2 shows the total GDT-TS scores of PSI-BLAST-multi and PSI-BLAST-single for the top 1 model and the best model on the same set of 71 targets for which both methods made predictions. The results show that PSI-BLAST-multi has a slightly better performance than PSI-BLAST-single. However, it was hard to quantify the contributions of multiple template combination here because the templates used for each target by the two methods may be different.

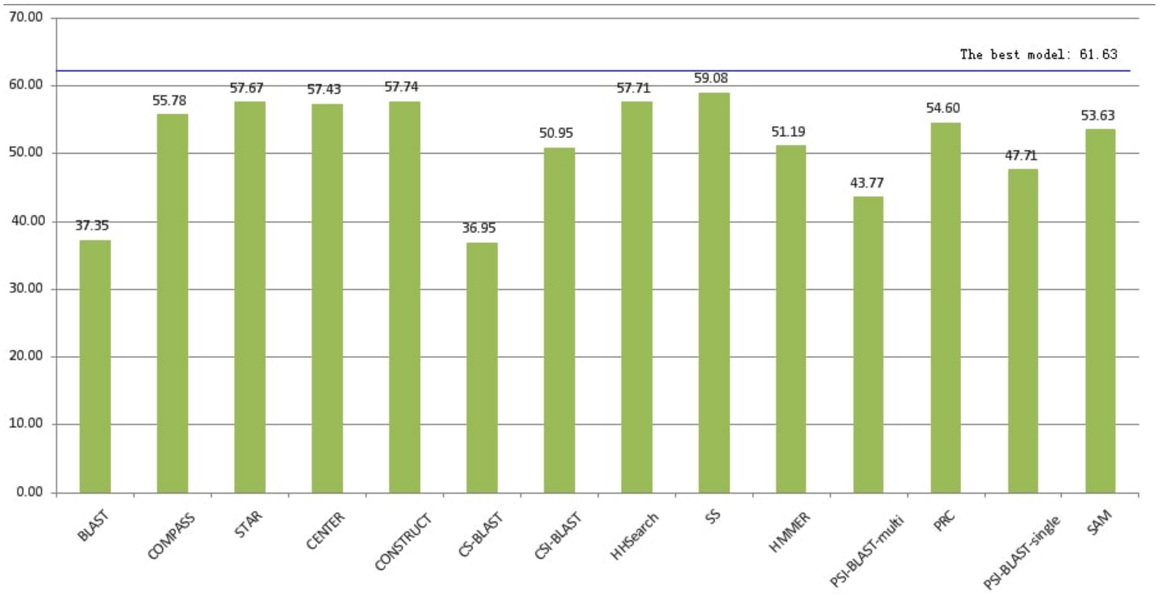


Figure 4.6: The total GDT-TS scores of the best model of each individual method and the best model of all the methods for 107 CASP9 targets. The vertical bars represent the total scores of individual methods. The blue line denotes the total score of the best model of all the methods.

In order to investigate the impact of structural consistency checking for HHSearch modeling, we assessed and compared three kinds of HHSearch [94] models (i.e. HH with structural consistency checking, SS with structural consistency checking, and HS without structural consistency checking). All of the generated models of HH, SS, and HS for 107 CASP9 targets were aligned with the query's true structure, and their GDT-TS scores were calculated using the TM-score program [88]. The total GDT-TS scores of the best models of HH and SS with structural consistency checking are 57.77 and 59.2 respectively, clearly higher than that of HS without the consistency check which scores 52.44. Despite some difference in HHSearch versions, profiles, and other parameters, this may still imply that methods with structural consistency checking have better performance than methods without a structural consistency check.

STAR models (HMM), CENTER models (MUSCLE), and CONSTRUCT models were compared in order to assess the quality of the multiple sequence alignments generated. All of the generated models of STAR, CENTER, and CONSTRUCT for 107 CASP9 targets were aligned with the query's true structure and their GDT-TS scores were calculated using the TM-score program [88]. The total GDT-TS scores of the best models of STAR, CENTER, and CONSTRUCT with highest GDT-TS score for 107 CASP9 targets are 57.67, 57.43, and 59.07 respectively (see Figure 4.6), whereas the total GDT-TS scores of the top 1 ranked models of these methods are similar (see Figure 4.5).

4.4.4 Comparison of Model Generation Protocols

We compared the performance of the ab initio model generation method and the template-based method on hard targets by comparing HHSearch models, SS models and ab initio models. Hard targets are template-free targets that did not have a reasonable template in the protein structure database. All of the generated models of HHSearch, SS, and ab initio for 8 CASP9 hard targets [112] were aligned with the query's true structure and their GDT-TS scores were calculated using the TM-score program [88]. The total GDT-TS score of the best models of ab initio with highest GDT-TS score is 2.55, clearly higher than 1.88 of HHSearch and 1.79 of SS. This suggests that the ab initio models generated by the fragment assembly based ab initio method were better than the models generated by the template-based method with incorrect templates.

We further compared four template-based model generation protocols (i.e. auto model, loop model, dope_loop model, and dope_hr_loop model) of Modeller [25].

All of the models generated by these four protocols from HHSearch [94] alignments for 107 CASP9 targets were aligned with the query's true structures. Their GDT-TS scores were calculated using the TM-score program [88]. Table 4.3 illustrates the total GDT-TS scores of the best models with highest GDT-TS score generated by these protocols. It was quite surprising that the total GDT-TS score of the simplest auto model protocol is clearly higher than the other, more advanced protocols.

4.4.5 Comparison of Model Selection Methods

We evaluated two kinds of model quality assessment methods (the white box approach and the black box approach) on the CASP9 targets. We applied the SVM alignment-based predictor (the white box approach) trained on alignments of 50 CASP9 targets to blindly score the models generated from PSI-BLAST-single alignments of the other 46 CASP9 targets. The total real GDT-TS score of the top 1 models selected by the SVM predictor for these targets was compared with that of the top 1 models simply ranked by the e-values of the PSI-BLAST alignment. The total GDT-TS score of the models selected by the SVM predictor is 20.95, higher than 20.10 of the naive e-value based model selection method. Moreover, A t-test and a wilcox-test were performed to check if the two scores are significantly different (p-value$<$0.05). The p-value of t-test is 0.044 and the p-value of wilcox-test 0.042. The results seem to show that incorporating multiple alignment features in a SVM can significantly improve model selection over the naive e-value based method.

As for the black box model selection methods, we evaluated a single-model absolute model quality predictor (ModelEvaluator), the secondary structure score ranking method, the solvent accessibility score ranking method, the secondary structure segment (SOV) score ranking, a pairwise model comparison method (APOLLO), and an energy ranking method (SELECTpro [83]). APOLLO generated three kinds of scores for a model, i.e. TM-score, GDT-TS score, and Max-Sub score, and these were evaluated separately. All these methods were used to select one model with the highest predicted score from all the models predicted for each of the CASP9 target. The total real GDT-TS scores of the models selected by each method is reported in Figure 4.7. The results show that ModelEvaluator yielded the best performance, which is only slightly better than that of SELECTpro and APOLLO. The performance of these three comprehensive quality predictors was substantially better than that of the ranking based methods on a single feature (i.e., SS, SA, SOV).

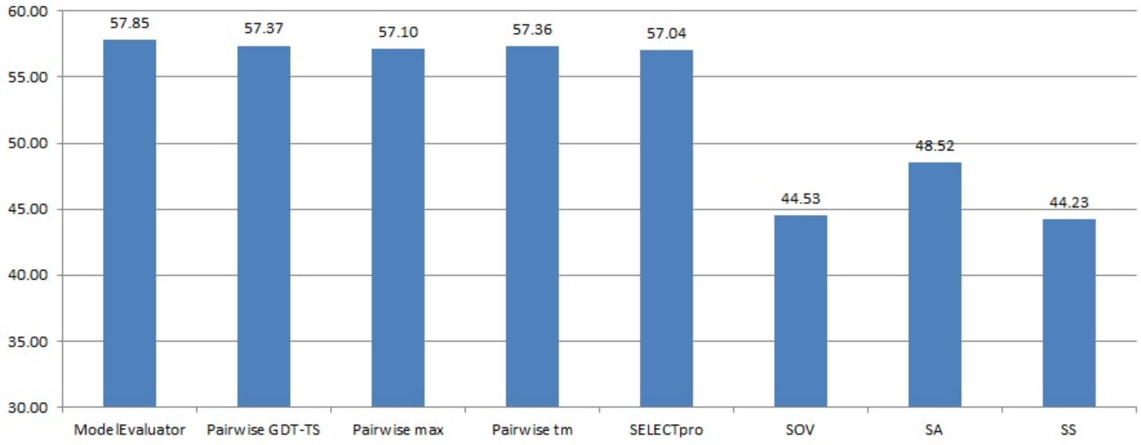


Figure 4.7: The total GDT-TS scores of the top models selected by different model-ranking technologies for 107 CASP9 targets.

In addition to evaluating the quality of a model based on the coordinates of all of its residues, we investigated if removing potentially disordered regions from full-length models could improve model quality assessment. In contrast to previous work that excluded potentially disordered residues from model generation resulting in a partially constructed model, our approach removes them from a full-length model containing all the residues in order to improve the accuracy of evaluating its quality. We used PreDisorder [113] to predict the putative disordered residues of each target and then filtered out the coordinates of the N-/C-terminal disordered residues from all the models. ModelEvaluator, APOLLO, and SELECTpro were used to assess the filtered models and to select one model with the highest score from all the filtered models for each of the CASP9 target. The performance of these methods applied to the filtered models was compared with that of the same methods when applied to the full-length models. The total real GDT-TS scores of the best models selected by these methods are reported in Table 4.4. The results show that removing N/C-terminal disordered regions from full-length models improves the performance of all the quality assessment methods. The improvement on the pairwise quality assessment method (Apollo) and the energy-based method (SELECTpro) was more pronounced, indicating that these methods were more sensitive to the noise caused by the disordered residues than ModelEvaluator. Overall, our experiment suggests that disorder prediction may help significantly improve model ranking, which has been a long-standing challenging problem.

4.4.6 Impact of model combination on model quality

In order to assess the impact of the simple model combination method on model quality, we compare the total GDT-TS score, TM-Score and MolProbity score of the combined models with those of the top ranked models of 107 CASP9 targets (see Table 4.5). Different from that GDT-TS score and TM-Score measures the accuracy of the backbone of a model, MoProbity evaluates how realistic a model is according to its all-atom conformation. The results show that the GDT-TS scores and TM-

Scores of the combined models and the top ranked models are almost the same, the MolProbity score of the former is better (i.e. lower) than that of the latter, suggesting combining models may make models more protein-like.

4.5 Conclusion and future work

Developing high-quality protein structure prediction systems is critical for addressing the protein structure challenges faced in the post genomic era. In this work, we described how to construct a protein structure system (MULTICOM) under a five-layer open architecture, which can integrate complementary component methods and multiple sources of information to reliably and accurately predict protein structure from sequence. We focused on investigating and validating the effectiveness and complementarity of different components employed in each layer. The experiments provided insights about how to select, use, and combine existing techniques to improve protein tertiary structure prediction under an open architecture. Additionally, the experiments provide a direct, comprehensive and quantitative assessment of various components of a single protein structure prediction system in a blind prediction setting and some interesting findings such as the impact of protein disorder prediction on protein model selection. These results shed new light on designing and developing better protein structure prediction systems and algorithms.

However, despite the reasonable performance of the MULTICOM protein structure prediction system achieved on most protein targets, our benchmark suggests there is still the room of improvement in each step of protein structure prediction process. In the future, we plan to add more sensitive or complementary template identification methods into the system to address the failure of identifying good templates for some hard targets. These improvements will include more complementary or even better alignment methods to generate more accurate target-template alignments, improve alignment-based model quality assessment methods with more features and multiple-template information, incorporate residue-residue contact information to improve ab initio model generation (i.e., a major bottleneck of protein structure prediction), and explore the usage of residue disorder prediction in both template-based and ab initio model generation.

Tool	The top 1 model		The best model		# targets with templates
	Specificity	Sensitivity	Specificity	Sensitivity	
PSI-BLAST -multi	80.28%	53.27%	88.73%	58.88%	71
CS-BLAST	73.97%	50.47%	78.08%	53.27%	73
CENTER	67.29%	67.29%	71.96%	71.96%	107
STAR	67.29%	67.29%	71.96%	71.96%	107
HMMER	66.67%	56.07%	77.78%	65.42%	90
SS	66.04%	65.42%	71.96%	71.96%	107
HHSearch	65.42%	65.42%	72.90%	72.90%	107
BLAST	65.38%	47.66%	69.23%	50.47%	78
CSI-BLAST	62.63%	57.94%	66.67%	61.68%	99
COMPASS	62.50%	60.75%	71.15%	69.16%	104
PSI-BLAST -single	62.50%	56.07%	67.71%	60.75%	96
PRC	62.14%	59.81%	69.90%	67.29%	103
SAM	61.32%	60.75%	67.92%	67.29%	106
CONSTRUCT	60.75%	60.75%	71.96%	71.96%	107

Table 4.1: The specificity and sensitivity for the top 1 template and the best template among the top 5 templates for each tool based on 107 CASP9 targets and the number of targets that have templates identified for each tool. The specificity is the fraction of the targets with at least one template identified by a method having a GDT-TS score >= 0.5, i.e. the number of targets for which a good template (i.e. its GDT-TS score >= 0.5) is identified by the method divided by the number of targets for which at least one template is identified. The specificity measures the precision of template identification of a method. The sensitivity is the number of targets for which a good template (i.e. its GDT-TS score > 0.5) is identified by a method divided by all the targets in consideration in this experiment (i.e. 107), assuming that all the targets have at least one reasonable template. The two measures (i.e. sensitivity and specificity) are complementary.

Tool	Total GDT-TS score	
	The top 1 model	The best model
PSI-BLAST-multi	42.18	43.77
PSI-BLAST-single	41.51	43.33

Table 4.2: The total GDT-TS scores of PSI-BLAST-multi and PSI-BLAST-single on same set of 71 targets for which both methods made predictions.

Method	The total GDT-TS score
auto model	53.55
loop model	48.41
dope_loop model	47.95
dope_hr_loop model	48.04

Table 4.3: The total GDT-TS scores of the best models generated by four model generation protocols for 107 CASP9 targets.

Model	The total GDT-TS score				
	ModelEvaluator	APOLLO			SELECTpro
		tm	max	GDT-TS	
The best model without the tail disorder regions	57.88	61.12	60.92	61.01	59.94
The best model with the tail disorder regions	57.85	57.36	57.10	57.37	57.04

Table 4.4: The total GDT-TS scores of the best models without the tail disorder regions and the best models with the tail disorder regions for 107 CASP9 targets.

Models	TM-score	GDT-TS score	MolProbity score
The combined, refined models	64.20	57.14	340.98
The top selected models	64.28	57.21	351.18

Table 4.5: The total TM-score, GDT-TS score, and MolProbity score of the combined, refined models and top selected models of 107 CASP9 targets.

Chapter 5

From Gigabyte to Kilobyte: A Bioinformatics Protocol for Mining Large RNA-Seq Transcriptomics Data

5.1 Abstract

RNA-Seq techniques generate hundreds of millions of short RNA reads using next-generation sequencing (NGS). These RNA reads can be mapped to reference genomes to investigate changes of gene expression but improved procedures for mining large RNA-Seq datasets to extract valuable biological knowledge are needed. RNAMiner - a multi-level bioinformatics protocol and pipeline - has been developed for such datasets. It includes five steps: mapping RNA-Seq reads to a reference genome, calculating gene expression values, identifying differentially expressed genes, predicting gene functions, and constructing gene regulatory networks. To demonstrate its utility, we applied RNAMiner to datasets generated from *Human*, *Mouse*, *Arabidop-*

sis thaliana, and *Drosophila melanogaster* cells, and successfully identified differentially expressed genes, clustered them into cohesive functional groups, and constructed novel gene regulatory networks. The RNAMiner web service is available at http://calla.rnet.missouri.edu/rnaminer/index.html.

5.2 Introduction

Transcriptome analysis is essential for determining the relationship between the information encoded in a genome, its expression, and phenotypic variation [16, 17]. Next-generation sequencing (NGS) of RNAs (RNA-Seq) has emerged as a powerful approach for transcriptome analysis [14, 15] that has many advantages over microarray technologies [18, 114, 115].

A RNA-Seq experiment typically generates hundreds of millions of short reads that are mapped to reference genomes and counted as a measure of expression [14, 15]. Mining the gigabytes or even terabytes of RNA-Seq raw data is an essential, but challenging step in the analysis.

In order to address these challenges, RNAMiner has been developed to convert gigabytes of raw RNA-Seq data into kilobytes of valuable biological knowledge through a five-step data mining and knowledge discovery process. RNAMiner integrates both public tools (e.g., TopHat2 [116], Bowtie2 [117], Cufflinks [118], HTSeq [119], edgeR [120], and DESeq2 [121]) with our in-house tools (MULTICOM-MAP [122, 123, 124]) to preprocess data and identify differentially expressed genes in the first three steps. In the last two steps, RNAMiner uses our in-house tools MULTICOM-PDCN [125, 126] and MULTICOM-GNET [127] to predict both functions and gene regulatory networks of differentially expressed genes, respectively.

As proof of principle, we have applied the RNAMiner protocol to RNA-Seq data generated from *Human*, *Mouse*, *Arabidopsis thaliana*, and *Drosophila melanogaster* cells. The data mining process successfully produced valuable biological knowledge such as differentially expressed genes, cohesive functional gene groups, and novel hypothetical gene regulatory networks by reducing the size of the initial data set over a thousand-fold.

5.3 Methods

Some RNA-Seq data analysis pipelines (e.g. Galaxy [128], KBase, iPlant [129]) provide users with a convenient and free platform for RNA-Seq data analysis by combing public tools, such as TopHat [130], Bowtie [131], Cufflinks [118], Cuffmerge [118], and Cuffdiff [118]. As with these pipelines, RNAMiner combines these public tools such as TopHat2 [116], Bowtie2 [117], Cufflinks, Cuffdiff, and it is free. However, there are several differences between RNAMiner and other pipelines. First, RNAMiner integrates more tools, such as HTSeq [119], edgeR [120], DESeq2 [121], and our in-house MULTICOM-MAP [122, 123, 124], to calculate gene expression values and identify differentially expressed genes. These tools can generate more accurate consensus results. For example, RNAMiner uses Cuffdiff, edgeR, and DESeq2 to identify differentially expressed genes based on TopHat mapping results and gene expression values calculated by HTSeq and MULTICOM-MAP. RNAMiner generates up to five distinct lists and one consensus list of differentially expressed genes, which usually produces more accurate results. Second, RNAMiner predicts functions of differentially expressed genes and builds gene regulatory networks by integrating our in-house tools MULTICOM-PDCN [125, 126] and MULTICOM-GNET [127]. These analyses provide more biological information. Other pipelines (e.g. Galaxy and iPlant) do not provide these analyses. Another software package - KBase - contains a service to predict gene functions, but the service only provides GO annotation for plant genomes. Third, without requirements for user registration and selection of many parameters, RNAMiner is easier to use than other pipelines. Compared to running each tool separately, users can easily run all these tools integrated in RNAMiner at one time and download results generated by all the tools at the RNAMiner web site.

The five data analysis steps of the RNAMiner protocol (Figure 5.1) are described individually in sub-sections below. Tables 5.1 and 5.2 list the versions and the parameters of all the public tools used in RNAMiner and describe the meanings of the parameters.

Tool	Version
TopHat2	2.0.6
Bowtie2	2.1.0
Cufflinks	2.2.1
HTSeq	0.5.3p7
edgeR	3.4.2
DESeq2	1.2.10

Table 5.1: The versions of the public tools used in RNAMiner.

5.3.1 Mapping RNA-Seq reads to a reference genome

We use two public tools, TopHat2 [116] and Bowtie2 [117], to map RNA-Seq reads to reference genomes in the UCSC genome browser [132] in conjunction with the RefSeq genome reference annotations [133]. The workflow of mapping RNA-Seq reads to

Tool	Parameter	Value
TopHat2	–read-mismatches	2
	–read-gap-length	2
	–splice-mismatches	0
	–segment-mismatches	2
	–segment-length	25
Bowtie2	–end-to-end	
	–sensitive	
	–frag-len-std-dev	80
	–min-isoform-fraction	0.10
	–pre-mrna-fraction	0.15
	–max-intron-length	300000
Cuffdiff	–min-alignment-count	10
	–FDR	0.05
	–frag-len-mean	200
	–frag-len-std-dev	80
HTSeq	-a	10
	-i	gene_id
	-m	Union
DESeq2	Test	LRT
	fitType	parametric
edgeR	Pair	NULL
	Dispersion	NULL
	common.disp	TRUE

Table 5.2: The parameters of the public tools used in RNAMiner.

a reference genome and calculating gene expression values is illustrated in Figure 5.2. It is worth noting that, since the RefSeq genome reference annotations contain information about some non-coding small RNAs, the reads of the non-coding RNAs are mapped and counted in addition to regular protein coding mRNAs. MULTICOM-MAP [122, 123, 124] is used to remove reads mapped to multiple locations in a reference genome from the mapping data in BAM/SAM format [134] generated by TopHat2 and Bowtie2. Only reads mapped to a unique location on the genome are retained to calculate the read counts of the genes. We use MULTICOM-MAP to analyze the mapping results to obtain baseline information, such as the total number of reads, the number of reads mapped to a unique location, and the number of reads mapped to multiple locations. This mapping process can generally reduce the size of datasets by several orders of magnitude.

5.3.2 Calculating gene expression values

For RNAMiner, MULTICOM-MAP [122, 123, 124] and two public tools: HTSeq [119] and Cufflinks [118] are used to calculate gene expression values according to the genome mapping output and the RefSeq genome reference annotation [133]. MULTICOM-MAP and HTSeq produce raw read counts, while Cufflinks generates normalized values in terms of FPKM, i.e., fragments per kilobase of exon model per million mapped fragments. The normalized gene expression values generated by Cufflinks are used to identify differentially expressed genes in the next step. The read counts generated by MULTICOM-MAP and HTSeq are fed separately into two R Bioconductor packages, edgeR [120] and DESeq2 [121], to identify differentially expressed genes. The normalized gene expression values (RPKM, reads per kilobase of exon model per million mapped reads) of MULTICOM-MAP are used to construct gene regulatory networks in the last step. Cufflinks, MULTICOM-MAP, and HTSeq are largely complementary and mostly differ in how they handle the reads mapped to common exons of multiple isoforms of a gene. Cufflinks distributes the count of such reads to each isoform proportionally according the estimated probability that the reads were derived from the isoform. In contrast, MULTICOM-MAP distributes the total count of such reads to each isoform, while HTSeq discards the reads without counting them for any isoform. This analysis step generates the overall expression profile of most genes in a transcriptome and can reduce the size of data from Step 1 by around one thousand-fold, from gigabytes to several megabytes.

5.3.3 Identifying differentially expressed genes

We use Cuffdiff [118] and two R Bioconductor packages, edgeR [120] and DESeq2 [121] to identify differentially expressed genes separately (see Figure 5.3 for the workflow). EdgeR and DESeq2 identify differentially expressed genes based on the raw read counts calculated by MULTICOM-MAP and HTSeq, resulting in four lists of differentially expressed genes (i.e., edgeR + MULTICOM-MAP, edgeR + HTSeq, DESeq2 + MULTICOM-MAP, and DESeq2 + HTSeq). In contrast Cuffdiff identifies differentially expressed genes directly from the genome mapping outputs containing only reads mapped to a unique location on the genome, resulting in one list of differentially expressed genes. Cuffdiff, edgeR and DESeq2 further adjust p-values by multiple testing using Benjamini and Hochberg's approach, which controls the false discovery rate (FDR). Usually, the cut-off of p-value (or q-value) is set to 0.05. Based on the five lists of differentially expressed genes generated by Cuffdiff, edgeR + MULTICOM-MAP, DESeq2 + MULTICOM-MAP, edgeR + HTSeq, and DESeq2 + HTSeq, a consensus list of differentially expressed genes is generated as the final output which usually comes from the overlap of at least three lists of differentially expressed genes. This step generates valuable information that may play an important role in the biological experiment. For example, the significantly differentially expressed genes identified by RNAMiner could be the targets for new biological experiments. This analysis step can generally reduce the size of data of the previous step by a couple orders of magnitude, condensing the data set size to several hundred kilobytes.

5.3.4 Predicting gene functions

We use MULTICOM-PDCN [125, 126], a protein function prediction method ranked among the top methods in the 2011-2012 Critical Assessment of Function Annotation (CAFA) [135], to predict functions of differentially expressed genes (see Figure 5.4 for the workflow). MULTICOM-PDCN integrates sequence-profile and profile-profile alignment methods (PSI-BLAST [64] and HHSearch [94]) with protein function databases such as the Gene Ontology database [136], the Swiss-Prot database [137], and the Pfam database [138], to predict functions of proteins in Gene Ontology [136] terms in three categories: biological process, molecular function, and cellular component. MULTICOM-PDCN also provides some statistical information about the predicted functions, such as the number of differentially expressed genes predicted in each function. We then use the Cochran-Mantel-Haenszel test implemented by R program mantelhaen.test [139, 140] to check if predicted function terms are good for Fisher's exact test to identify the significantly enriched GO function terms. A p-value from the MH test lower than 0.05 suggests the two nominal variables (e.g., two function terms) are conditionally independent in each stratum [139, 140]. We then calculate a p-value of enrichment for each predicted function using R function fisher.test [139, 140, 141, 142, 143, 144] and sort the predicted functions by their p-value in ascending order, from the most significant ones to the least significant ones. The list of the most significantly enriched functions can provide an overview of the biological processes differentially perturbed in two biological conditions. Although the physical size of the data and knowledge generated in this step is comparable to the size of the data in the previous step, the differentially expressed genes can be organized in three functional perspectives: biological process, molecular function, and cellular component.

5.3.5 Constructing gene regulatory networks

We use MULTICOM-GNET [127] to construct gene regulatory networks based on differentially expressed genes and transcription factors in a genome (see Figure 5.5 for the workflow). MULTICOM-GNET firstly clusters differentially expressed genes with similar expression patterns into functional clusters using the K-means clustering algorithm. Secondly, it builds a binary decision tree to represent potential regulatory relationships between several selected transcription factors (TFs) and the genes in each cluster. Thirdly, it re-assigns differentially expressed genes into clusters whose gene regulatory tree best explained the expression patterns of the genes. The last two steps are repeated until the maximized likelihood of the gene expression data is reached. We also use a R network analysis and visualization package "igraph" [145] to visualize gene regulatory networks by linking the regulatory relationships between and within all the gene regulatory modules predicted by MULTICOM-GNET together. The regulatory network construction step generates a comprehensive understanding of underlying mechanisms controlling the expression of a transcriptome and can significantly reduce the size of data. The hundreds of kilobytes of the biological network data provide a system view of the cellular systems, which can be more readily utilized to generate valuable hypotheses for biological experiments.

For replicates from RNA-Seq experiments, RNAMiner maps reads of the replicates to reference genomes and calculates gene expression values separately. The gene expression values of the replicates of two samples are combined into a profile (i.e. a vector of the expression values of a gene in each replicate of each condition), which is input into edgeR and DESeq2 to identify differentially expressed genes. Additionally, the TopHat mapping results of the replicates of two samples are input into Cuffdiff to identify differentially expressed genes. EdgeR, DESeq2, and Cuffdiff handle the replicates by modeling the variance (dispersion) in counts across the replicates as a function of the mean count of the replicates. EdgeR [120] estimates the variance by conditional maximum likelihood conditioned on the total count for the gene. DESeq2 [121] uses a flexible and mean-dependent local regression to estimate the variance between the replicates by pooling genes with similar expression levels to enhance the variance estimation. Cuffdiff [118] estimates the variance based on a negative binomial model and uses t-test to calculate the test statistics. Cuffdiff can make a model on each condition with replicates, or use a global model for all conditions together.

Before calling a tool to do data analysis, RNAMiner checks whether the data is appropriate to the tool. For example, MULTICOM-GNET is not applied if no transcription factors exist in differentially expressed genes because MULTICOM-GNET needs at least one transcription factor to build gene regulatory networks. Another example is, for some special datasets, overexpression of some treatments in some regions of the genome in one condition leads to very large read counts of some genes in this condition, and dramatic differences of gene expressions between two conditions. This violates the assumption of edgeR's normalization method [120] that the majority of the genes should have similar expression levels. Therefore, calculating a normalization factor across all loci is difficult. RNAMiner will check this assumption and will not call edgeR if it is violated.

5.4 Evaluation and Discussion

We tested the RNAMiner protocol on six sets of RNA-Seq data generated from *Human*, *Mouse*, *Arabidopsis thaliana* and *Drosophila melanogaster* cells in order to evaluate its effectiveness. The details such as organisms, biological conditions, and experimental settings about the six sets of RNA-Seq data were reported in Table 5.3. The results of each of the five analysis steps are described and discussed as follows.

Data set	Organism	Conditions	Replicates
First	*Mouse*	Control, two botanicals (Lessertia frutescens and Sambucus nigra), and Nrf2 activator CDDO (2-cyano-3, 12-dioxooleana-1, 9-dien-28-oic acid)	No
Second	*Mouse*	FruHis (0, 1, 2, 4, 8, 16 mM) in the absence (samples 2A, 2B, 2C, 2D, 2E, 2F) or presence (samples 2G, 2H, 3A, 3B, 3C, 3D) of 4 μM lycopene	No
Third	*Drosophila melanogaster*	CF (Control Female), CM (Control Male), HMF (H83M2 Female), and HMM (H83M2 Male)	Three
Fourth	*Drosophila melanogaster*	CF (Control Female), CM (Control Male), and mF (Meta Female)	Three
Fifth	*Arabidopsis thaliana*	Columbia wild-type and hae-3 hsl2-3 mutants	Three
Sixth	*Human*	1Sfesrrb, 2pc3, 3DY131, and 4ctrl	Two

Table 5.3: The organisms, conditions, and replicate number of the six sets of RNA-Seq data.

5.4.1 Results of mapping RNA-Seq reads to a reference genome

RNAMiner used TopHat2 [116] and Bowtie [146] to map RNA-Seq reads in the first and second data sets to the *Mouse* reference genome (mm9) in the UCSC genome browser [132] in conjunction with the RefSeq genome reference annotation (mm9) [133], map RNA-Seq reads in the third and fourth data sets to the *Drosophila melanogaster* reference genome (dm3) in the UCSC genome browser in conjunction with the RefSeq genome reference annotation (dm3), map RNA-Seq reads in the fifth data set to the *Arabidopsis thaliana* reference genome (ftp://ftp.arabidopsis.org/home /tair/Sequences/whole_chromosomes/) in conjunction with the *Arabidopsis thaliana* genome reference annotation (ftp://ftp.arabidopsis.org/home/tair/Genes/TAIR10_g enome_release/TAIR10_gff3/), and used TopHat2 [116] and Bowtie2 [117] to map RNA-Seq reads in the sixth data set to the *Homo sapiens* reference genomes (hg19) in the UCSC genome browser in conjunction with the RefSeq genome reference annotation (hg19). Tables 5.4-5.9 show the mapping statistics of six sets of RNA-Seq data. Overall, more than 70% of reads were mapped to the genome successfully. Particularly, a very high mapping rate was reached on the sixth data set. These mapping success rates were within the reasonable range, suggesting the good quality of the data and the correctness of the mapping process. This reads mapping process reduced the size of data by several orders of magnitude.

5.4.2 Gene expression values calculated from the reads mapping data

RNAMiner removed reads that mapped to multiple locations on a reference genome from the mapping data. The gene expression values were calculated by Cufflinks [118], MULTICOM-MAP [122, 123, 124], and HTSeq [119] on the remaining RNA-Seq reads mapped to unique locations on the genome. Compiling reads mappings into gene expression values generates an overall profile of the expression levels of most genes in a transcriptome, which can reduce the size of dataset by about one thousand-fold

Samples	# reads	# reads mapped to unique site	# reads mapped to multiple sites	% of reads mapped
Mutant, Control	22,053,527	14,120,075	2,577,992	75.72%
Wild-type, Control	29,483,443	19,560,525	3,077,978	76.78%
Mutant, CDDO	16,050,830	10,500,068	1,832,101	76.83%
Wild-type, CDDO	26,643,277	17,185,336	2,840,999	75.16%
Mutant, Sutherlandia	37,321,607	23,776,732	3,690,121	73.60%
Wild-type, Sutherlandia	27,678,509	18,150,349	2,717,683	75.39%
Mutant, Elderberry	25,750,508	17,155,488	2,631,750	76.84%
Wild-type, Elderberry	24,036,293	15,882,208	2,386,226	76.00%

Table 5.4: Mapping statistics of the first set of RNA-Seq data of *Mouse*.

(i.e., from gigabytes to megabytes) in our experiments. The compilation process transforms the raw data into meaningful expression profiles of genes. For example, three gene expression plots for comparisons between Control and each treatment in mutant mouse in the first data set are shown in Figure 5.6, and two gene expression plots for comparisons between 2A and 2B, between 2A and 3D in the second dataset are shown in Figure 5.7. In these figures, gene expression values calculated by MULTICOM-MAP were used, and the range of these values was constrained to [0, 100] while keeping the original ratios in order to make these figures readable. Usually, the points beyond the diagonal are candidates of differentially expressed genes.

MULTICOM-MAP and HTSeq were used to calculate the raw read counts in the third and fourth sets of data. The counts were normalized by dividing them by the total number of uniquely mapped reads in the replicate. The normalized

Samples	# reads	# reads mapped to unique site	# reads mapped to multiple sites	% of reads mapped
2A	12,390,167	9,016,108	1,513,122	84.98%
2B	11,760,788	8,220,731	1,445,292	82.19%
2C	9,481,395	6,892,027	1,178,753	85.12%
2D	19,450,682	13,849,985	2,406,684	83.58%
2E	11,743,452	8,381,763	1,418,645	83.45%
2F	12,104,053	8,692,100	1,510,391	84.29%
2G	13,301,646	9,427,606	1,642,257	83.22%
2H	15,766,959	11,158,652	1,950,974	83.15%
3A	22,688,673	16,126,579	2,773,025	83.30%
3B	20,352,253	14,506,676	2,503,008	83.58%
3C	20,301,445	14,486,410	2,401,849	83.19%
3D	14,985,494	10,729,926	1,876,610	84.12%

Table 5.5: Mapping statistics of the second set of RNA-Seq data of *Mouse*.

count of a gene was an indicator of the relative expression level of the gene in the replicate. The normalized counts of a gene in multiple replicates of a sample were further averaged and used as the measure of the relative expression level of the gene in the sample. Figure 5.8 shows one gene expression plot for the comparison between CF (Control Female) and CM (Control Male) in the third data set. In this figure, gene expression values were calculated by MULTICOM-MAP, and the values were transformed by $\log_2$ in order to make the figure readable. Two gene expression plots for the comparison between Col (Wild-Type) and hae-3 hsl2-3 (mutant) in the fifth data set are illustrated in Figure 5.9, and two gene expression plots for the comparison between 2pc3 and 1Sfesrrb in the sixth data set are illustrated in Figure 5.10. The left plot in each figure was generated from all the genes, and the right one was generated from differentially expressed genes. The gene expression values were calculated by MULTICOM-MAP and normalized by $\log_2$. According to the two plots in Figures 5.9 and 5.10, the distribution of expression values of differentially expressed genes is

Samples	# reads	# reads mapped to unique site	# reads mapped to multiple sites	% of reads mapped
CF1	51,144,998	45,977,130	900,427	91.66%
CF2	81,302,211	74,246,307	1,415,608	93.06%
CF3	123,512,038	108,573,161	1,797,884	89.36%
CM1	77,424,855	70,221,478	1,520,820	92.66%
CM2	61,946,818	55,327,486	1,103,757	91.10%
CM3	69,294,584	61,985,415	1,518,519	91.64%
HMF1	85,587,833	78,370,743	1,323,717	93.11%
HMF2	44,339,865	39,020,383	701,810	89.59%
HMF3	75,974,183	68,654,562	1,302,500	92.08%
HMM1	74,429,022	67,318,010	1,682,459	92.71%
HMM2	68,985,281	61,450,714	1,302,852	90.97%
HMM3	76,796,015	69,056,721	1,578,337	91.98%

Table 5.6: Mapping statistics of the third set of RNA-Seq data of *Drosophila melanogaster*.

quite different than that of the rest of the genes.

5.4.3 Differentially expressed genes identified from the RNA-Seq data

RNAMiner identified differentially expressed genes between control and each treatment using Cuffdiff [118], edgeR [120], and DESeq [121]. The threshold of p-value was set to 0.05 to select differentially expressed genes. For example, the number of differentially expressed genes for each comparison and their overlaps in both mutant mouse and wild-type mouse in the first data set are shown in Figure 5.11. The number of differentially expressed genes for each comparison in the second data set is shown in Figure 5.12. These differentially expressed genes were derived from the overlaps of three sets of differentially expressed genes separately identified by Cuffdiff, MULTICOM-MAP + edgeR, and MULTICOM-MAP + DESeq. As shown in Fig-

Samples	# reads	# reads mapped to unique site	# reads mapped to multiple sites	% of reads mapped
CF1	83,480,611	56,708,379	2,163,167	70.52%
CF2	56,660,705	42,714,627	1,398,576	77.86%
CF3	67,314,472	50,492,765	1,681,644	77.51%
CM1	50,000,247	38,470,206	1,206,401	79.35%
CM2	70,869,571	53,559,942	1,657,406	77.91%
CM3	68,530,284	51,799,947	1,627,155	77.96%
mF1	78,004,841	61,015,420	2,721,937	81.71%
mF2	51,629,214	40,273,082	1,573,248	81.05%
mF3	75,882,842	59,657,154	2,740,131	82.23%

Table 5.7: Mapping statistics of the fourth set of RNA-Seq data of *Drosophila melanogaster*.

ure 5.12, the number of differentially expressed genes increased with the increase of FruHis concentration in the absence or presence of 4 μM lycopene.

The number of differentially expressed genes for two comparisons between Col (Wild-Type) and hae-3 hsl2-3 (mutant), between Col_qtrim (Wild-Type) and hae-3 hsl2-3_qtrim (mutant), and their overlaps in the fifth data set are shown in Figure 5.13. These differentially expressed genes were derived from the overlaps of three sets of differentially expressed genes generated separately by Cuffdiff, MULTICOM-MAP + edgeR, MULTICOM-MAP + DESeq. We also identified differentially expressed genes for two comparisons: between 2pc3 and 1Sfesrrb, between 4ctrl and 3DY131 in the sixth data set using edgeR based on read counts calculated by MULTICOM-MAP. EdgeR identified 6,210 differentially expressed genes for the comparison between 2pc3 and 1Sfesrrb, and 590 differentially expressed genes for the comparison between 4ctrl and 3DY131. On the RNAMiner web service, users can select different p-value (or q-value) thresholds to select a specific number of differentially expressed genes according to their needs. In addition to generating the testable biological hypotheses (e.g.

Samples	# reads	# reads mapped to unique site	# reads mapped to multiple sites	% of reads mapped
Col_1	27,725,818	24,853,210	973,400	93.15%
Col_2	34,323,205	30,712,426	1,319,275	93.32%
Col_3	27,486,337	24,759,189	836,816	93.12%
Col_1_qtrim	17,555,221	15,965,329	636,099	94.57%
Col_2_qtrim	22,064,711	20,041,732	876,453	94.80%
Col_3_qtrim	17,459,673	15,956,994	548,720	94.54%
hae-3 hsl2-3_1	26,356,053	23,676,670	886,816	93.20%
hae-3 hsl2-3_2	20,998,406	18,793,308	727,901	92.97%
hae-3 hsl2-3_3	28,372,647	25,669,013	914,982	93.70%
hae-3 hsl2-3_1 _qtrim	16,641,162	15,168,566	578,226	94.63%
hae-3 hsl2-3_2 _qtrim	13,167,066	11,963,242	473,323	94.45%
hae-3 hsl2-3_3 _qtrim	17,927,078	16,467,776	597,035	95.19%

Table 5.8: Mapping statistics of the fifth set of RNA-Seq data of *Arabidopsis thaliana*.

gene targets for experimental testing), differential gene expression analysis generally reduces the size of data by about two folds, shifting point of interest from almost all the genes in a genome to a small portion of genes most relevant to the biological experiment.

5.4.4 Predicted functions of differentially expressed genes

RNAMiner predicted functions of differentially expressed genes using MULTICOM-PDCN. The predicted function terms were ranked by their significance of enrichment among the differentially expressed genes. For example, Figure 5.14 shows the top 10 most significantly enriched biological process functions for the comparison between Control and CDDO in both mutant mouse and wild-type mouse in the first data set. The two comparisons have 5 common biological processes in the top 10 biological pro-

Samples	# reads	# reads mapped to unique site	# reads mapped to multiple sites	% of reads mapped
1Sfesrrb-2	17,448,758	16,392,693	692,905	97.92%
1Sfesrrb-3	16,228,533	15,239,649	662,064	97.99%
2pc3-1	15,582,276	14,641,199	626,397	97.98%
2pc3-3	17,066,953	16,009,327	707,445	97.95%
3DY131-1	17,130,579	15,966,495	806,969	97.92%
3DY131-2	15,500,204	14,623,868	576,060	98.06%
4ctrl-1	19,117,412	17,885,236	858,147	98.04%
4ctrl-2	16,269,465	15,280,726	663,813	98.00%

Table 5.9: Mapping statistics of the sixth set of RNA-Seq data of *Human*.

cesses. The top 10 biological process functions for two comparisons between 2A and 2F (FruHis=16 without Lycopene), between 2A and 3D (FruHis=16 with Lycopene) in the second data set are shown in Figure 5.15. The two comparisons have 3 common biological processes in the top 10 biological processes. The top 10 biological process functions for the comparison between Col (Wild-Type) and hae-3 hsl2-3 (mutant) in the fifth data set are reported in Figure 5.16, and the two comparisons share 8 common biological processes in the top 10 biological processes. In these figures, the number besides each column is p-value of the enrichment of each predicted function.

Although the step of gene function analysis does not substantially reduce the size of data physically, it can logically summarize hundreds of differentially expressed genes into a small number (i.e., tens) of biological processes activated or deactivated in the biological experiment which sheds light into the potential biological mechanism relevant to the experiment.

5.4.5 Constructed gene regulatory networks

RNAMiner used MULTICOM-GNET [127] to construct gene regulatory networks based on differentially expressed genes and transcription factors. For example, a repression gene regulatory module with expression correlation 0.85 in mutant mouse in the first data set is illustrated in Figure 5.17. This module was comprised of 21 differentially expressed genes. Three transcription factors: Tgfb1i1, Htatip2, and Jun, were predicted to collaboratively regulate this group of genes. An activation gene regulatory module for the comparison between Col (Wild-Type) and hae-3 hsl2-3 (mutant) with expression correlation 0.85 in the fifth data set is shown in Figure 5.18. This module was comprised of 35 differentially expressed genes. Four transcription factors, AT3G59580, AT1G56650, AT1G28050, and AT1G52890, were predicted to collaboratively regulate this group of genes.

RNAMiner also used a R package "igraph" [145] to visualize gene regulatory networks by linking the regulatory relationships between and within all the gene regulatory modules predicted by MULTICOM-GNET together. Figure 5.19 shows a gene regulatory network representing the regulatory relationships of top 10 gene regulatory modules ranked by expression correlation scores on the first data set. There are 14 transcription factors (red nodes), 338 genes (blue nodes), and 1,280 edges (regulatory relationships) in the network.

The step of gene regulatory network reconstruction condenses hundreds of differentially expressed genes and their expression data into dozens of valuable gene regulatory modules, which may reveal the underlying biological mechanism controlling the expression in the biological experiment. The network modules not only provide the human comprehensible interpretation of the gene expression levels, but also the important transcription factors and their target genes that are very valuable for generating hypotheses for new biological experiments.

5.5 Use of the RNAMiner web service

The RNAMiner web service (Figure 5.20) is available at http://calla.rnet.missouri.edu /rnaminer/index.html. Users can submit requests on the home page and receive an email with a link to the data analysis results.

5.5.1 Submit a request

Here are steps of submitting a request:

1) Prepare RNA-Seq reads files (.fastq). The acceptable formats by the RNAMiner web service include ".fastq.gz" and ".fastq.tar.gz".

2) Choose the analysis categories. Each category needs the results in the previous categories. If one category is chosen, the previous categories will be executed automatically. For example, if "predicting gene functions" is chosen, the first three categories will be executed automatically.

3) Choose the species. RNAMiner can analyze RNA-Seq data on four species: *Human*, *Mouse*, *Drosophila melanogaster*, and *Arabidopsis thaliana*.

4) Choose criterion of identifying differentially expressed genes. It is p-value or q-value.

5) Set threshold of p-value or q-value for identifying differentially expressed genes. The value should be between 0 and 1. The default value is 0.05.

6) Input email address. An email with a link to the data analysis results will be sent to this email address when the data analysis is finished.

7) Input sample names.

8) Upload reads files. The last three categories request users to upload reads files for both two samples. Users can upload more than one reads files for each sample.

9) Click "Submit".

After a request is submitted successfully, one web page (Figure 5.21) will be shown saying the data is in process. If one user submitted one request to the RNAMiner web service and it is running or it is in the waiting queue, he/she cannot submit another request.

5.5.2 Receive the results

When the data analysis is finished, users will receive an email with a link to one web page (Figure 5.22) with the data analysis information and a result link. The result page (Figure 5.23) will be shown by clicking the result link. Users can view and download the analysis data on the result page.

The time expense of analyzing a set of RNA-Seq data by RNAMiner depends on how big the data is, how many reads files there are in the data set, and how many jobs there are in the waiting queue. Normally a data analysis can be finished by RNAMiner in several hours. However, the time expense will be longer if there are a lot of jobs in the waiting queue. Our server cannot handle too many jobs at the same time because of CPU and space limitations.

5.6 Conclusions

The RNAMiner protocol and pipeline can progressively reduce the size of large datasets to produce valuable and comprehensible biological knowledge of manageable size, ranging from gene expression values, differentially expressed genes, gene function predictions, and gene regulatory networks. The test results on six RNA-Seq datasets of four different species help demonstrate its utility and versatility.

In order to further improve the quality of RNA-Seq data analysis, additional tools can be plugged into the RNAMiner protocol. In the future, we will add a high-speed RNA mapping tool - Gsnap [131] and a high-accuracy RNA mapping tool - Stampy [147] into the pipeline to map RNA reads to reference genomes. For identifying differentially expressed genes, we will include baySeq [148], ShrinkSeq [149], and NOISeq [150] into the pipeline in order to handle various sources of noise in RNA-Seq data even better. Furthermore, we will include an in-house tool of constructing biological networks from a group of co-expressed genes to reconstruct highly valuable metabolic networks and signal transduction networks for gene clusters identified by the RNAMiner protocol. Moreover, we will add the capability of analyzing the function of non-coding small RNAs into RNAMiner and use the information during the reconstruction of biological networks. The new improvements will be incorporated into the RNAMiner web service for the community to use.

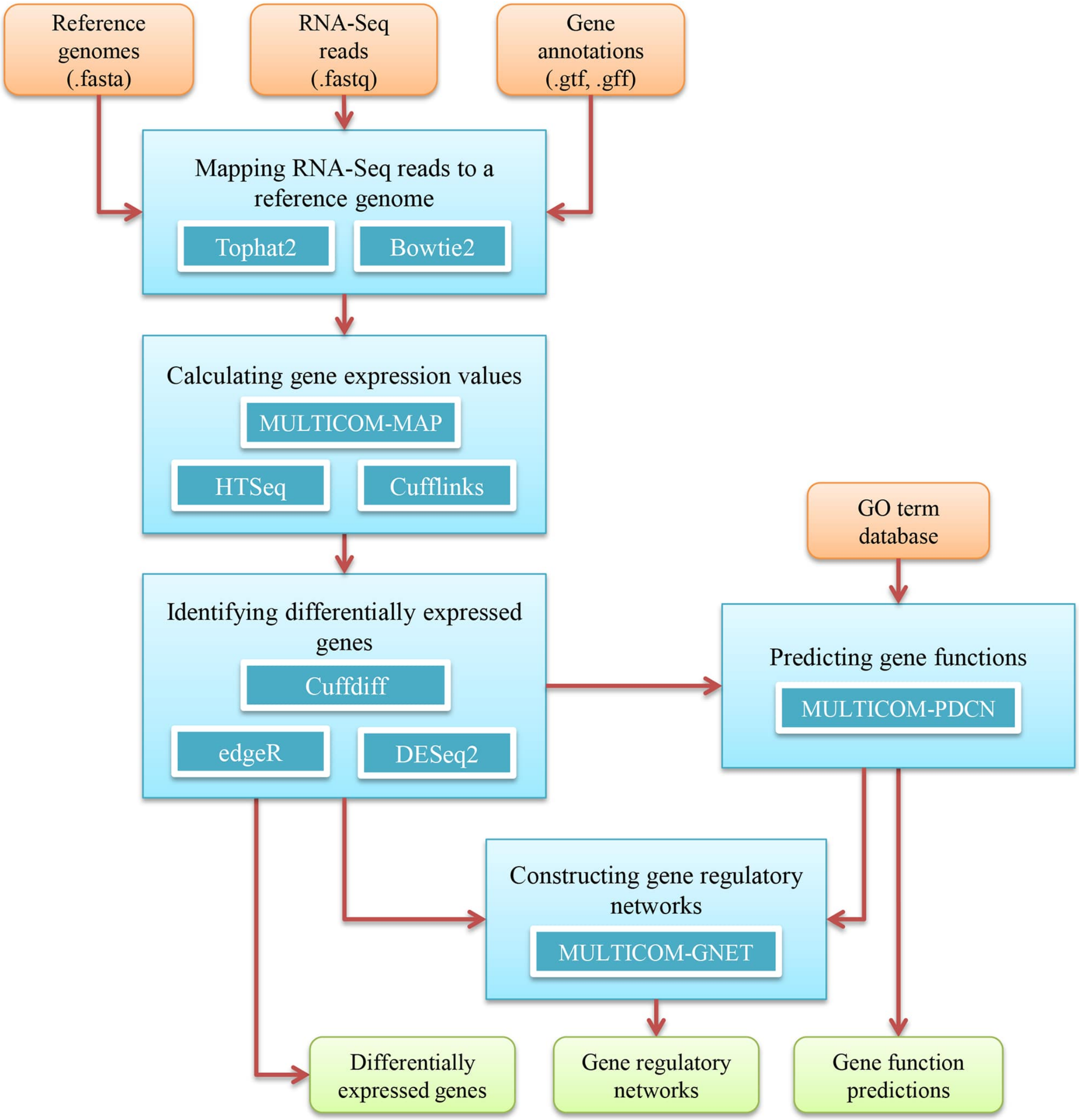

Reference genomes (.fasta)
RNA-Seq reads (.fastq)
Gene annotations (.gtf, .gff)
Mapping RNA-Seq reads to a reference genome
Tophat2
Bowtie2
Calculating gene expression values
MULTICOM-MAP
HTSeq
Cufflinks
GO term database
Identifying differentially expressed genes
Cuffdiff
edgeR
DESeq2
Predicting gene functions
MULTICOM-PDCN
Constructing gene regulatory networks
MULTICOM-GNET
Differentially expressed genes
Gene regulatory networks
Gene function predictions

Figure 5.1: The RNAMiner protocol for big transcriptomics data analysis. Five blue boxes denote five data analysis steps, i.e. mapping RNA-Seq reads to a reference genome, calculating gene expression values, identifying differentially expressed genes, predicting gene functions, and constructing gene regulatory networks. The tools used in each step are listed inside each box. The external input information is represented by brown boxes and the final output information is represented by green boxes. The information flow between these components is denoted by arrows.

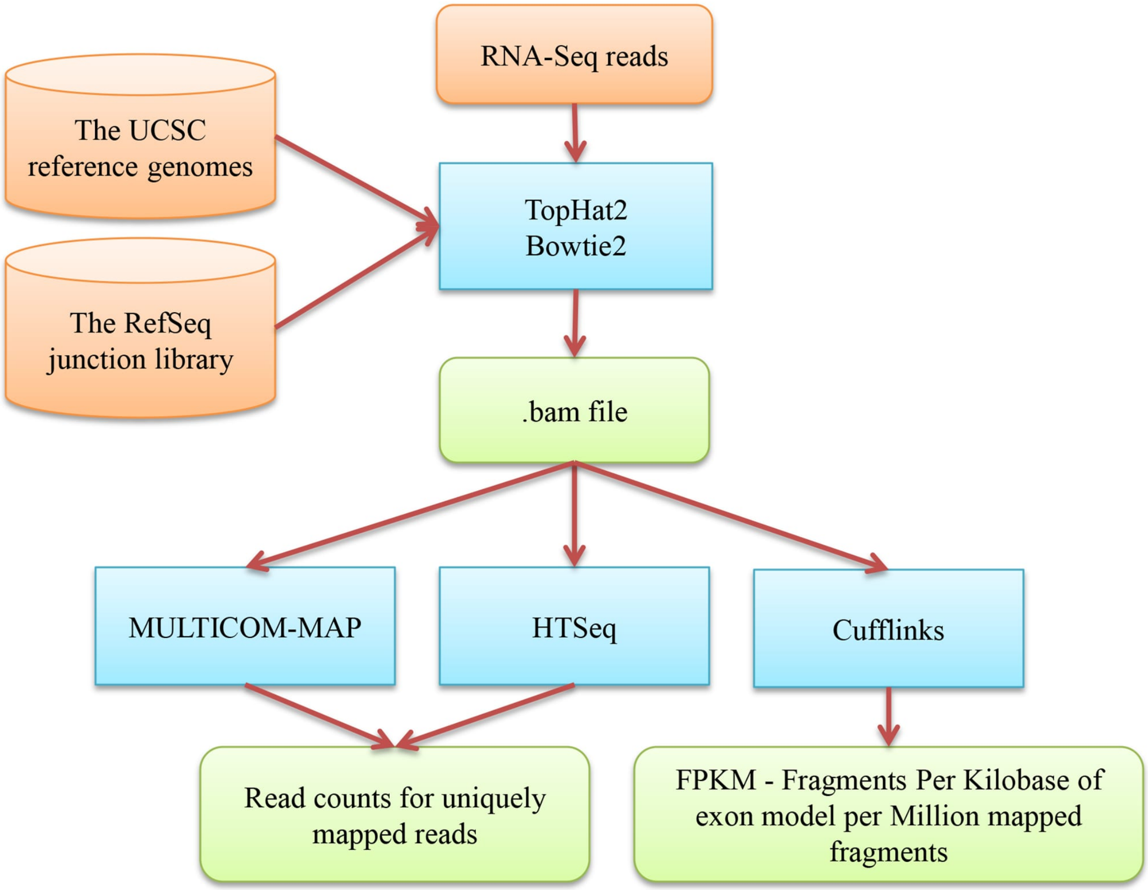

RNA-Seq reads
The UCSC reference genomes
The RefSeq junction library
TopHat2 Bowtie2
.bam file
MULTICOM-MAP
HTSeq
Cufflinks
Read counts for uniquely mapped reads
FPKM - Fragments Per Kilobase of exon model per Million mapped fragments

Figure 5.2: The workflow of mapping RNA-Seq reads to a reference genome and calculating gene expression values. The blue boxes denote the tools (TopHat2, Bowtie2, MULTICOM-MAP, HTSeq, Cufflinks) used in the steps of mapping RNA-Seq reads to a reference genome and calculating gene expression values. The external input information is represented by brown boxes and the output information is represented by green boxes. The information flow between these components is denoted by arrows.

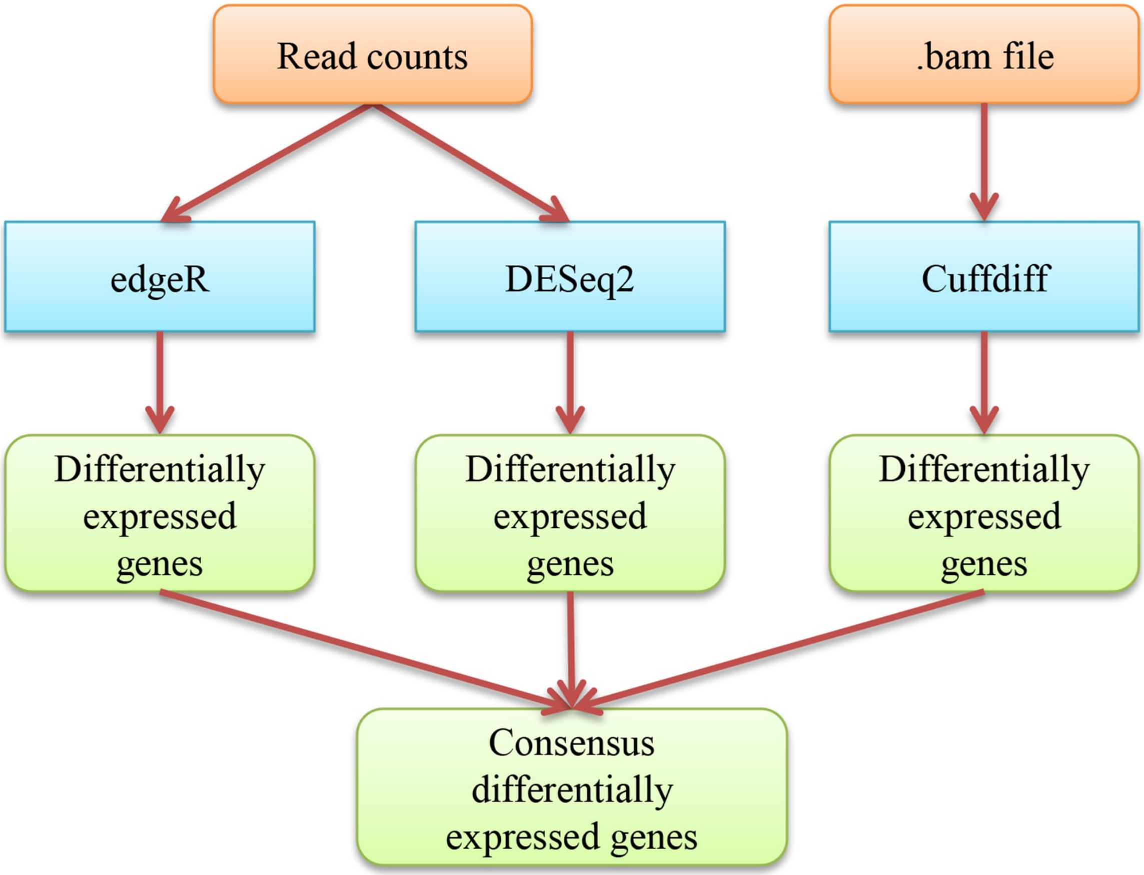

Read counts
.bam file
edgeR
DESeq2
Cuffdiff
Differentially expressed genes
Differentially expressed genes
Differentially expressed genes
Consensus differentially expressed genes

Figure 5.3: The workflow of identifying differentially expressed genes. The blue boxes denote the tools (edgeR, DESeq2, Cuffdiff) used in the step of identifying differentially expressed genes. The external input information is represented by brown boxes and the output information is represented by green boxes. The information flow between these components is denoted by arrows.

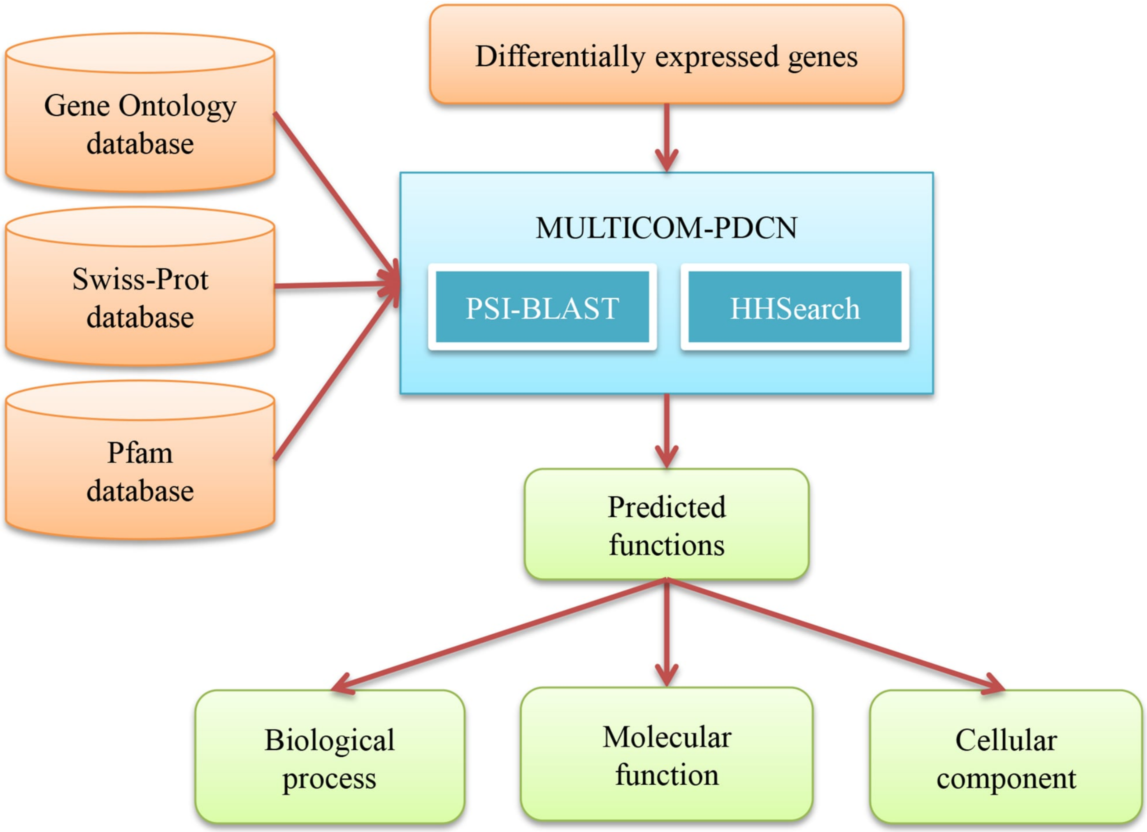


Figure 5.4: The workflow of predicting gene functions. The blue box denotes the tool used in the step of predicting gene functions. The tools of PSI-BLAST and HHSearch used in MULTICOM-PDCN are listed inside the blue box. The external input information is represented by brown boxes and the output information is represented by green boxes. The information flow between these components is denoted by arrows.

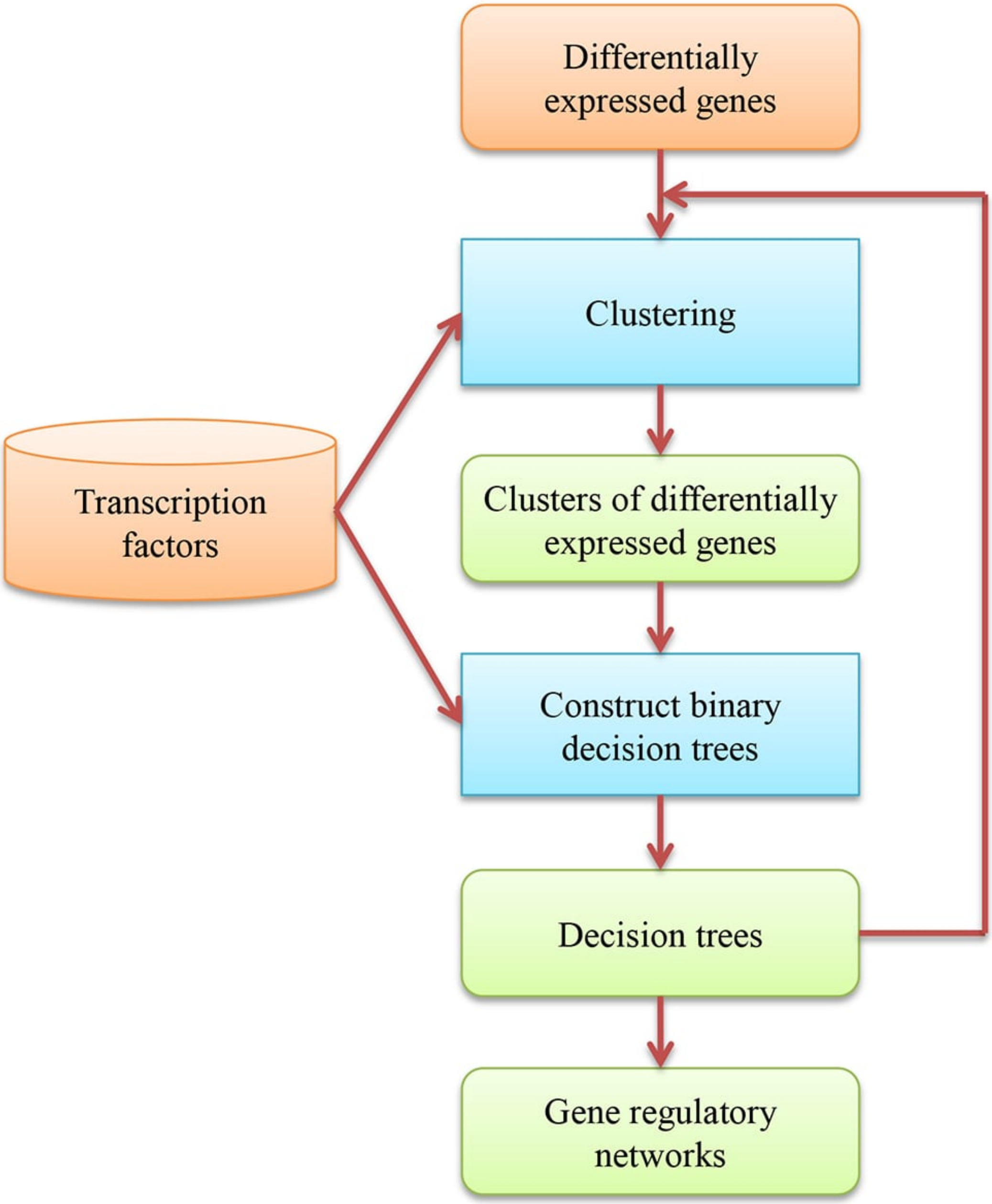

Differentially expressed genes
Clustering
Transcription factors
Clusters of differentially expressed genes
Construct binary decision trees
Decision trees
Gene regulatory networks

Figure 5.5: The workflow of constructing gene regulatory networks. The blue boxes denote the methods used by MULTICOM-GNET in constructing gene regulatory networks. The external input information is represented by brown boxes and the output information is represented by green boxes. The information flow between these components is denoted by arrows.

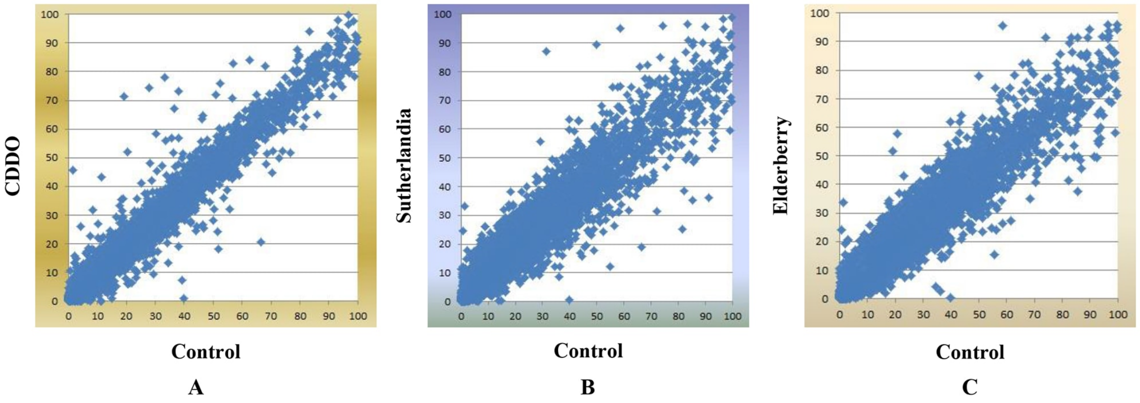


Figure 5.6: Three gene expression plots in the first data set. These plots are for comparisons between Control and each treatment in mutant mouse. The x-axis represents Control and the y-axis represents CDDO treatment in A, Sutherlandia treatment in B, and Elderberry treatment in C. We used gene expression values calculated by MULTICOM-MAP to make the plots. The range of these values was constrained to [0, 100] while keeping the original ratios in order to make these figures readable.

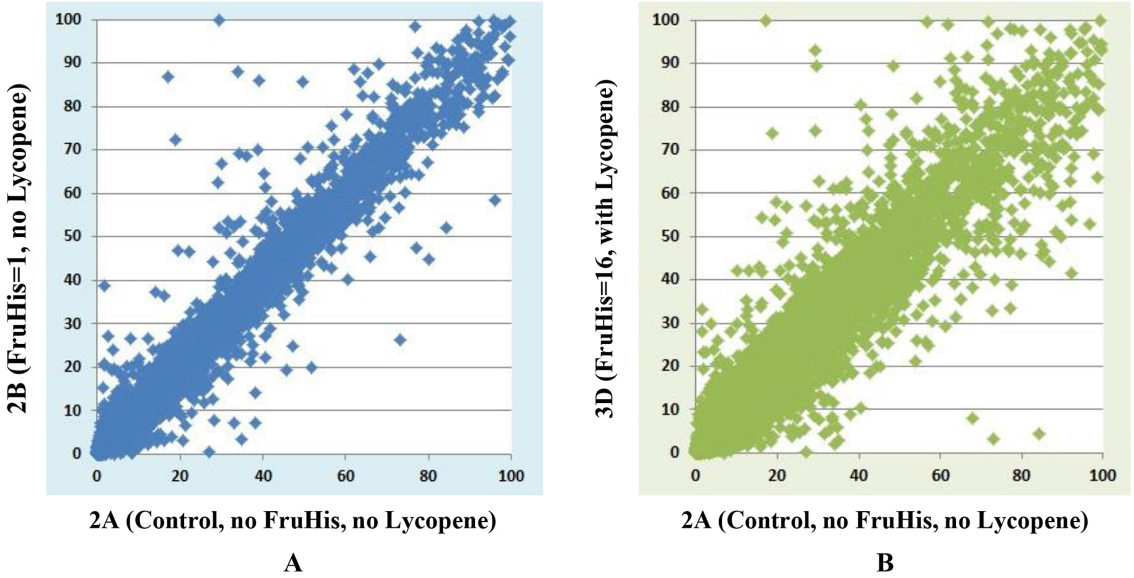


Figure 5.7: Two gene expression plots in the second data set. These plots are for comparisons between 2A and 2B, between 2A and 3D. The x-axis represents 2A (Control, no FruHis, no Lycopene) and the y-axis represents 2B (FruHis=1, no Lycopene) in A and 3D (FruHis=16, with Lycopene) in B. We used gene expression values calculated by MULTICOM-MAP to make the plots. The range of these values was constrained to [0, 100] while keeping the original ratios in order to make these figures readable.

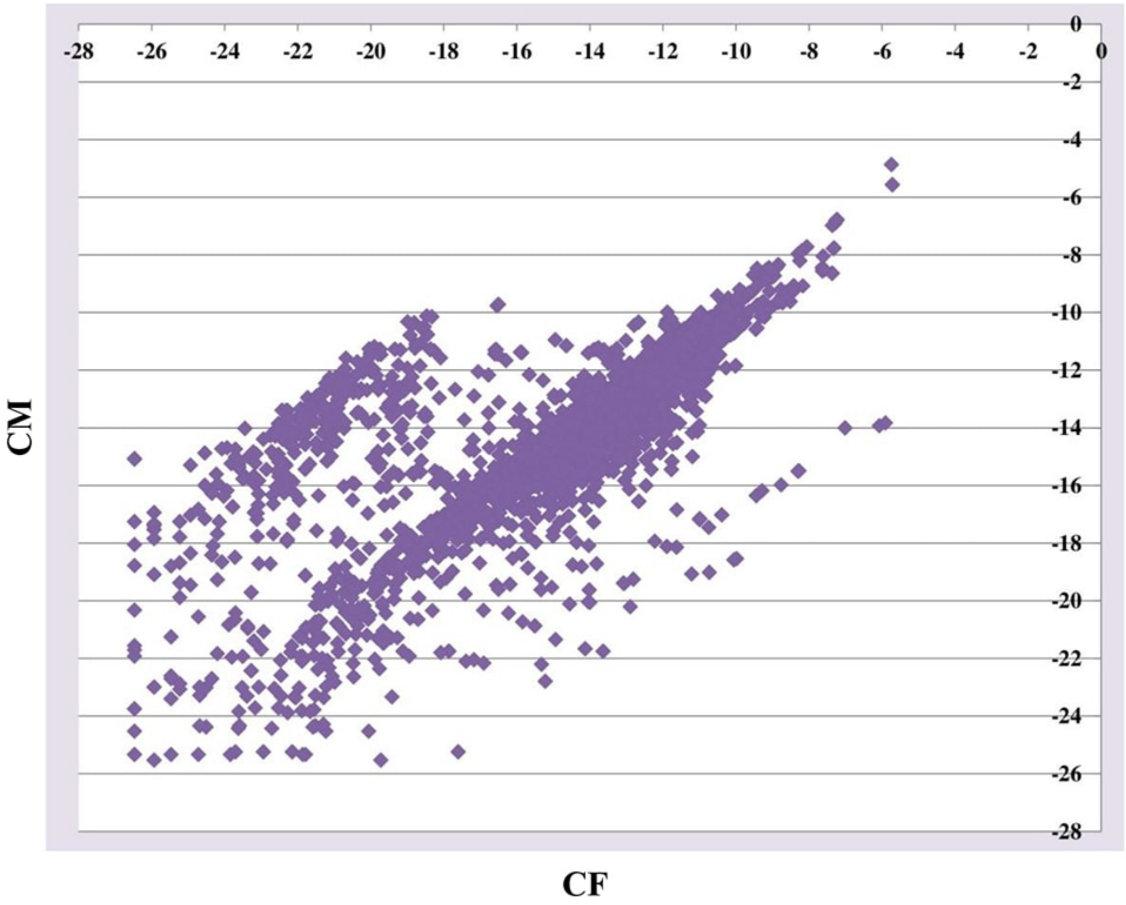
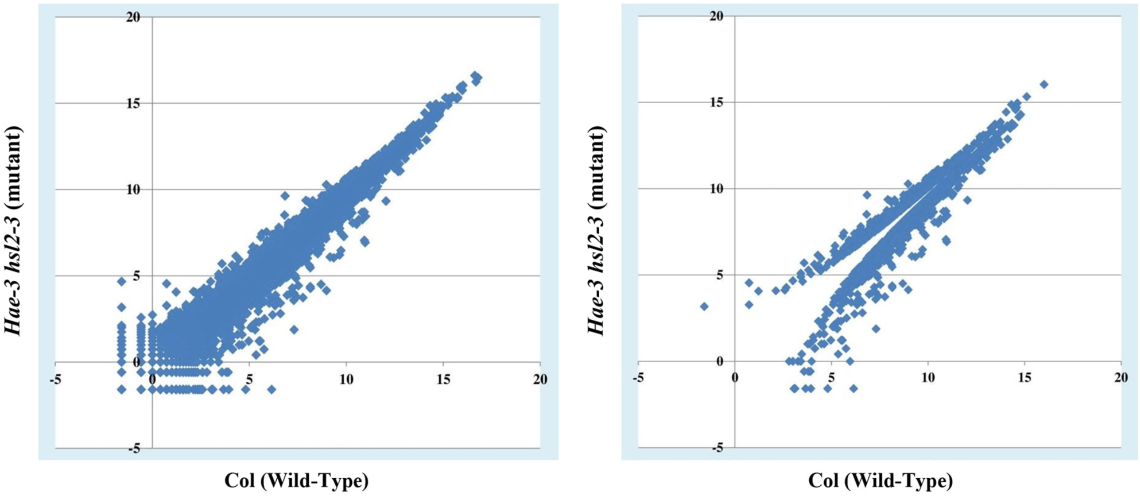
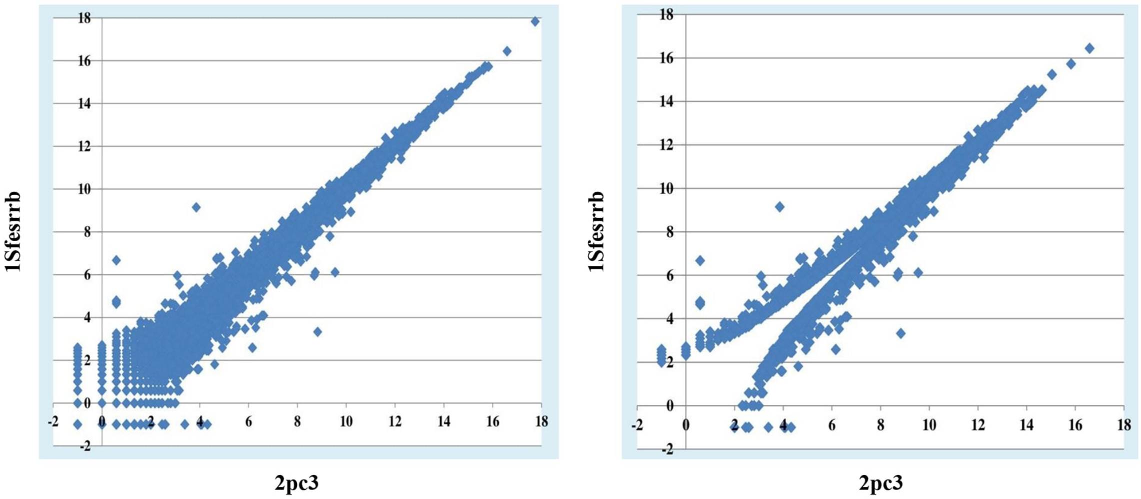


Figure 5.8: One gene expression plot in the third data set. The plot is for the comparison between CF and CM. The x-axis represents CF and the y-axis represents CM. We used gene expression values calculated by MULTICOM-MAP to make the plot. The raw counts were transformed by $\log_2$ in order to make the figure readable.



A (All the genes) **B (Differentially expressed genes)**

Figure 5.9: Two gene expression plots in the fifth data set. These plots are for the comparison between Col and hae-3 hsl2-3. The x-axis represents Col (Wild-Type) and the y-axis represents hae-3 hsl2-3 (mutant). The left plot visualizes the expression values of all the genes, and the right one displays the expression values of differentially expressed genes. The gene expression values were calculated by MULTICOM-MAP. The raw counts were transformed by $\log_2$.



A (All the genes)

B (Differentially expressed genes)

Figure 5.10: Two gene expression plots in the sixth data set. These plots are for the comparison between 2pc3 and 1Sfesrrb. The x-axis represents 2pc3 and the y-axis represents 1Sfesrrb. The left plot visualizes the expression values of all the genes, and the right one displays the expression values of differentially expressed genes. The gene expression values were calculated by MULTICOM-MAP. The raw counts were transformed by $\log_2$.

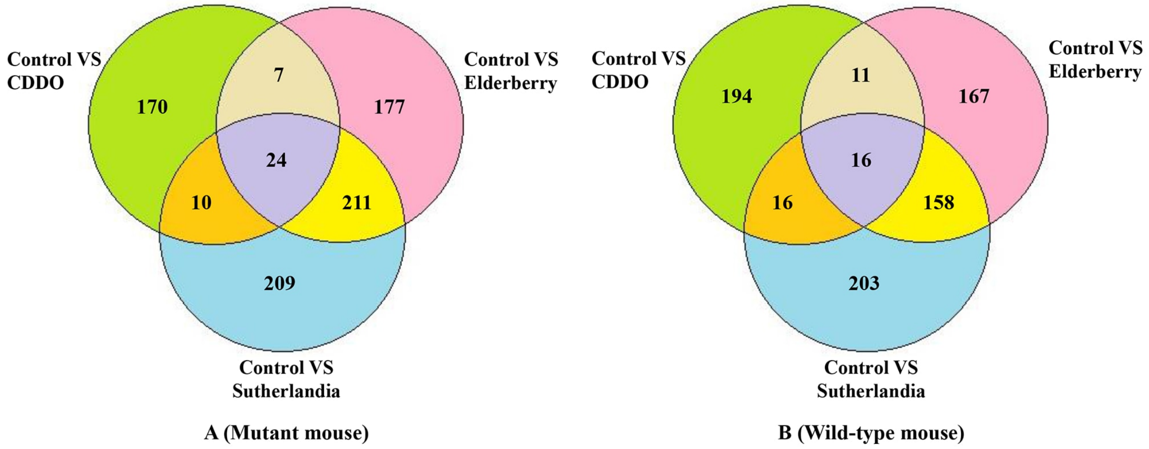


Figure 5.11: The number of differentially expressed genes in the first data set. These numbers were calculated for different pairs of comparisons between Control and each treatment, and their overlaps in both mutant mouse and wild-type mouse cells. The differentially expressed genes in each comparison were derived from the overlaps of three sets of differentially expressed genes generated by Cuffdiff, MULTICOM-MAP + edgeR, and MULTICOM-MAP + DESeq.

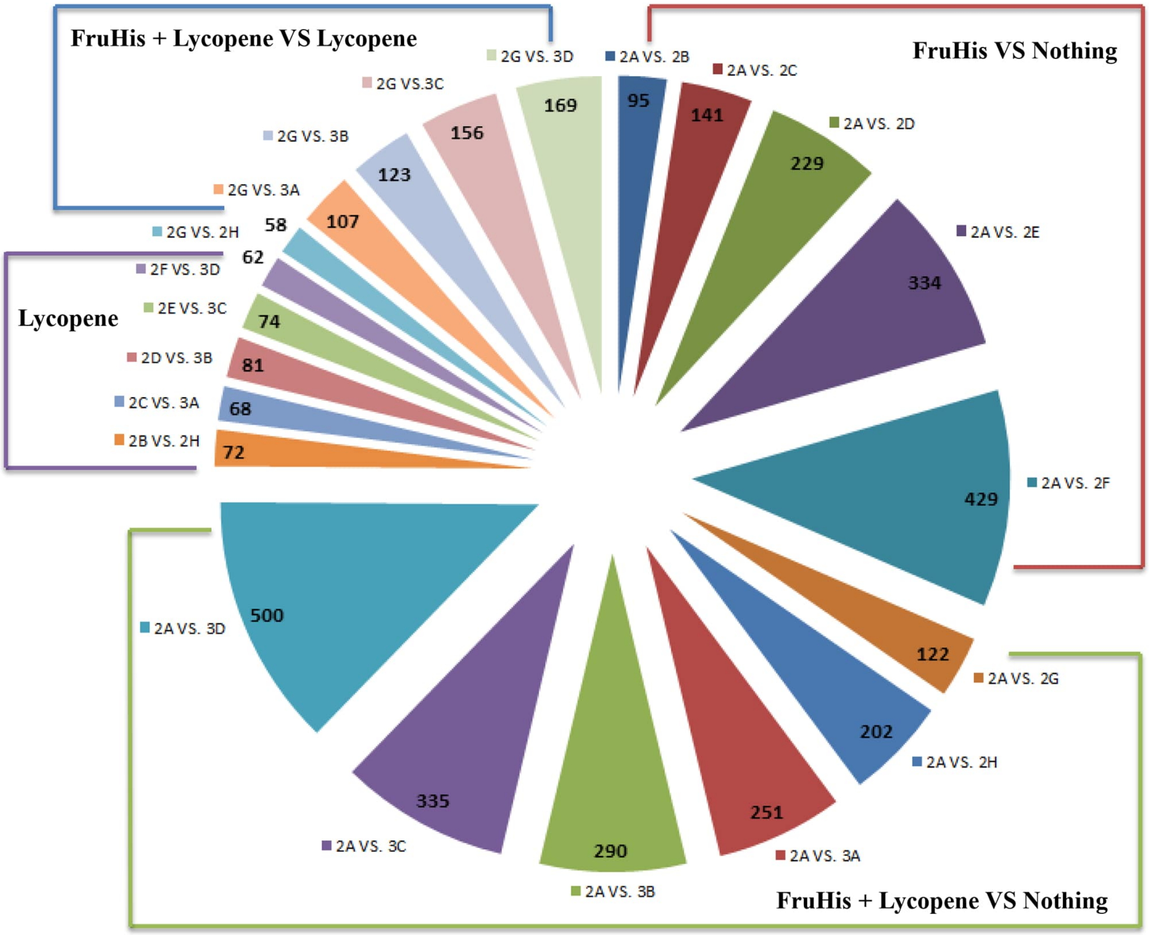


Figure 5.12: The number of differentially expressed genes for each pairwise comparison in the second data set. The differentially expressed genes in each comparison were derived from the overlaps of three sets of differentially expressed genes generated by Cuffdiff, MULTICOM-MAP + edgeR, and MULTICOM-MAP + DESeq.

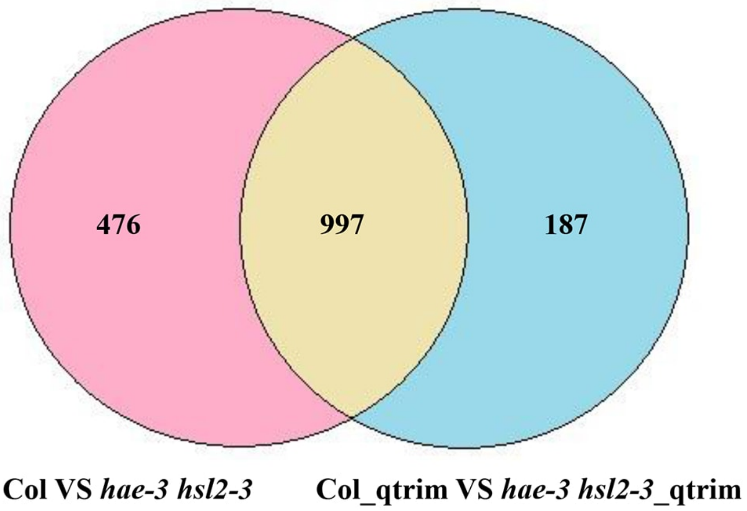


Figure 5.13: The number of differentially expressed genes in the fifth data set. These numbers were calculated for two comparisons between Col and hae-3 hsl2-3, between Col_qtrim and hae-3 hsl2-3_qtrim, and their overlap. The differentially expressed genes in each comparison were derived from the overlaps of three sets of differentially expressed genes generated by Cuffdiff, MULTICOM-MAP + edgeR, MULTICOM-MAP + DESeq.

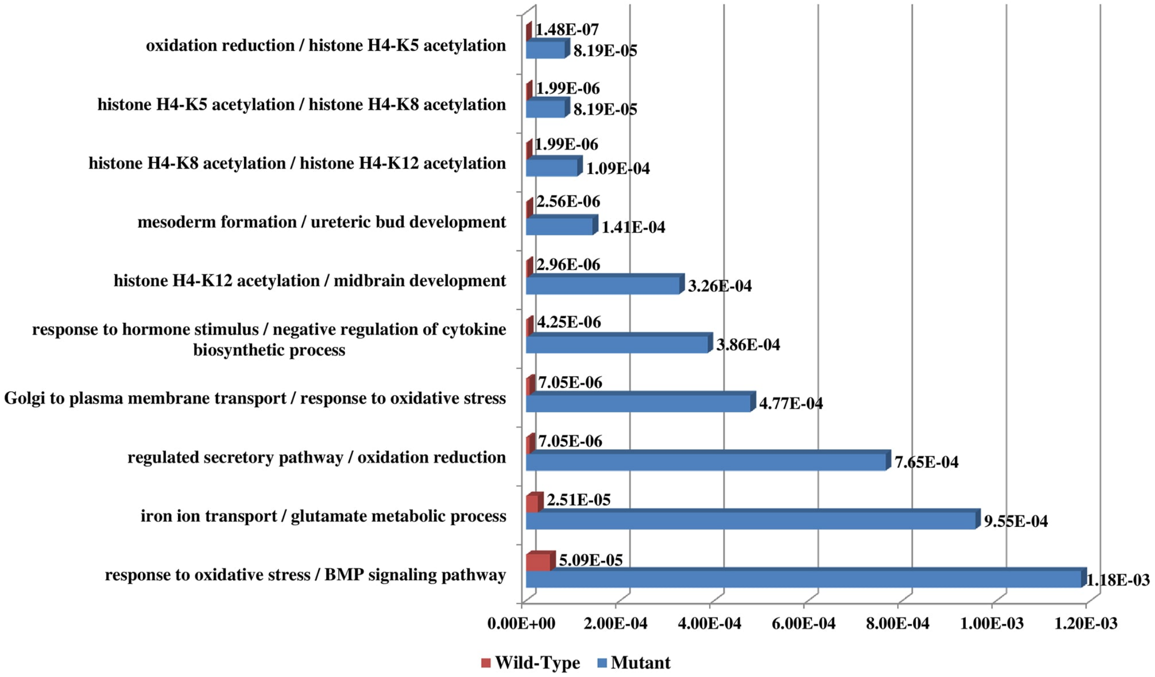


Figure 5.14: Top 10 biological process functions in the first data set. These functions were predicted for the comparison between Control and CDDO in both mutant mouse and wild-type mouse cells. Red bars denote the p-values of the top 10 predicted functions for wild-type mouse and blue bars denote the p-values of the top 10 predicted functions for mutant mouse. The number besides each bar is the significance of enrichment (p-value) of the predicted function. The p-value was calculated by Fisher's exact test. The function names of wild-type mouse and mutant mouse are listed on the left separated by "/". The two comparisons have five common biological processes among the top 10 biological processes.

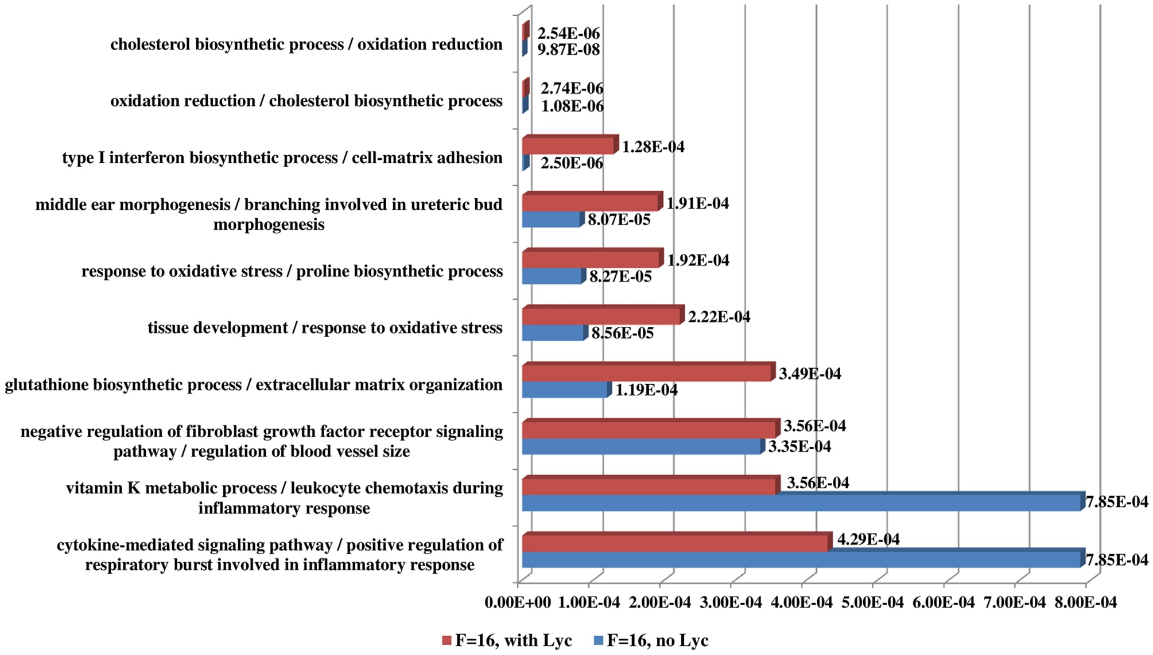


Figure 5.15: Top 10 biological process functions in the second data set. These functions were predicted for two comparisons between 2A (Control, no FruHis, no Lycopene) and 2F (FruHis=16 without Lycopene), between 2A and 3D (FruHis=16 with Lycopene). Red bars denote the p-values of the top 10 predicted functions for the comparison between 2A and 3D and blue bars denote the p-values of the top 10 predicted functions for the comparison between 2A and 2F. The number besides each bar is the significance of enrichment (p-value) of the predicted function. The p-value was calculated by Fisher's exact test. The function names of the two comparisons between 2A and 3D, between 2A and 2F are listed on the left separated by "/". The two comparisons have three common biological processes among the top 10 biological processes.

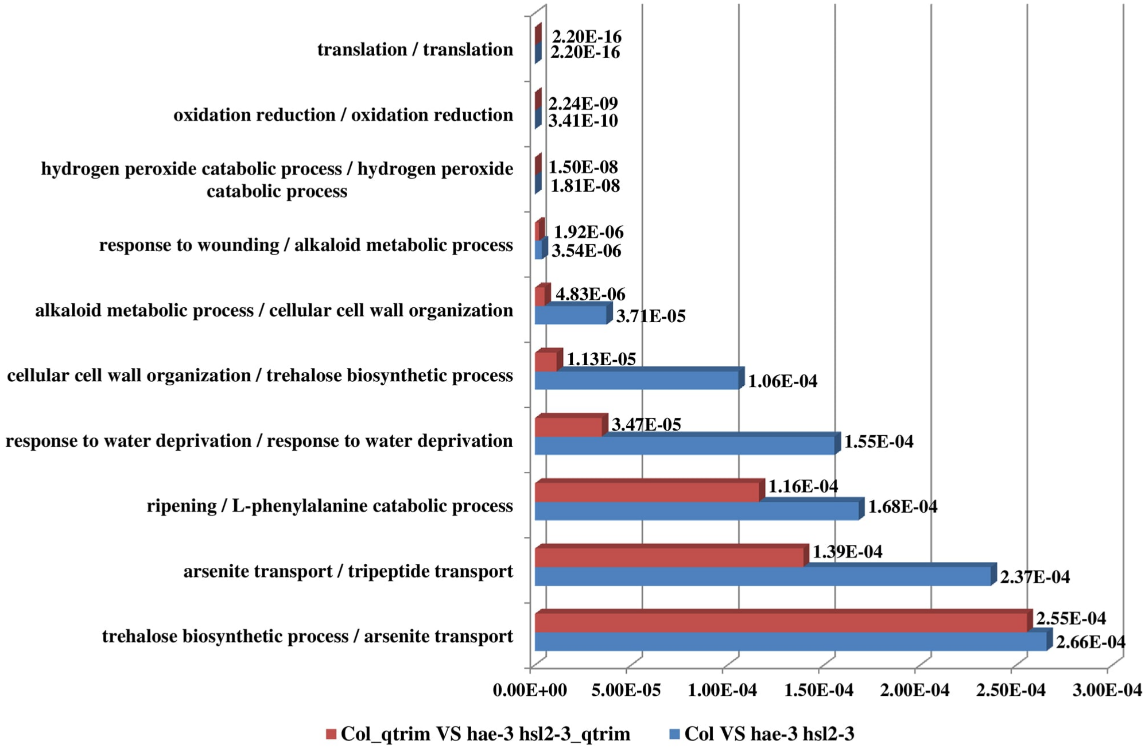


Figure 5.16: Top 10 biological process functions in the fifth data set. These functions were predicted for two comparisons between Col and hae-3 hsl2-3, between Col_qtrim and hae-3 hsl2-3_qtrim. Red bars denote the p-values of the top 10 predicted functions for the comparison between Col_qtrim and hae-3 hsl2-3_qtrim and blue bars denote the p-values of the top 10 predicted functions for the comparison between Col and hae-3 hsl2-3. The number besides each bar is the significance of enrichment (p-value) of the predicted function. The p-value was calculated by Fisher's exact test. The function names of the two comparisons between Col_qtrim and hae-3 hsl2-3_qtrim, between Col and hae-3 hsl2-3 are listed on the left separated by "/". The two comparisons have 8 common biological processes among the top 10 biological processes.

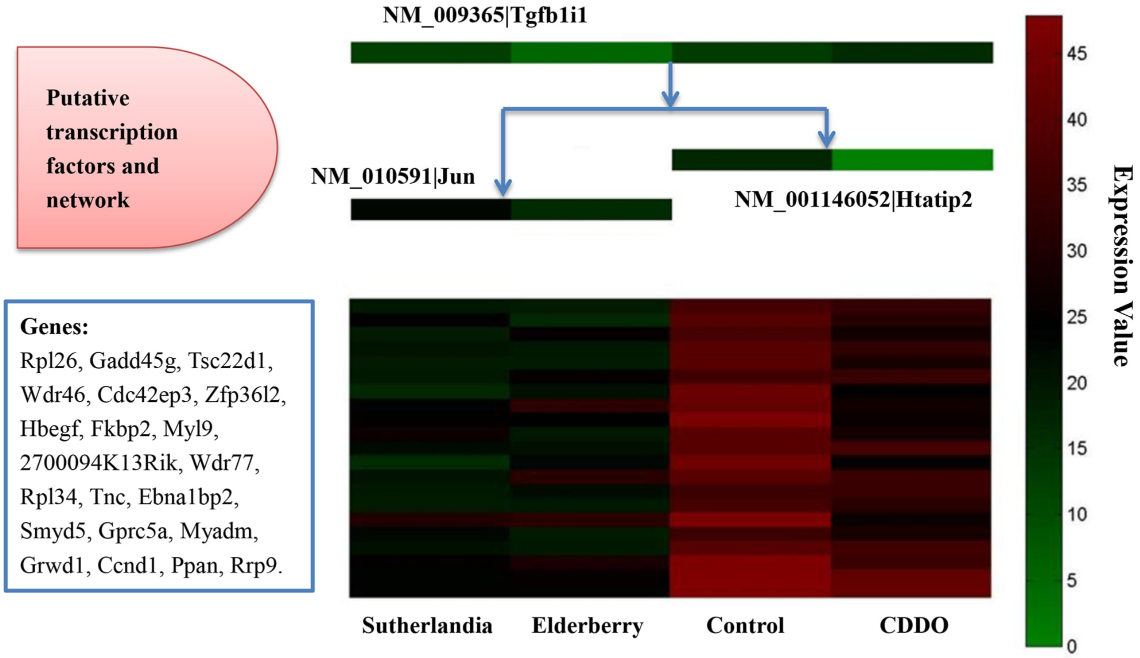


Figure 5.17: One repression gene regulatory module in mutant mouse cells in the first data set. The expression correlation score of the module was 0.85. The decision tree on the middle top illustrates how three putative transcription factors (Tgfb1i1, Htatip2, Jun) may collaboratively regulate the cluster of co-expressed genes in the middle bottom, where each row denotes a gene listed in the bottom left box and each column denotes one of four biological conditions (i.e. Control, CDDO, Sutherlandia, and Elderberry). The levels of gene expression values were represented by different colors ranging from lowest (green) to highest (red). The expression of the genes in the cluster under each condition is predicted to be regulated according to the expression levels of transcription factors listed on top of the condition. For example, under Sutherlandia treatment, the relatively low expression of Tgfbli1 and the medium expression of Jun caused the repression of the group of genes.

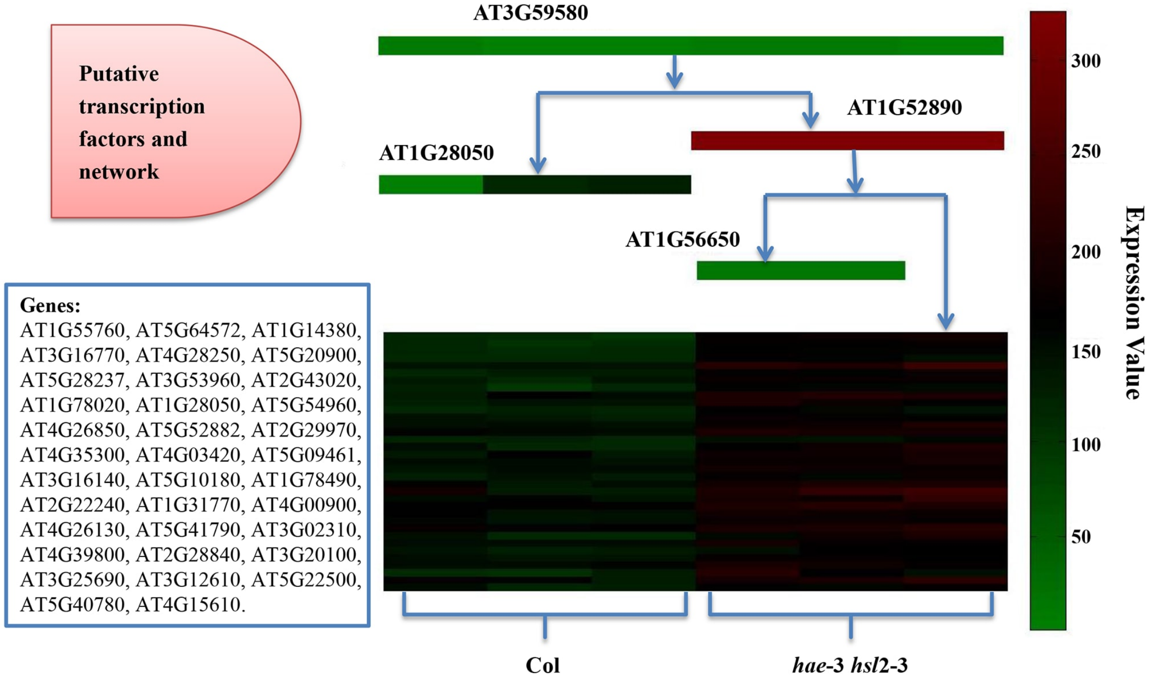


Figure 5.18: One activation gene regulatory module in the fifth data set. The gene regulatory module was constructed for the comparison between Col (Wild-Type) and hae-3 hsl2-3 (mutant), and the expression correlation score of the module was 0.85. The decision tree on the middle top illustrates how four putative transcription factors (AT3G59580, AT1G28050, AT1G52890, AT1G56650) may collaboratively regulate the cluster of co-expressed genes in the middle bottom, where each row denotes a gene listed in the bottom left box and each column denotes one of six biological replicates of two samples (i.e. Col and hae-3 hsl2-3). The levels of gene expression values were represented by different colors ranging from lowest (green) to highest (red). The expression of the genes in the cluster under each sample is predicted to be regulated according to the expression levels of transcription factors listed on top of the condition. For example, under the first replicate of Col, the low expression of AT3G59580 and the low expression of AT1G28050 caused the repression of the group of genes.

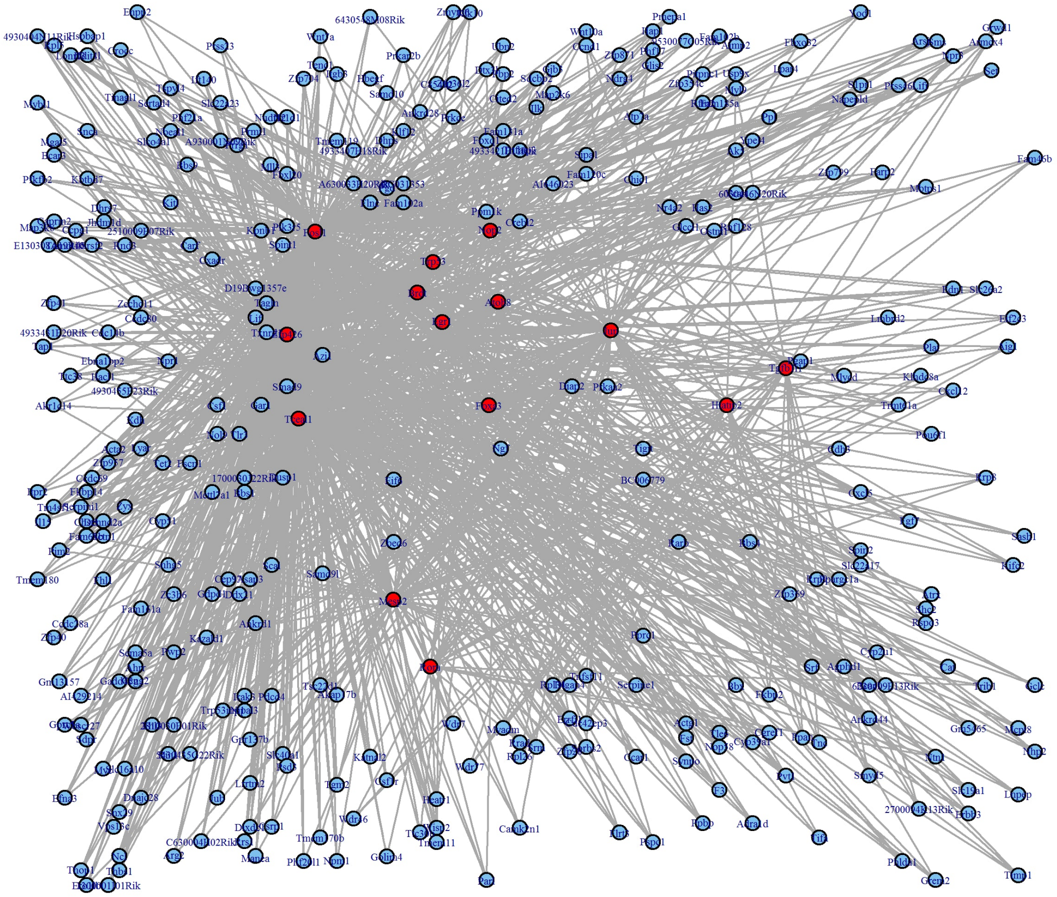
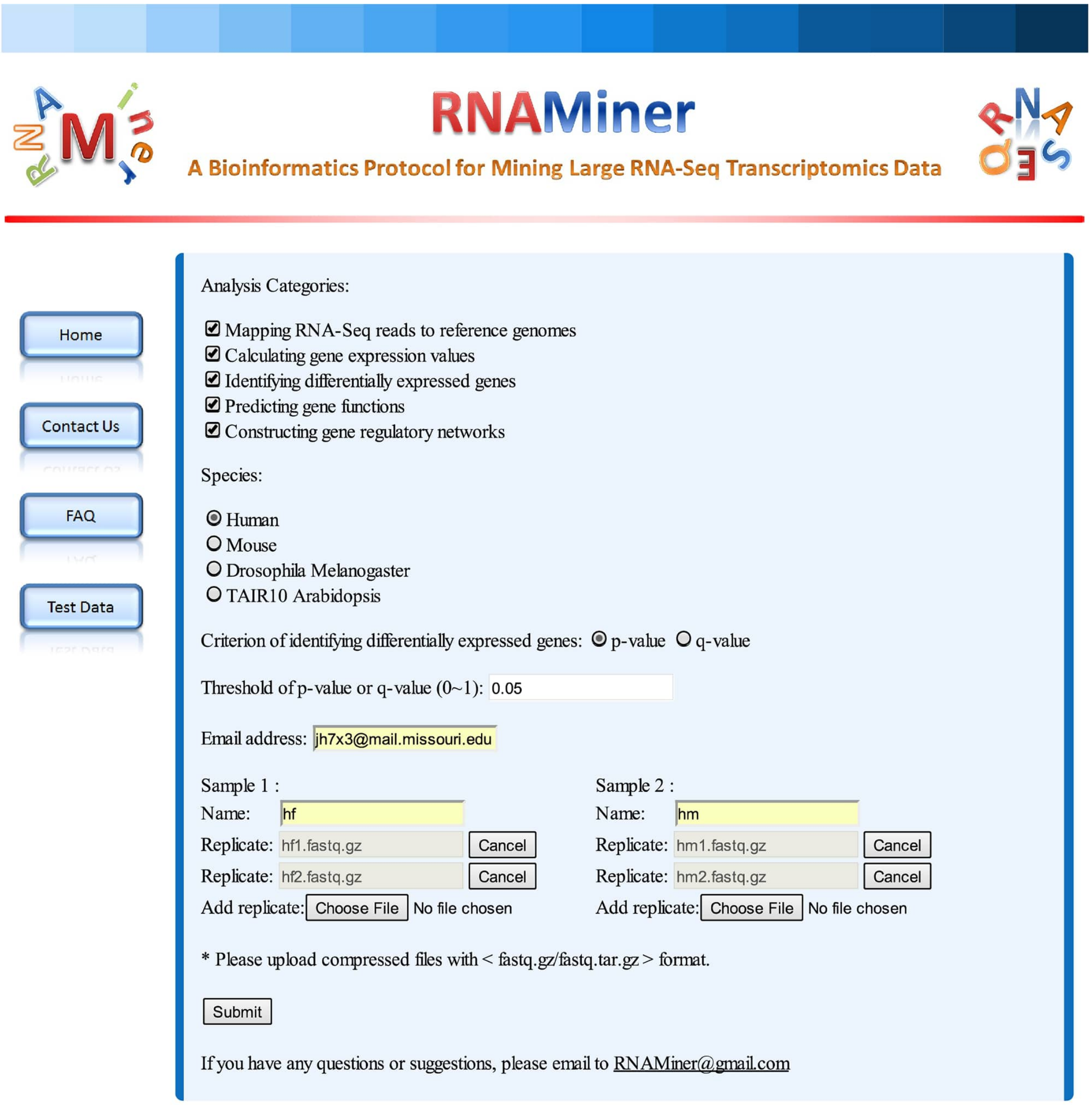


Figure 5.19: A visualized global gene regulatory network on the first dataset. The network includes all the gene regulatory relationships between and within top 10 gene regulatory modules ranked by expression correlation scores on the first dataset. Blue nodes represent target genes, and red nodes represent transcription factors which regulate the target genes. Each edge represents a regulatory relationship between a transcription factor and a gene.



RNAMiner
A Bioinformatics Protocol for Mining Large RNA-Seq Transcriptomics Data
Home
Contact Us
FAQ
Test Data

Analysis Categories:

☑ Mapping RNA-Seq reads to reference genomes
☑ Calculating gene expression values
☑ Identifying differentially expressed genes
☑ Predicting gene functions
☑ Constructing gene regulatory networks

Species:

● Human
○ Mouse
○ Drosophila Melanogaster
○ TAIR10 Arabidopsis

Criterion of identifying differentially expressed genes: ● p-value ○ q-value

Threshold of p-value or q-value (0~1): 0.05

Email address: jh7x3@mail.missouri.edu

Sample 1 :
Name: hf
Replicate: hf1.fastq.gz Cancel
Replicate: hf2.fastq.gz Cancel
Add replicate: Choose File No file chosen

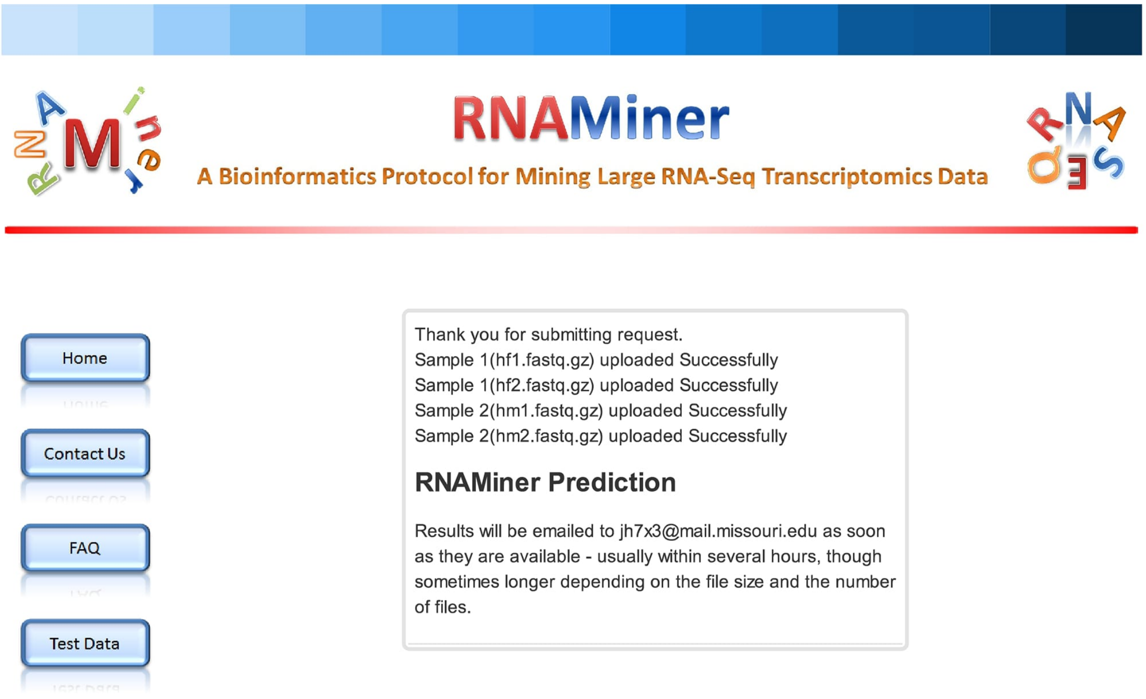
Sample 2 :
Name: hm
Replicate: hm1.fastq.gz Cancel
Replicate: hm2.fastq.gz Cancel
Add replicate: Choose File No file chosen

* Please upload compressed files with < fastq.gz/fastq.tar.gz > format.

Submit

If you have any questions or suggestions, please email to RNAMiner@gmail.com

Figure 5.20: The home page of the RNAMiner web service. Users can submit requests on the home page of the RNAMiner web service, and also can learn how to use RNAMiner, find contact information, and download the test data by clicking the navigation buttons on left.

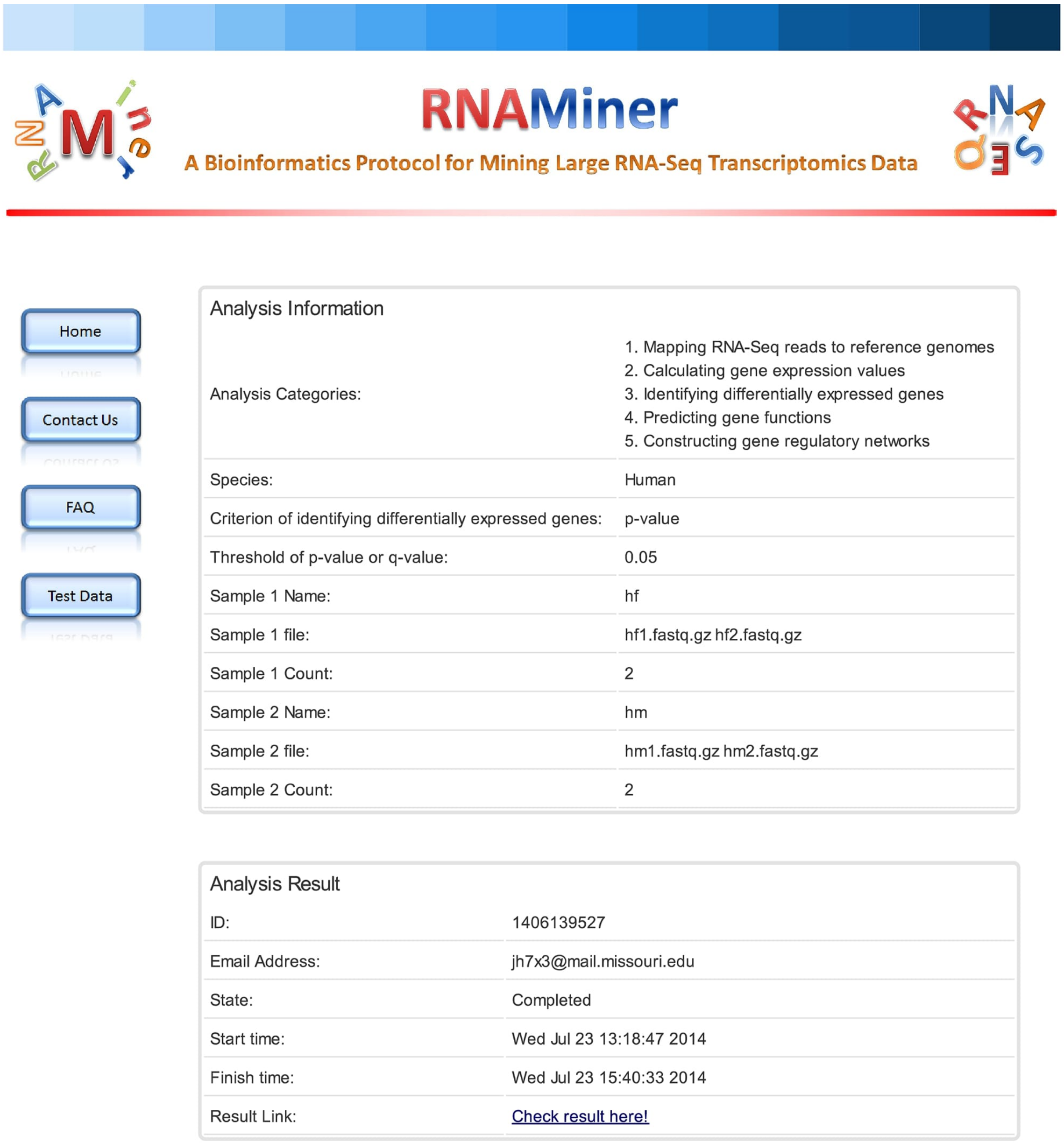

RNAMiner
A Bioinformatics Protocol for Mining Large RNA-Seq Transcriptomics Data
Home
Contact Us
FAQ
Test Data

Thank you for submitting request.
Sample 1(hf1.fastq.gz) uploaded Successfully
Sample 1(hf2.fastq.gz) uploaded Successfully
Sample 2(hm1.fastq.gz) uploaded Successfully
Sample 2(hm2.fastq.gz) uploaded Successfully

RNAMiner Prediction

Results will be emailed to jh7x3@mail.missouri.edu as soon as they are available - usually within several hours, though sometimes longer depending on the file size and the number of files.

Figure 5.21: One web page showing the successful submission of one request. After one request is submitted successfully, one web page will be shown which informs users that the data is in process.



RNAMiner

A Bioinformatics Protocol for Mining Large RNA-Seq Transcriptomics Data

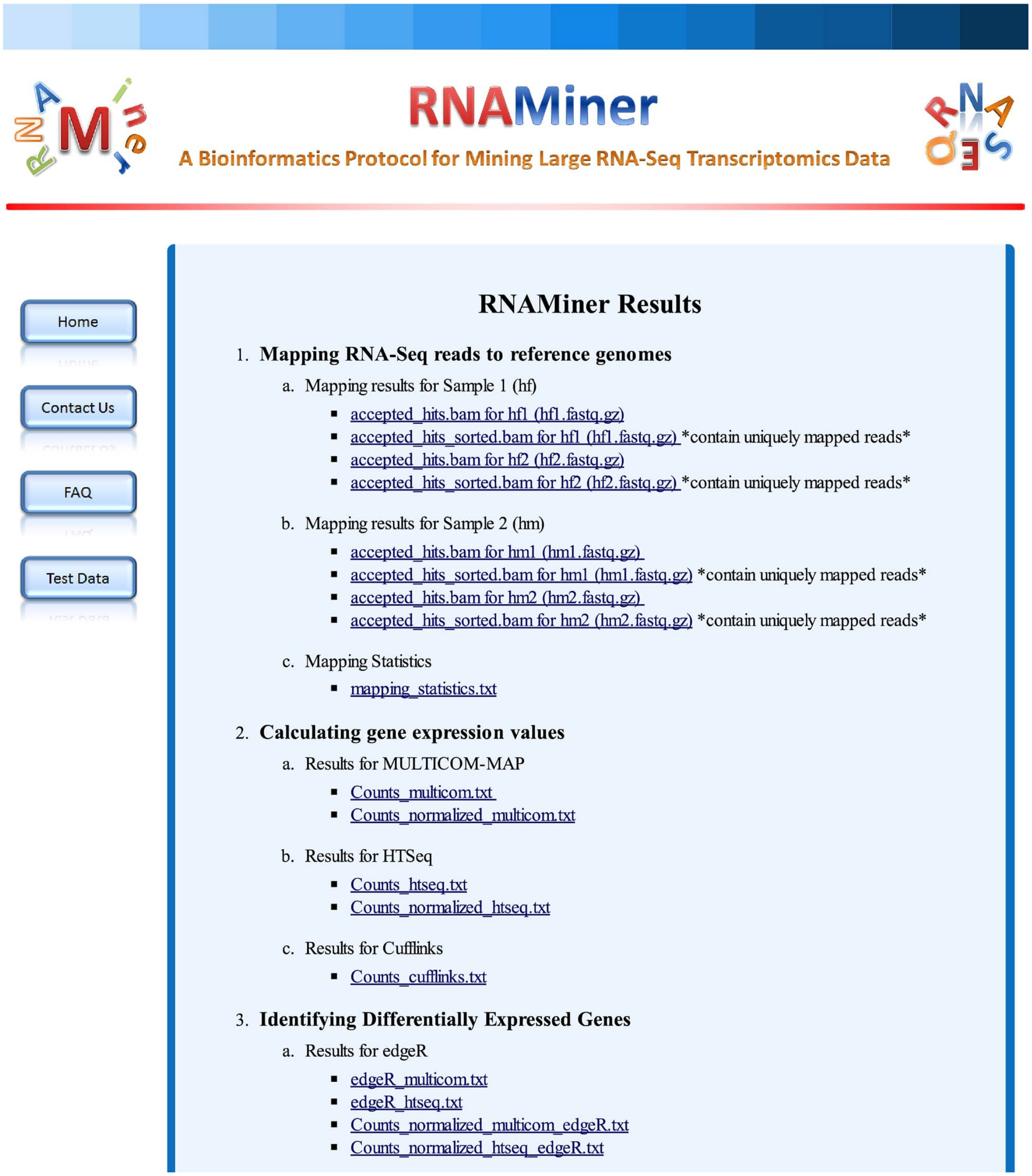
Analysis Information

Analysis Categories:	1. Mapping RNA-Seq reads to reference genomes 2. Calculating gene expression values 3. Identifying differentially expressed genes 4. Predicting gene functions 5. Constructing gene regulatory networks
Species:	Human
Criterion of identifying differentially expressed genes:	p-value
Threshold of p-value or q-value:	0.05
Sample 1 Name:	hf
Sample 1 file:	hf1.fastq.gz hf2.fastq.gz
Sample 1 Count:	2
Sample 2 Name:	hm
Sample 2 file:	hm1.fastq.gz hm2.fastq.gz
Sample 2 Count:	2

Analysis Result

ID:	1406139527
Email Address:	jh7x3@mail.missouri.edu
State:	Completed
Start time:	Wed Jul 23 13:18:47 2014
Finish time:	Wed Jul 23 15:40:33 2014
Result Link:	Check result here!

Figure 5.22: One web page with data analysis information and a result link. After the data analysis is finished, users will receive an email with a link to one web page with data analysis information and a result link. On this page, users can check the data analysis information and go to the result page.



RNAMiner

A Bioinformatics Protocol for Mining Large RNA-Seq Transcriptomics Data

RNAMiner Results

1. **Mapping RNA-Seq reads to reference genomes**
   a. Mapping results for Sample 1 (hf)
      - accepted_hits.bam for hf1 (hf1.fastq.gz)
      - accepted_hits_sorted.bam for hf1 (hf1.fastq.gz) *contain uniquely mapped reads*
      - accepted_hits.bam for hf2 (hf2.fastq.gz)
      - accepted_hits_sorted.bam for hf2 (hf2.fastq.gz) *contain uniquely mapped reads*

   b. Mapping results for Sample 2 (hm)
      - accepted_hits.bam for hm1 (hm1.fastq.gz)
      - accepted_hits_sorted.bam for hm1 (hm1.fastq.gz) *contain uniquely mapped reads*
      - accepted_hits.bam for hm2 (hm2.fastq.gz)
      - accepted_hits_sorted.bam for hm2 (hm2.fastq.gz) *contain uniquely mapped reads*

   c. Mapping Statistics
      - mapping_statistics.txt

2. **Calculating gene expression values**
   a. Results for MULTICOM-MAP
      - Counts_multicom.txt
      - Counts_normalized_multicom.txt

   b. Results for HTSeq
      - Counts_htseq.txt
      - Counts_normalized_htseq.txt

   c. Results for Cufflinks
      - Counts_cufflinks.txt

3. **Identifying Differentially Expressed Genes**
   a. Results for edgeR
      - edgeR_multicom.txt
      - edgeR_htseq.txt
      - Counts_normalized_multicom_edgeR.txt
      - Counts_normalized_htseq_edgeR.txt

Figure 5.23: One web page with data analysis results. Users can view and download the data analysis results for each analysis category on this web page.

Appendix A

Web-based Bioinformatics Tools and Services

A.1 MTMG: A Software Package for Multi-Template Protein Comparative Modeling

A.1.1 Overview

MTMG is a stochastic point cloud sampling method for multi-template protein model generation. The stochastic sampling and simulated annealing protocol in MTMG has the capability to improve the global quality and reduce atom clashes in models.

A.1.2 URL

The MTMG software package can be downloaded at http://sysbio.rnet.missouri.edu/multicom_toolbox/.

A.1.3 Input

The inputs of MTMG include a sequence alignment in .pir format and template structures of a target protein.

A.1.4 Output

MTMG outputs a protein 3D model with potentially lowest energy.

A.1.5 Software Architecture

MTMG is implemented using C++ and can be installed locally. It needs the pre-installed R for point sampling.

A.2 MULTICOM: A Web Server for Protein Tertiary Structure Prediction

A.2.1 Overview

MULTICOM is a large-scale conformation sampling and evaluation method and it can use a variety of alignment methods, template-based and template-free modeling methods, and a large number of protein model quality assessment methods to improve the reliability and robustness of protein structure prediction.

A.2.2 URL

http://sysbio.rnet.missouri.edu/multicom_cluster/.

A.2.3 Input

The input of MULTICOM must be the amino acid sequence of a target protein in FASTA format.

A.2.4 Output

MULTICOM outputs five protein 3D models predicted by the MULTICOM-CLUSTER server.

A.2.5 Software Architecture

PERL CGI was used in the server end of MULTICOM. It executes algorithms in the background and outputs the results by sending emails.

A.3 RNAMiner: A Web Service for RNA-Seq Data Analysis

A.3.1 Overview

RNAMiner can convert gigabytes of raw RNA-Seq data into kilobytes of valuable biological knowledge through a five-step data mining and knowledge discovery process.

A.3.2 URL

http://calla.rnet.missouri.edu/rnaminer/.

A.3.3 Input

The input of RNAMiner must be RNA-Seq reads files in FASTQ format. RNAMiner accepts replicates of reads files for one single sample or two samples in 6 species.

A.3.4 Output

RNAMiner outputs results from each step, which including mapping results (.bam) from TopHat and Bowtie, read counts from MULTICOM-MAP, HTSeq, and Cufflinks, differentially expressed genes from edgeR, DESeq, and Cuffdiff, predicted gene functions from MULTICOM-PDCN, and predicted gene regulatory networks from MULTICOM-GNET.

A.3.5 Software Architecture

PERL CGI was used in the server end of RNAMiner. It executes algorithms in the background and output the results by sending emails or directly displaying them at the webpage.

Bibliography

[1] Frank Eisenhaber, Bengt Persson, and Patrick Argos. Protein structure prediction: recognition of primary, secondary, and tertiary structural features from amino acid sequence. *Critical reviews in biochemistry and molecular biology*, 30(1):1–94, 1995.

[2] Burkhard Rost. Protein structure prediction in 1d, 2d, and 3d. *Encyclopedia of Computational Chemistry*, 1998.

[3] CA Floudas. Computational methods in protein structure prediction. *Biotechnology and bioengineering*, 97(2):207–213, 2007.

[4] Jesper Lundström, Leszek Rychlewski, Janusz Bujnicki, and Arne Elofsson. Pcons: A neural-network–based consensus predictor that improves fold recognition. *Protein Science*, 10(11):2354–2362, 2001.

[5] Björn Wallner, Huisheng Fang, and Arne Elofsson. Automatic consensus-based fold recognition using pcons, proq, and pmodeller. *Proteins: Structure, Function, and Bioinformatics*, 53(S6):534–541, 2003.

[6] Morten Källberg, Haipeng Wang, Sheng Wang, Jian Peng, Zhiyong Wang, Hui Lu, and Jinbo Xu. Template-based protein structure modeling using the raptorx web server. *Nature protocols*, 7(8):1511–1522, 2012.

[7] Liam J McGuffin. The modfold server for the quality assessment of protein structural models. *Bioinformatics*, 24(4):586–587, 2008.

[8] Hongyi Zhou and Yaoqi Zhou. Fold recognition by combining sequence profiles derived from evolution and from depth-dependent structural alignment of fragments. *Proteins: Structure, Function, and Bioinformatics*, 58(2):321–328, 2005.

[9] David T Jones. Genthreader: an efficient and reliable protein fold recognition method for genomic sequences. *Journal of molecular biology*, 287(4):797–815, 1999.

[10] Ambrish Roy, Alper Kucukural, and Yang Zhang. I-tasser: a unified platform for automated protein structure and function prediction. *Nature protocols*, 5(4):725–738, 2010.

[11] Jilong Li, Debswapna Bhattacharya, Renzhi Cao, Badri Adhikari, Xin Deng, Jesse Eickholt, and Jianlin Cheng. The multicom protein tertiary structure prediction system. *Methods Moleculer Biololy*, 1137:29–41, 2014.

[12] Davide Baú, Alberto JM Martin, Catherine Mooney, Alessandro Vullo, Ian Walsh, and Gianluca Pollastri. Distill: a suite of web servers for the prediction of one-, two-and three-dimensional structural features of proteins. *BMC bioinformatics*, 7(1):402, 2006.

[13] Kim T Simons, Charles Kooperberg, Enoch Huang, and David Baker. Assembly of protein tertiary structures from fragments with similar local sequences using simulated annealing and bayesian scoring functions. *Journal of molecular biology*, 268(1):209–225, 1997.

[14] Ali Mortazavi, Brian A Williams, Kenneth McCue, Lorian Schaeffer, and Barbara Wold. Mapping and quantifying mammalian transcriptomes by rna-seq. *Nature methods*, 5(7):621–628, 2008.

[15] Geng Chen, Charles Wang, and TieLiu Shi. Overview of available methods for diverse rna-seq data analyses. *Science China Life Sciences*, 54(12):1121–1128, 2011.

[16] Zhide Fang, JA Martin, and Zhong Wang. Statistical methods for identifying differentially expressed genes in rna-seq experiments. *Cell Bioscience*, 2(1):26, 2012.

[17] Charlotte Soneson and Mauro Delorenzi. A comparison of methods for differential expression analysis of rna-seq data. *BMC bioinformatics*, 14(1):91, 2013.

[18] Alicia Oshlack, Mark D Robinson, Matthew D Young, et al. From rna-seq reads to differential expression results. *Genome biol*, 11(12):220, 2010.

[19] Jian Peng and Jinbo Xu. A multiple-template approach to protein threading. *Proteins: Structure, Function, and Bioinformatics*, 79(6):1930–1939, 2011.

[20] Armin Meier and Johannes Söding. Automatic prediction of protein 3d structures by probabilistic multi-template homology modeling. *PLoS Comput Biol*, 11(10):e1004343, 2015.

[21] Roberto Sanchez and Andrej Sali. Evaluation of comparative protein structure modeling by modeller-3. *Proteins Structure Function and Genetics*, 29(s 1):50–58, 1997.

[22] Česlovas Venclovas and Mindaugas Margelevičius. Comparative modeling in casp6 using consensus approach to template selection, sequence-structure alignment, and structure assessment. *Proteins: Structure, Function, and Bioinformatics*, 61(S7):99–105, 2005.

[23] Per Larsson, Björn Wallner, Erik Lindahl, and Arne Elofsson. Using multiple templates to improve quality of homology models in automated homology modeling. *Protein Science*, 17(6):990–1002, 2008.

[24] Andrej Šali and Tom L Blundell. Comparative protein modelling by satisfaction of spatial restraints. *Journal of molecular biology*, 234(3):779–815, 1993.

[25] Andras Fiser and Andrej Šali. Modeller: generation and refinement of homology-based protein structure models. *Methods in enzymology*, 374:461–491, 2003.

[26] Torsten Schwede, Jürgen Kopp, Nicolas Guex, and Manuel C Peitsch. Swiss-model: an automated protein homology-modeling server. *Nucleic acids research*, 31(13):3381–3385, 2003.

[27] Michael Levitt. Accurate modeling of protein conformation by automatic segment matching. *Journal of molecular biology*, 226(2):507–533, 1992.

[28] Donald Petrey, Zhexin Xiang, Christopher L Tang, Lei Xie, Marina Gimpelev, Therese Mitros, Cinque S Soto, Sharon Goldsmith-Fischman, Andrew Kernyt-

sky, Avner Schlessinger, et al. Using multiple structure alignments, fast model building, and energetic analysis in fold recognition and homology modeling. *Proteins: Structure, Function, and Bioinformatics*, 53(S6):430–435, 2003.

[29] Jianlin Cheng. A multi-template combination algorithm for protein comparative modeling. *BMC Structural Biology*, 8(1):18, 2008.

[30] Jian Zhang and Yang Zhang. A novel side-chain orientation dependent potential derived from random-walk reference state for protein fold selection and structure prediction. *PloS one*, 5(10):e15386, 2010.

[31] Adam Zemla. Lga: a method for finding 3d similarities in protein structures. *Nucleic acids research*, 31(13):3370–3374, 2003.

[32] Yang Zhang and Jeffrey Skolnick. Scoring function for automated assessment of protein structure template quality. *Proteins: Structure, Function, and Bioinformatics*, 57(4):702–710, 2004.

[33] Domenico Cozzetto, Andriy Kryshtafovych, Krzysztof Fidelis, John Moult, Burkhard Rost, and Anna Tramontano. Evaluation of template-based models in casp8 with standard measures. *Proteins: Structure, Function, and Bioinformatics*, 77(S9):18–28, 2009.

[34] Yuanpeng J Huang, Binchen Mao, James M Aramini, and Gaetano T Montelione. Assessment of template-based protein structure predictions in casp10. *Proteins: Structure, Function, and Bioinformatics*, 82(S2):43–56, 2014.

[35] Jilong Li, Xin Deng, Jesse Eickholt, and Jianlin Cheng. Designing and benchmarking the multicom protein structure prediction system. *BMC structural biology*, 13(1):2, 2013.

[36] Jianlin Cheng, Jilong Li, Zheng Wang, Jesse Eickholt, and Xin Deng. The multicom toolbox for protein structure prediction. *BMC bioinformatics*, 13(1):65, 2012.

[37] Zheng Wang, Jesse Eickholt, and Jianlin Cheng. Multicom: a multi-level combination approach to protein structure prediction and its assessments in casp8. *Bioinformatics*, 26(7):882–888, 2010.

[38] Jilong Li, Badri Adhikari, and Jianlin Cheng. An improved integration of template-based and template-free protein structure modeling methods and its assessment in casp11. *Protein and peptide letters*, 22(7):586–593, 2015.

[39] Renzhi Cao, Debswapna Bhattacharya, Badri Adhikari, Jilong Li, and Jianlin Cheng. Large-scale model quality assessment for improving protein tertiary structure prediction. *Bioinformatics*, 31(12):i116–i123, 2015.

[40] Jilong Li, Renzhi Cao, and Jianlin Cheng. A large-scale conformation sampling and evaluation server for protein tertiary structure prediction and its assessment in casp11. *BMC bioinformatics*, 16(1):337, 2015.

[41] Johannes Söding. Protein homology detection by hmm–hmm comparison. *Bioinformatics*, 21(7):951–960, 2005.

[42] Robert D Finn, Jody Clements, and Sean R Eddy. Hmmer web server: interactive sequence similarity searching. *Nucleic acids research*, page gkr367, 2011.

[43] Andreas Biegert and Johannes Söding. Sequence context-specific profiles for homology searching. *Proceedings of the National Academy of Sciences*, 106(10):3770–3775, 2009.

[44] Bernard L Welch. The generalization of student's' problem when several different population variances are involved. *Biometrika*, 34(1/2):28–35, 1947.

[45] Vincent B Chen, W Bryan Arendall, Jeffrey J Headd, Daniel A Keedy, Robert M Immormino, Gary J Kapral, Laura W Murray, Jane S Richardson, and David C Richardson. Molprobity: all-atom structure validation for macromolecular crystallography. *Acta Crystallographica Section D: Biological Crystallography*, 66(1):12–21, 2010.

[46] Dong Xu and Yang Zhang. Improving the physical realism and structural accuracy of protein models by a two-step atomic-level energy minimization. *Biophysical journal*, 101(10):2525–2534, 2011.

[47] Yung Liang Tong. *The multivariate normal distribution.* Springer Science & Business Media, 2012.

[48] William N Venables and Brian D Ripley. *Modern applied statistics with S-PLUS.* Springer Science & Business Media, 2013.

[49] Scott Kirkpatrick, Mario P Vecchi, et al. Optimization by simmulated annealing. *science*, 220(4598):671–680, 1983.

[50] Vladimír Černỳ. Thermodynamical approach to the traveling salesman problem: An efficient simulation algorithm. *Journal of optimization theory and applications*, 45(1):41–51, 1985.

[51] Camillo J Taylor and David J Kriegman. Minimization on the lie group so (3) and related manifolds. *Yale University*, 1994.

[52] Steven Henikoff and Jorja G Henikoff. Amino acid substitution matrices from protein blocks. *Proceedings of the National Academy of Sciences*, 89(22):10915–10919, 1992.

[53] Piotr Rotkiewicz and Jeffrey Skolnick. Fast procedure for reconstruction of full-atom protein models from reduced representations. *Journal of computational chemistry*, 29(9):1460–1465, 2008.

[54] Georgii G Krivov, Maxim V Shapovalov, and Roland L Dunbrack. Improved prediction of protein side-chain conformations with scwrl4. *Proteins: Structure, Function, and Bioinformatics*, 77(4):778–795, 2009.

[55] Christian B Anfinsen, Edgar Haber, Michael Sela, and FH White Jr. The kinetics of formation of native ribonuclease during oxidation of the reduced polypeptide chain. *Proceedings of the National Academy of Sciences of the United States of America*, 47(9):1309, 1961.

[56] Manesh Shah, Sergei Passovets, Dongsup Kim, Kyle Ellrott, Li Wang, Inna Vokler, Philip LoCascio, Dong Xu, and Ying Xu. A computational pipeline for protein structure prediction and analysis at genome scale. *Bioinformatics*, 19(15):1985–1996, 2003.

[57] Brian G Fox, Celia Goulding, Michael G Malkowski, Lance Stewart, and Ashley Deacon. Structural genomics: from genes to structures with valuable materials and many questions in between. *Nature methods*, 5(2):129–132, 2008.

[58] Christian M-R Lemer, Marianne J Rooman, and Shoshana J Wodak. Protein structure prediction by threading methods: evaluation of current techniques. *Proteins: Structure, Function, and Bioinformatics*, 23(3):337–355, 1995.

[59] John Moult, Jan T Pedersen, Richard Judson, and Krzysztof Fidelis. A large-scale experiment to assess protein structure prediction methods. *Proteins: Structure, Function, and Bioinformatics*, 23(3):ii–iv, 1995.

[60] Cyrus Chothia and Arthur M Lesk. The relation between the divergence of sequence and structure in proteins. *The EMBO journal*, 5(4):823, 1986.

[61] Janusz M Bujnicki. Protein-structure prediction by recombination of fragments. *Chembiochem*, 7(1):19–27, 2006.

[62] Elmar Krieger, Sander B Nabuurs, and Gert Vriend. Homology modeling. *Methods of biochemical analysis*, 44:509–524, 2003.

[63] Jianlin Cheng, Jesse Eickholt, Zheng Wang, and Xin Deng. Recursive protein modeling: a divide and conquer strategy for protein structure prediction and its case study in casp9. *Journal of bioinformatics and computational biology*, 10(03):1242003, 2012.

[64] Stephen F Altschul, Thomas L Madden, Alejandro A Schäffer, Jinghui Zhang, Zheng Zhang, Webb Miller, and David J Lipman. Gapped blast and psi-blast:

a new generation of protein database search programs. *Nucleic acids research*, 25(17):3389–3402, 1997.

[65] Stephen F Altschul, Warren Gish, Webb Miller, Eugene W Myers, and David J Lipman. Basic local alignment search tool. *Journal of molecular biology*, 215(3):403–410, 1990.

[66] Richard Hughey and Anders Krogh. *SAM: Sequence alignment and modeling software system.* University of California at Santa Cruz, 1995.

[67] Frances C Bernstein, Thomas F Koetzle, Graheme JB Williams, Edgar F Meyer, Michael D Brice, John R Rodgers, Olga Kennard, Takehiko Shimanouchi, and Mitsuo Tasumi. The protein data bank: a computer-based archival file for macromolecular structures. *Archives of biochemistry and biophysics*, 185(2):584–591, 1978.

[68] Michael Remmert, Andreas Biegert, Andreas Hauser, and Johannes Söding. Hhblits: lightning-fast iterative protein sequence searching by hmm-hmm alignment. *Nature methods*, 9(2):173–175, 2012.

[69] Martin Madera. Profile comparer: a program for scoring and aligning profile hidden markov models. *Bioinformatics*, 24(22):2630–2631, 2008.

[70] Leszek Rychlewski, Weizhong Li, Lukasz Jaroszewski, and Adam Godzik. Comparison of sequence profiles. strategies for structural predictions using sequence information. *Protein Science*, 9(2):232–241, 2000.

[71] Lukasz Jaroszewski, Leszek Rychlewski, Zhanwen Li, Weizhong Li, and Adam Godzik. Ffas03: a server for profile–profile sequence alignments. *Nucleic acids research*, 33(suppl 2):W284–W288, 2005.

[72] Ruslan Sadreyev and Nick Grishin. Compass: a tool for comparison of multiple protein alignments with assessment of statistical significance. *Journal of molecular biology*, 326(1):317–336, 2003.

[73] Sitao Wu and Yang Zhang. Muster: improving protein sequence profile–profile alignments by using multiple sources of structure information. *Proteins: Structure, Function, and Bioinformatics*, 72(2):547–556, 2008.

[74] Robert C Edgar. Muscle: multiple sequence alignment with high accuracy and high throughput. *Nucleic acids research*, 32(5):1792–1797, 2004.

[75] Xin Deng and Jianlin Cheng. Enhancing hmm-based protein profile-profile alignment with structural features and evolutionary coupling information. *BMC bioinformatics*, 15(1):252, 2014.

[76] Yongchao Liu, Bertil Schmidt, and Douglas L Maskell. Msaprobs: multiple sequence alignment based on pair hidden markov models and partition function posterior probabilities. *Bioinformatics*, 26(16):1958–1964, 2010.

[77] Andrew Leaver-Fay, Michael Tyka, Steven M Lewis, Oliver F Lange, James Thompson, Ron Jacak, Kristian Kaufman, P Douglas Renfrew, Colin A Smith, Will Sheffler, et al. Rosetta3: an object-oriented software suite for the simulation and design of macromolecules. *Methods in enzymology*, 487:545, 2011.

[78] Liam J McGuffin and Daniel B Roche. Rapid model quality assessment for protein structure predictions using the comparison of multiple models without structural alignments. *Bioinformatics*, 26(2):182–188, 2010.

[79] Arjun Ray, Erik Lindahl, and Björn Wallner. Improved model quality assessment using proq2. *BMC bioinformatics*, 13(1):224, 2012.

[80] Björn Wallner and Arne Elofsson. Identification of correct regions in protein models using structural, alignment, and consensus information. *Protein Science*, 15(4):900–913, 2006.

[81] Zheng Wang, Jesse Eickholt, and Jianlin Cheng. Apollo: a quality assessment service for single and multiple protein models. *Bioinformatics*, 27(12):1715–1716, 2011.

[82] Zheng Wang, Allison N Tegge, and Jianlin Cheng. Evaluating the absolute quality of a single protein model using structural features and support vector machines. *Proteins: Structure, Function, and Bioinformatics*, 75(3):638–647, 2009.

[83] Arlo Randall and Pierre Baldi. Selectpro: effective protein model selection using a structure-based energy function resistant to blunders. *BMC structural biology*, 8(1):52, 2008.

[84] Min-yi Shen and Andrej Sali. Statistical potential for assessment and prediction of protein structures. *Protein science*, 15(11):2507–2524, 2006.

[85] Yuedong Yang and Yaoqi Zhou. Ab initio folding of terminal segments with secondary structures reveals the fine difference between two closely related all-atom statistical energy functions. *Protein science*, 17(7):1212–1219, 2008.

[86] Mingyang Lu, Athanasios D Dousis, and Jianpeng Ma. Opus-psp: an orientation-dependent statistical all-atom potential derived from side-chain packing. *Journal of molecular biology*, 376(1):288–301, 2008.

[87] Dmitry Rykunov and András Fiser. Effects of amino acid composition, finite size of proteins, and sparse statistics on distance-dependent statistical pair potentials. *Proteins: Structure, Function, and Bioinformatics*, 67(3):559–568, 2007.

[88] Andriy Kryshtafovych, Alessandro Barbato, Krzysztof Fidelis, Bohdan Monastyrskyy, Torsten Schwede, and Anna Tramontano. Assessment of the assessment: evaluation of the model quality estimates in casp10. *Proteins: Structure, Function, and Bioinformatics*, 82(S2):112–126, 2014.

[89] Andriy Kryshtafovych, Alessandro Barbato, Bohdan Monastyrskyy, Krzysztof Fidelis, Torsten Schwede, and Anna Tramontano. Methods of model accuracy estimation can help selecting the best models from decoy sets: assessment of model accuracy estimations in casp11. *Proteins: Structure, Function, and Bioinformatics*, 2015.

[90] Renzhi Cao, Zheng Wang, Yiheng Wang, and Jianlin Cheng. Smoq: a tool for predicting the absolute residue-specific quality of a single protein model with support vector machines. *BMC bioinformatics*, 15(1):120, 2014.

[91] Burkhard Rost, Jinfeng Liu, Dariusz Przybylski, Rajesh Nair, Kazimierz O Wrzeszczynski, Henry Bigelow, and Yanay Ofran. Predict protein structure through evolution. *Proteins*, 23(1995 Suppl 1):2, 1997.

[92] Wolfgang Kabsch and Christian Sander. Dictionary of protein secondary structure: pattern recognition of hydrogen-bonded and geometrical features. *Biopolymers*, 22(12):2577–2637, 1983.

[93] Robbie P Joosten, Tim AH Te Beek, Elmar Krieger, Maarten L Hekkelman, Rob WW Hooft, Reinhard Schneider, Chris Sander, and Gert Vriend. A series of pdb related databases for everyday needs. *Nucleic acids research*, 39(suppl 1):D411–D419, 2011.

[94] Johannes Söding, Andreas Biegert, and Andrei N Lupas. The hhpred interactive server for protein homology detection and structure prediction. *Nucleic acids research*, 33(suppl 2):W244–W248, 2005.

[95] Jianlin Cheng, Arlo Z Randall, Michael J Sweredoski, and Pierre Baldi. Scratch: a protein structure and structural feature prediction server. *Nucleic acids research*, 33(suppl 2):W72–W76, 2005.

[96] Liam J McGuffin, Kevin Bryson, and David T Jones. The psipred protein structure prediction server. *Bioinformatics*, 16(4):404–405, 2000.

[97] Yang Zhang and Jeffrey Skolnick. Tm-align: a protein structure alignment algorithm based on the tm-score. *Nucleic acids research*, 33(7):2302–2309, 2005.

[98] Hongyi Zhou and Yaoqi Zhou. Spem: improving multiple sequence alignment with sequence profiles and predicted secondary structures. *Bioinformatics*, 21(18):3615–3621, 2005.

[99] Hao Chen and Daisuke Kihara. Estimating quality of template-based protein models by alignment stability. *Proteins: Structure, Function, and Bioinformatics*, 71(3):1255–1274, 2008.

[100] Pascal Benkert, Silvio CE Tosatto, and Dietmar Schomburg. Qmean: A comprehensive scoring function for model quality assessment. *Proteins: Structure, Function, and Bioinformatics*, 71(1):261–277, 2008.

[101] Hongyi Zhou and Jeffrey Skolnick. Protein model quality assessment prediction by combining fragment comparisons and a consensus cα contact potential. *PROTEINS: Structure, Function, and Bioinformatics*, 71(3):1211–1218, 2008.

[102] Qiwen Dong, Yufei Chen, and Shuigeng Zhou. A machine learning-based method for protein global model quality assessment. *International Journal of General Systems*, 40(04):417–425, 2011.

[103] Björn Wallner and Arne Elofsson. Can correct protein models be identified? *Protein Science*, 12(5):1073–1086, 2003.

[104] Qingguo Wang, Kittinun Vantasin, Dong Xu, and Yi Shang. Mufold-wqa: A new selective consensus method for quality assessment in protein structure prediction. *Proteins: Structure, Function, and Bioinformatics*, 79(S10):185–195, 2011.

[105] Rafal Adamczak, Jaroslaw Pillardy, Brinda K Vallat, and Jaroslaw Meller. Fast geometric consensus approach for protein model quality assessment. *Journal of Computational Biology*, 18(12):1807–1818, 2011.

[106] Krzysztof Ginalski, Arne Elofsson, Daniel Fischer, and Leszek Rychlewski. 3d-jury: a simple approach to improve protein structure predictions. *Bioinformatics*, 19(8):1015–1018, 2003.

[107] Björn Wallner and Arne Elofsson. Pcons5: combining consensus, structural evaluation and fold recognition scores. *Bioinformatics*, 21(23):4248–4254, 2005.

[108] Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine learning*, 20(3):273–297, 1995.

[109] Jianlin Cheng, Zheng Wang, Allison N Tegge, and Jesse Eickholt. Prediction of global and local quality of casp8 models by multicom series. *Proteins: Structure, Function, and Bioinformatics*, 77(S9):181–184, 2009.

[110] Allison N Tegge, Zheng Wang, Jesse Eickholt, and Jianlin Cheng. Nncon: improved protein contact map prediction using 2d-recursive neural networks. *Nucleic acids research*, 37(suppl 2):W515–W518, 2009.

[111] Jianlin Cheng and Pierre Baldi. Three-stage prediction of protein β-sheets by neural networks, alignments and graph algorithms. *Bioinformatics*, 21(suppl 1):i75–i84, 2005.

[112] Lisa N Kinch, Shuoyong Shi, Hua Cheng, Qian Cong, Jimin Pei, Valerio Mariani, Torsten Schwede, and Nick V Grishin. Casp9 target classification. *Proteins: Structure, Function, and Bioinformatics*, 79(S10):21–36, 2011.

[113] Xin Deng, Jesse Eickholt, and Jianlin Cheng. Predisorder: ab initio sequence-based prediction of protein disordered regions. *BMC bioinformatics*, 10(1):436, 2009.

[114] Lin Wang, Pinghua Li, and Thomas P Brutnell. Exploring plant transcriptomes using ultra high-throughput sequencing. *Briefings in Functional Genomics*, 9(2):118–128, 2010.

[115] Vanessa M Kvam, Peng Liu, and Yaqing Si. A comparison of statistical methods for detecting differentially expressed genes from rna-seq data. *American journal of botany*, 99(2):248–256, 2012.

[116] Daehwan Kim, Geo Pertea, Cole Trapnell, Harold Pimentel, Ryan Kelley, and Steven L Salzberg. Tophat2: accurate alignment of transcriptomes in the presence of insertions, deletions and gene fusions. *Genome Biol*, 14(4):R36, 2013.

[117] Ben Langmead and Steven L Salzberg. Fast gapped-read alignment with bowtie 2. *Nature methods*, 9(4):357–359, 2012.

[118] Cole Trapnell, Brian A Williams, Geo Pertea, Ali Mortazavi, Gordon Kwan, Marijke J Van Baren, Steven L Salzberg, Barbara J Wold, and Lior Pachter. Transcript assembly and quantification by rna-seq reveals unannotated transcripts and isoform switching during cell differentiation. *Nature biotechnology*, 28(5):511–515, 2010.

[119] Simon Anders. Htseq: Analysing high-throughput sequencing data with python. *URL http://www-huber. embl. de/users/anders/HTSeq/doc/overview. html*, 2010.

[120] Gordon Robertson, Jacqueline Schein, Readman Chiu, Richard Corbett, Matthew Field, Shaun D Jackman, Karen Mungall, Sam Lee, Hisanaga Mark Okada, Jenny Q Qian, et al. De novo assembly and analysis of rna-seq data. *Nature methods*, 7(11):909–912, 2010.

[121] Simon Anders and Wolfgang Huber. Differential expression analysis for sequence count data. *Genome biol*, 11(10):R106, 2010.

[122] Lin Sun, Harvey R Fernandez, Ryan C Donohue, Jilong Li, Jianlin Cheng, and James A Birchler. Male-specific lethal complex in drosophila counteracts histone acetylation and does not mediate dosage compensation. *Proceedings of the National Academy of Sciences*, 110(9):E808–E817, 2013.

[123] Lin Sun, Adam F Johnson, Ryan C Donohue, Jilong Li, Jianlin Cheng, and James A Birchler. Dosage compensation and inverse effects in triple x metafemales of drosophila. *Proceedings of the National Academy of Sciences*, 110(18):7383–7388, 2013.

[124] Lin Sun, Adam F Johnson, Jilong Li, Aaron S Lambdin, Jianlin Cheng, and James A Birchler. Differential effect of aneuploidy on the x chromosome and genes with sex-biased expression in drosophila. *Proceedings of the National Academy of Sciences*, 110(41):16514–16519, 2013.

[125] Zheng Wang, Renzhi Cao, and Jianlin Cheng. Three-level prediction of protein function by combining profile-sequence search, profile-profile search, and domain co-occurrence networks. *BMC bioinformatics*, 14(Suppl 3):S3, 2013.

[126] Zheng Wang, Xue-Cheng Zhang, Mi Ha Le, Dong Xu, Gary Stacey, and Jianlin Cheng. A protein domain co-occurrence network approach for predicting protein function and inferring species phylogeny. *PloS one*, 6(3):e17906, 2011.

[127] Mingzhu Zhu, Xin Deng, Trupti Joshi, Dong Xu, Gary Stacey, and Jianlin Cheng. Reconstructing differentially co-expressed gene modules and regulatory networks of soybean cells. *BMC genomics*, 13(1):437, 2012.

[128] Belinda Giardine, Cathy Riemer, Ross C Hardison, Richard Burhans, Laura Elnitski, Prachi Shah, Yi Zhang, Daniel Blankenberg, Istvan Albert, James Taylor, et al. Galaxy: a platform for interactive large-scale genome analysis. *Genome research*, 15(10):1451–1455, 2005.

[129] Stephen A Goff, Matthew Vaughn, Sheldon McKay, Eric Lyons, Ann E Stapleton, Damian Gessler, Naim Matasci, Liya Wang, Matthew Hanlon, Andrew Lenards, et al. The iplant collaborative: cyberinfrastructure for plant biology. *Frontiers in plant science*, 2, 2011.

[130] Cole Trapnell, Lior Pachter, and Steven L Salzberg. Tophat: discovering splice junctions with rna-seq. *Bioinformatics*, 25(9):1105–1111, 2009.

[131] Thomas D Wu and Serban Nacu. Fast and snp-tolerant detection of complex variants and splicing in short reads. *Bioinformatics*, 26(7):873–881, 2010.

[132] Donna Karolchik, Robert Baertsch, Mark Diekhans, Terrence S Furey, Angie Hinrichs, YT Lu, Krishna M Roskin, Matthias Schwartz, Charles W Sugnet, Daryl J Thomas, et al. The ucsc genome browser database. *Nucleic acids research*, 31(1):51–54, 2003.

[133] Kim D Pruitt, Tatiana Tatusova, and Donna R Maglott. Ncbi reference sequences (refseq): a curated non-redundant sequence database of genomes, transcripts and proteins. *Nucleic acids research*, 35(suppl 1):D61–D65, 2007.

[134] Heng Li, Bob Handsaker, Alec Wysoker, Tim Fennell, Jue Ruan, Nils Homer, Gabor Marth, Goncalo Abecasis, Richard Durbin, et al. The sequence alignment/map format and samtools. *Bioinformatics*, 25(16):2078–2079, 2009.

[135] Predrag Radivojac, Wyatt T Clark, Tal Ronnen Oron, Alexandra M Schnoes, Tobias Wittkop, Artem Sokolov, Kiley Graim, Christopher Funk, Karin Verspoor, Asa Ben-Hur, et al. A large-scale evaluation of computational protein function prediction. *Nature methods*, 10(3):221–227, 2013.

[136] Michael Ashburner, Catherine A Ball, Judith A Blake, David Botstein, Heather Butler, J Michael Cherry, Allan P Davis, Kara Dolinski, Selina S Dwight, Janan T Eppig, et al. Gene ontology: tool for the unification of biology. *Nature genetics*, 25(1):25–29, 2000.

[137] Brigitte Boeckmann, Amos Bairoch, Rolf Apweiler, Marie-Claude Blatter, Anne Estreicher, Elisabeth Gasteiger, Maria J Martin, Karine Michoud, Claire O'Donovan, Isabelle Phan, et al. The swiss-prot protein knowledgebase and its supplement trembl in 2003. *Nucleic acids research*, 31(1):365–370, 2003.

[138] Alex Bateman, Lachlan Coin, Richard Durbin, Robert D Finn, Volker Hollich, Sam Griffiths-Jones, Ajay Khanna, Mhairi Marshall, Simon Moxon, Erik LL Sonnhammer, et al. The pfam protein families database. *Nucleic acids research*, 32(suppl 1):D138–D141, 2004.

[139] Alan Agresti and Maria Kateri. *Categorical data analysis*. Springer, 2011.

[140] Ronald A Fisher. The logic of inductive inference. *Journal of the Royal Statistical Society*, pages 39–82, 1935.

[141] Sir Ronald A Fisher. Confidence limits for a cross-product ratio. *Australian Journal of Statistics*, 4(1):41–41, 1962.

[142] Cyrus R Mehta and Nitin R Patel. Algorithm 643: Fexact: a fortran subroutine for fisher's exact test on unordered r× c contingency tables. *ACM Transactions on Mathematical Software (TOMS)*, 12(2):154–161, 1986.

[143] Douglas B Clarkson, Yuan-An Fan, and Harry Joe. A remark on algorithm 643: Fexact: An algorithm for performing fisher's exact test in rxc contingency tables. *ACM Transactions on Mathematical Software (TOMS)*, 19(4):484–488, 1993.

[144] WM Patefield. Algorithm as 159: An efficient method of generating random r× c tables with given row and column totals. *Applied Statistics*, pages 91–97, 1981.

[145] Gabor Csardi and Tamas Nepusz. The igraph software package for complex network research. *InterJournal, Complex Systems*, 1695(5):1–9, 2006.

[146] Ben Langmead, Cole Trapnell, Mihai Pop, Steven L Salzberg, et al. Ultrafast and memory-efficient alignment of short dna sequences to the human genome. *Genome biol*, 10(3):R25, 2009.

[147] Gerton Lunter and Martin Goodson. Stampy: a statistical algorithm for sensitive and fast mapping of illumina sequence reads. *Genome research*, 21(6):936–939, 2011.

[148] Thomas J Hardcastle and Krystyna A Kelly. bayseq: empirical bayesian methods for identifying differential expression in sequence count data. *BMC bioinformatics*, 11(1):422, 2010.

[149] Mark A Van De Wiel, Gwenaël GR Leday, Luba Pardo, Håvard Rue, Aad W Van Der Vaart, and Wessel N Van Wieringen. Bayesian analysis of rna sequencing data by estimating multiple shrinkage priors. *Biostatistics*, page kxs031, 2012.

[150] Sonia Tarazona, Fernando García-Alcalde, Joaquín Dopazo, Alberto Ferrer, and Ana Conesa. Differential expression in rna-seq: a matter of depth. *Genome research*, 21(12):2213–2223, 2011.

VITA

Jilong Li was born in Qiqihar, Heilongjiang, China. He received a Bachelor's degree of Computer Science from Heilongjiang University, China, in 2003, and a Master's degree of Computer Software and Theory from Harbin Engineering University, China, in 2008. He started his Ph.D. studies in the Computer Science Department at the University of Missouri in spring 2011. He received a job offer of Senior Developer at OmicSoft Corporation before he earned his Ph.D. degree at the University of Missouri.

Protein structure prediction and next-generation sequencing data analysis are two areas that he has been continuously working in since 2011. He started from next-generation sequencing data analysis and worked on protein structure prediction during his Ph.D. studies. He is also interested in other bioinformatics problems including protein secondary structure prediction, protein sequence alignment generation, and metabolic pathway prediction.