2012

COMPUTER METHODS FOR PRE-MICRORNA SECONDARY STRUCTURE PREDICTION

Dianwei Han
University of Kentucky, dianweih@hotmail.com

STUDENT AGREEMENT:

I represent that my thesis or dissertation and abstract are my original work. Proper attribution has been given to all outside sources. I understand that I am solely responsible for obtaining any needed copyright permissions. I have obtained and attached hereto needed written permission statements(s) from the owner(s) of each third-party copyrighted matter to be included in my work, allowing electronic distribution (if such use is not permitted by the fair use doctrine).

I hereby grant to The University of Kentucky and its agents the non-exclusive license to archive and make accessible my work in whole or in part in all forms of media, now or hereafter known. I agree that the document mentioned above may be made available immediately for worldwide access unless a preapproved embargo applies.

I retain all other ownership rights to the copyright of my work. I also retain the right to use in future works (such as articles or books) all or part of my work. I understand that I am free to register the copyright to my work.

REVIEW, APPROVAL AND ACCEPTANCE

The document mentioned above has been reviewed and accepted by the student's advisor, on behalf of the advisory committee, and by the Director of Graduate Studies (DGS), on behalf of the program; we verify that this is the final, approved version of the student's dissertation including all changes required by the advisory committee. The undersigned agree to abide by the statements above.

Dianwei Han, Student

Dr. Jun Zhang, Major Professor

Dr. Raphael Finkel, Director of Graduate Studies

COMPUTER METHODS FOR PRE-MICRORNA SECONDARY STRUCTURE PREDICTION

DISSERTATION

A dissertation submitted in partial fulfillment of the
requirements for the degree of Doctor of Philosophy in the
College of Engineering
at the University of Kentucky

By

Dianwei Han

Lexington, Kentucky

Director: Dr. Jun Zhang, Professor of Computer Science

Lexington, Kentucky

2012

ABSTRACT OF DISSERTATION

COMPUTER METHODS FOR PRE-MICRORNA SECONDARY STRUCTURE PREDICTION

This thesis presents a new algorithm to predict the pre-microRNA secondary structure. An accurate prediction of the pre-microRNA secondary structure is important in miRNA informatics. Based on a recently proposed model, nucleotide cyclic motifs (NCM), to predict RNA secondary structure, we propose and implement a Modified NCM (MNCM) model with a physics-based scoring strategy to tackle the problem of pre-microRNA folding. Our microRNAfold is implemented using a global optimal algorithm based on the bottom-up local optimal solutions.

It has been shown that studying the functions of multiple genes and predicting the secondary structure of multiple related microRNA is more important and meaningful since many polygenic traits in animals and plants can be controlled by more than a single gene. We propose a parallel algorithm based on the master-slave architecture to predict the secondary structure from an input sequence. The experimental results show that our algorithm is able to produce the optimal secondary structure of poly-cistronic microRNAs. The trend of speedups of our parallel algorithm matches that of theoretical speedups.

Conserved secondary structures are likely to be functional, and secondary structural characteristics that are shared between endogenous pre-miRNAs may contribute toward efficient biogenesis. So identifying conserved secondary structure is very meaningful and identifying conserved characteristics in RNA is a very important research field. After the characteristics are extracted from the secondary structures of RNAs, corresponding patterns or rules could be dug out and used.

We propose to use the conserved microRNA characteristics in two aspects: to improve prediction through knowledge base, and to classify the real specific microRNAs from pseudo microRNAs. Through statistical analysis of the performance of classification, we verify that the conserved characteristics extracted from microRNAs' secondary structures are precise enough.

Gene suppression is a powerful tool for functional genomics and elimination of specific gene products. However, current gene suppression vectors can only be used to silence a single gene at a time. So we design an efficient poly-cistronic microRNA vector and the web-based tool allows users to design their own microRNA vectors

online.

KEYWORDS: Secondary structure prediction, pre-microRNA, data mining, classification, clustering.

Dianwei Han

June 15, 2012

COMPUTER METHODS FOR PRE-MICRORNA SECONDARY STRUCTURE PREDICTION

By

Dianwei Han

Jun Zhang, Ph.D.

Director of Dissertation

Raphael Finkel, Ph.D.

Director of Graduate Studies

June 15, 2012

Date

RULES FOR THE USE OF DISSERTATIONS

Unpublished dissertations submitted for the Doctor's degree and deposited in the University of Kentucky Library are as a rule open for inspection, but are to be used only with due regard to the rights of the authors. Bibliographical references may be noted, but quotations or summaries of parts may be published only with the permission of the author, and with the usual scholarly acknowledgements.

Extensive copying or publication of the dissertation in whole or in part also requires the consent of the Dean of the Graduate School of the University of Kentucky.

A library that borrows this dissertation for use by its patrons is expected to secure the signature of each user.

Name	Date

ACKNOWLEDGEMENTS

The work with this dissertation has been extensive and trying, but in the first place it is exciting, instructive, and fun. Without help, support, and encouragement from other people, I would have never been able to finish this work. Here is my pleasure to express my gratitude to all of them.

First of all, I would like to thank my supervisor, Dr. Jun Zhang, for his inspiring and encouraging way to guide me to a deeper understanding of knowledge, and his invaluable comments during the whole work with this dissertation.

Besides my advisor, I would like to thank the rest of my Advisory Committee: Dr. Xuguo Zhou, Dr. Dakshnamoorthy Manivannan and Dr. Ruigang Yang who always give insightful comments and useful suggestions on my work. I would also like to thank the outside examiner Dr. Qiang Ye for his helpful comments on my dissertation.

Thanks also to all the friendly members in our lab who made the lab a great place to work. Let me say "thank you" to the following people: Dr. Ying Wang, Dr. Ning Kang, Dr. Wensheng Sheng, Dr. Shuting Xu, Dr. Eun-Joo Lee, Dr. Jie Wang, Mr. Ning Cao, Mr. Hao Ji, Mr. Pengpeng Lin, Ms. Ruxin Dai and other group members in our lab. Working together with all of you has been not only a unforgettable experience, but a great pleasure as well.

Last, but not least, I would like to thank my parents, for giving me life in the first place, for educating me, for unconditionally supporting and encouraging me to pursue my interests. Specially, I also would like to thank my wife for her love and support.

The research work with this dissertation was supported in part by:

- Kentucky New Economy Safety and Security Initiative (NESSI) Consortium.
- Kentucky Science and Engineering Foundation.

Table of Contents

List of Tables

List of Figures

1 Introduction

This thesis presents a new algorithm to predict the pre-microRNA secondary structure. Prior to 1982, RNA was thought to have three forms, which are mRNA, tRNA, and rRNA. mRNA carries the genetic information copied from DNA in the form of a series of three-base code words, each of which specifies a particular amino acid; tRNA is the adapter that connects the codons of the mRNA to the amino acids of the protein; and rRNA associates with a set of proteins to form ribosomes.

It has been shown that, for some functions, the absolute structure of the RNA involved is not critical. For example, the sequence of mRNA controls the synthesis of a given protein product; but the structure of the mRNA may not actually be important for this process to occur properly [113]. However, RNA molecules are involved in protein synthesis, and sometimes in order to understand the function of a given RNA molecule, scientists often need to know its structure because they believe that there is some relationship between the structure and function. RNA structures can be determined by X-ray crystallography and NMR spectroscopy, but producing RNA high-resolution structures by X-ray crystallography and NMR spectroscopy is slow compared to sequencing [94]. So prediction methods for RNA structures have to be developed and tested.

This chapter is organized as follows: MicroRNAs are introduced in Section 1.1. Each work that has been done and the motivations of this dissertation research are explained in Section 1.2. Finally, the contributions and organization of the dissertation are listed in Section 1.3.

1.1 MicroRNA

MicroRNAs (miRNAs) are newly discovered endogenous small non-coding RNAs (21-25nt) that are derived from larger hairpin RNA precursors and target their comple-

mentary gene transcripts for degradation or translational repression [16, 7, 46]. MicroRNAs are found to play an important role in regulation of gene expression in plants and animals [46]. Biologists assume that mammals have thousands of microRNAs in their genomes. MicroRNAs are expressed at different levels in animal and plant cells during cell differentiation, apoptosis, growth, and development [7, 16]. Understanding microRNA pathways and microRNA biogenesis is considered to be a crucial aspect in tool development for functional genomics and metabolic engineering.

At least three RNA species, primary miRNA (pri-miRNA), precursor miRNA (pre-miRNA), and mature miRNA, are made from miRNA genes through transcription and sequential endonucleolytic maturation steps [57]. Here, we focus on pre-miRNAs' secondary structure prediction.

1.2 Research work and the motivations

1.2.1 Predicting single microRNA structure

Basically, there are three kinds of structure of RNAs: primary structure, secondary structure, and three dimensional structure.

Primary structure is a sequence of nucleotides a, u, g and c. For example, GCUCUCGGAGAACAGGGAGCCACUCUGCGUUCACUCGGUGGGUAAUGAAGCGGGUGAACACAGCUGGUGGUAUCUCAGUUUUCUGAGGGC is the primary structure of ame-mir-317.

Secondary structure (two dimensional structure) is shown in Fig.1.1.

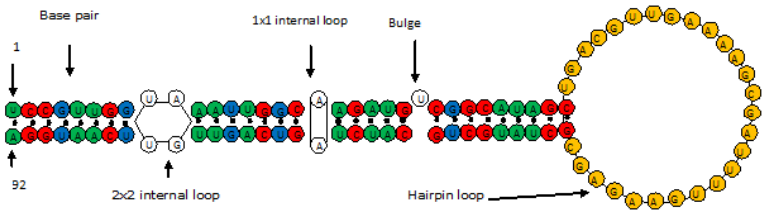


Figure 1.1: Secondary structure of a pre-microRNA

Primary sequence information helps people understand miRNA pathways. In-depth understanding of structure-function relationships requires knowledge of three-dimensional (3D) structure. It is very difficult and time-consuming to determine 3D structures for natural RNA molecules. In addition, it has been reported that miRNA genes are more conserved in the secondary structure than in the primary sequences [124]. Secondary structural features should be more fully exploited in the homologue search for new miRNA genes. RNA secondary structure can be predicted with some accuracy using computers and many bioinformatics applications use certain notions of secondary structure in the analysis of RNA [38].

There are some prediction methods, why do we still need to develop a new one? Generally there are two classes of algorithms available to predict the secondary structure of RNAs. The first class is the prediction methods based on phylogenetic sequence comparison, represented by Covariation prediction (Eddy et al.) [33] and Stochastic context free grammars (SCFG, Sakakibara et al.) [31, 105].

When not enough related sequences are available, however, the second class of methods must be used, which are based on thermodynamics and represented by free energy minimization method (Zuker et al.) [134] and partition function method (McCaskill et al.) [88].

The currently available leading prediction tools are designed for general RNA structure prediction, which do not consider much the features of the pre-miRNA secondary structures.

While the currently available leading prediction tools achieve good accuracies on true positive cases, their accuracies on Matthews coefficient ratio are relatively low.

1.2.2 Predicting multiple microRNAs' structure

For decades researchers have been studying the roles of a single gene in the development of animals and plants. However, recently, researchers have found that many

polygenic traits can be controlled by more than one single gene. In animals, it is well known that human hair, eye, and skin color are very complex and difficult to be predicted, because each of these traits is controlled by more than one gene. In plants, most polygenic traits of crops, which include disease tolerance, yield, stress tolerance, etc., are controlled by multiple genes [22, 108]. These genes can be regulated by different microRNAs. So studying the structures of these microRNAs definitely helps scientists better and more deeply understand the functions of microRNAs and the genes regulated by these microRNAs.

In addition, recent studies on developing RNAi technologies have shown that studying and analyzing the functions and structures of multiple microRNAs used to suppress specific genes are very important and meaningful, especially when endogenous microRNAs have been used to silence genes [55]. The discovery of RNAi and microRNA pathways has caused intensive studies on developing RNAi technologies for treating human diseases and for improving plant traits [54, 91, 115]. Currently available RNAi vectors [35] are designed to produce either short siRNAs, such as those produced by animal RNAi vectors, or long dsRNAs, such as those produced by plant RNAi vectors. Both animal and plant RNAi vectors have shown great successes in suppressing specific gene expression. In recent years, endogenous microRNAs have been designed to silence genes at high efficiency and in more gene specificity (Tang *et al.*.) [115]. Fortunately, this new type of RNAi vectors based on the microRNA structures provides us with a more stable and powerful tool for suppressing gene expression.

However, to our best knowledge, the currently available leading prediction tools are developed mainly for dealing with the general RNAs, and they could not directly been used to predict the structure of multiple microRNAs.

Based on the above observation, we think studying and designing an efficient algorithm to predict the structure of endogenous multiple microRNAs is needed. This

will pose us a big challenge. When a sequence is very long a sequential algorithm could not work very well. So a parallel algorithm is required. In Section 4, we demonstrate what we have done about this study.

1.2.3 Using conserved microRNA characteristics

Conserved secondary structures are likely to be functional, and secondary structural characteristics that are shared between endogenous pre-miRNAs may contribute toward efficient biogenesis [125]. So identifying conserved secondary structure is very meaningful and identifying conserved characteristics in RNA is a very important research field [5, 10, 125]. After the characteristics are extracted from the secondary structures of RNAs, corresponding patterns or rules could be dug out and used. For example, there are differences in the sequence variation between loop regions and helices. These patterns can be exploited in computational approaches [47] to discriminate functional RNAs from other types of conserved sequences [125].

We propose to use the conserved pre-microRNA structure characteristics in two aspects: to improve prediction, and to classify the real specific microRNAs from pseudo microRNAs.

In the aspect of machine learning, using information extracted from the data set to improve the prediction is a meaningful work. Especially when the conserved features, which are from secondary structures, are used to aid in prediction of other secondary structures, the performance of prediction system is expected to be improved. In addition, this strategy will strengthen the power of the original algorithm and make the prediction more accurate.

Furthermore, the conserved pre-microRNA structure features can be used in classification applications. We propose to carry out the classification of real and pseudo human microRNA precursors (pre-mirna). Through statistical analysis of the performance of classification, we can verify that the conserved characteristics extracted

from microRNAs' secondary structures are precise enough or not. The research work is introduced in detail in Sections 5 and 6.

1.2.4 A novel artificial poly-cistronic microRNA vector prediction and its application in silencing multiple genes in Arabidopsis

Gene suppression is a powerful tool for functional genomics and elimination of specific gene products. However, current gene suppression vectors can only be used to silence a single gene at a time. Moreover, several recent studies showed that siRNAs and long dsRNAs produced by these RNAi vectors tend to activate RNA-dependent protein kinase pathway and cause nonspecific cell death [12]. Metabolic engineering for novel plant natural products involves a regulation of carbon flow amongst many metabolic pathways and thus requires a silence of multiple genes of these pathways to redirect the metabolic flow to a specific pathway to overproduce specific gene products. In addition, traditional gene mutation, T-DNA insertion can not fulfill this objective in a short time. So, it is necessary for us to design a method through which artificial poly-cistronic microRNA vector can be predicted.

1.3 Contributions of the Dissertation

My research work for the dissertation is focused on studying and designing computer model to predict the secondary structure of pre-microRNAs. The contributions of the dissertation are:

- I propose a novel algorithm to predict the secondary structure of pre-microRNAs. It is the first computer model that combines the Modified NCM model with thermodynamic scoring strategy to deal with this domain problem.
- I propose a parallel algorithm to predict the secondary structure of endogenous polycistronic microRNAs. The actual speedups follow the theoretical speedup

trends.

- I propose a new effective method that can distinguish the real premiRNAs from pseudo pre-miRNAs and this approach is important for identifying novel and specific miRNAs. It is the first time to use characteristics from the structures themselves as features to perform classification.

- I propose to use Knowledge Base (KB) to support the pre-microRNA structure prediction. The production rules come from the secondary structure conserved characteristics and they are combined with fuzzy strategy. The prediction time has been reduced greatly due to the support from KB.

- I propose and implement a web-based application system that allows users to construct poly-cis miRNA vector online from gene sequence.

The organization of this dissertation is as follows:

- The main techniques we used in this thesis are introduced in Chapter 2.

  There, we introduce all the main techinques that are related to our research or some major techniques that we use in our research.

- In Chapter 3, we propose a novel algorithm that combines thermodynamics based scoring function with Modified NCM model to predict the secondary structure of pre-microRNAs. Instead of traditional dynamic programming method, we use iterative approach to handle the structure prediction problem.

- It has been shown that clustered miRNAs can be either encoded in a single polycistronic transcriptional unit or independently transcribed. In Chapter 4, we propose an efficient algorithm to predict the structures of poly-cistronic miRNAs. We also derive a series of theoretical speedups and analyze the actual speedups after the experiment.

- In order to study the important features that can be used to differentiate the real pre-miRNAs from other hairpin sequences with similar stem-loops, we carry out some experiments to classify the real pre-miRNAs and pseudo pre-miRNAs. In Chapter 5, we propose a prediction method which uses the characteristics from the secondary structures of pre-miRNAs as the features to do the classification with SVM.

- Chapter 6 is an extension of Chapter 5 in which we extracted important features and used those characteristics to construct the production rules. And these rules are used to construct the knowledge base to support the prediction.

- In Chapter 7, we propose a novel artificial poly-cistronic microRNA vector prediction and apply it to silence multiple genes in Arabidopsis.

- We conclude the dissertation in Chapter 8 and point out some directions for future work.

Two categories of prediction algorithms are available. The first class is the prediction methods based on phylogenetic sequence comparison, represented by Covariation prediction (Eddy *et al.*) [33] and Stochastic context free grammars (SCFG, Sakakibara *et al.*) [31][105]. Both of them are based on a probabilistic model and the assumption that large numbers of homologous sequences from different organisms are available. The second class of methods are based on thermodynamics and represented by free energy minimization method (Zuker et al.) [134] and partition function method (McCaskill et al.) [88].

Some of them will be introduced as follows.

In this chapter, we introduce several popular techniques that are related to our research domain. In Section 2.1, we introduce the Minimum Free Energy method. Then, Partition Function method is given in Section 2.2. Finally, the Stochastic Context Free Grammar technique is presented in Section 2.3.

2.1 MFE method

Minimum Free Energy is one of the deterministic methods and is the most popular method. There is a hypothesis, an RNA molecule will fold into a secondary structure that minimizes its free energy. The free energy of structure (at fixed temperature, ionic concentration) is the sum of each base pair and loop energies. Tables of pair and loop energies are used to calculate the energy of a structure. In MFE algorithm, the dynamic programming technique has been used.

W(i,j): energy of MFE structure from i to j.

V(i,j): energy of MFE structure from i to j, if any, in which the ith and jth bases are paired, otherwise ∞

$$\text{W(i,j)} = \begin{cases} 0, \text{ if } i = j \\ min\{V(i,j), min_{i \le k \le j}\{W(i,k) + W(k+1,j)\}\}, \\ \text{otherwise} \end{cases}$$

(2.1)

$$\text{V(i,j)} = \begin{cases} \infty, \text{ if pair at i, j is not AU, CG, or GU or if } j \le i+2 \\ -1 + min\{V(i+1,j-1), 1.1 + W(i+1,j-1)\} \\ otherwise \end{cases}$$

(2.2)

2.2 Partition function method

The free energy of a secondary structure is assumed additive in terms of its loops [88]

$$F(S) = \sum_{L \subset S} F_L. \tag{2.3}$$

where the free energies F_L have been obtained from experiments with model compounds.

According to the McCaskills Algorithm[88], the partition function Z can be computed as:

$$Z = \sum_{S \subset Q} e^{-\frac{F(S)}{RT}}. \tag{2.4}$$

where Q is a set of all possible states, $R = 8.3146$ joules per degree Kelvin, and T is absolute temperature in degree Kelvin.

The partition function can be related to thermodynamic properties because it has a very important statistical meaning. Boltzmann probability of a given secondary structure S_0 is

$$P_r[S_0] = \frac{e^{-\frac{F(S_0)}{RT}}}{Z}. \tag{2.5}$$

The partition function thus plays the role of a normalizing constant (note that it does not depend on s), ensuring that the probabilities sum up to one [95].

$$\sum_S P_s = \frac{\sum_S e^{-\frac{F(S)}{RT}}}{Z} = \frac{Z}{Z} = 1. \tag{2.6}$$

2.3 Stochastic Context Free Grammars (SCFG)

In the RNA secondary structure prediction problem, we are given an input sequence, and our goal is to predict the optimal structure based on the input. For the techniques based on probabilistic parsing, we need to compute the conditional probability $P(y|x)$, which is the probability of getting structure y given the sequence x [31].

SCFGs defines a set of transformation rules, and probability distribution over the transformation rules, and a mapping from derivations to secondary structures [31].

The following example shows how to predict the secondary structure using SCFGs.

(1) Transformation rules:

$$S \rightarrow aSu|uSa|cSg|gSc|gSu|uSg|aS|cS|gS|uS|\epsilon. \tag{2.7}$$

(2) Transformation probability. S $\rightarrow aSu$:P$_{S\rightarrow aSu}$.

(3) Mapping from parses to structures.

If the input is agucu and structure is ((.)), the parse is

$$S \rightarrow aSu \rightarrow agScu \rightarrow aguScu \rightarrow agucu. \tag{2.8}$$

The joint probability of getting parse σ is $P(x, \sigma)$.

$$P(x, \sigma) = P_{S\rightarrow aSu}.P_{S\rightarrow gSc}.P_{S\rightarrow uS}.P_{S\rightarrow \epsilon} = \frac{1}{11}^4.$$

3 Predicting single pre-microRNA structure

In this chapter, we propose a new pre-microRNA secondary structure prediction method based on Modified NCMs (MNCMs), which makes use of thermodynamics-based scoring function, implemented as a computer program: microRNAfold. MicroRNAfold employs a bottom-up algorithm to compute many local optimal solutions. The global optimal solution is produced by sorting these local optimal solutions. Our experimental results show that this algorithm is very efficient in predicting pre-microRNA secondary structure.

The structure of this chapter is as follows: In Section 3.1, we present the background to our research and research question. In Section 3.2, we introduce our MNCM model, a global optimal algorithm based on bottom-up local optimal solutions, and some evaluation metrics used in our study. The experiments and results are presented in Section 3.3. A brief discussion is given in Section 3.4. We sum up our work in Section 3.5.

3.1 Introduction

MicroRNAs (miRNAs) are newly discovered endogenous small non-coding RNAs (21-25nt) that are derived from larger hairpin RNA precursors and target their complementary gene transcripts for degradation or translational repression [7][16][46]. MicroRNAs are found to play an important role in regulation of gene expression in plants and animals [46]. Biologists assume that mammals have thousands of microRNAs in their genomes. MicroRNAs are expressed at different levels in animal and plant cells during cell differentiation, apoptosis, growth, and development [7][16]. Understanding microRNA pathways and microRNA biogenesis is considered to be a crucial aspect in tool development for functional genomics and metabolic engineering (Tang *et al.*) [114]. While the primary sequence information helps people understand miRNA path-

ways, in-depth understanding of structure-function relationships requires knowledge of three-dimensional (3D) structure. It is very difficult and time-consuming to determine 3D structures for natural RNA molecules. In addition, it has been reported that miRNA genes are more conserved in the secondary structure than in the primary sequences [124]. Secondary structural features should be more fully exploited in the homologue search for new miRNA genes. RNA secondary structure can be predicted with good accuracy using computers and many bioinformatics applications use certain notions of secondary structure in the analysis of RNA [38] [92].

Generally there are two classes of algorithms available to predict the secondary structure of RNAs. The first class is the prediction methods based on phylogenetic sequence comparison, represented by Covariation prediction (Eddy *et al.*) [33] and Stochastic context free grammars (SCFG, Sakakibara *et al.*) [31][105]. Both of them are based on a probabilistic model and the assumption that large numbers of homologous sequences from different organisms are available. When not enough related sequences are available, however, the second class of methods must be used, which are based on thermodynamics and represented by free energy minimization method (Zuker *et al.*) [134][135] and partition function method (McCaskill *et al.*) [88].

We examined several different theoretical strategies and studied their merits. Recent attempts to replace thermodynamics by statistical scores [11] led to similar or only slightly improved predictive power. More recently, Major *et al.* [94] proposed a new approach termed nucleotide cyclic motif (NCM), and developed MC-FOLD software to predict RNA structures. However, MC-FOLD can only deal with short RNA input sequences. In addition, there are several other RNA structural prediction software packages, such as Vienna package by Hofacker *et al.* [48], Mfold by Zuker *et al.* [134], CONTRAfold by Do *et al.* [31]. To better and more specifically predict pre-microRNA secondary structure, in our opinion, the following aspects remain to be further investigated:

(1) The currently available leading prediction tools are designed for general RNA structure prediction, which do not consider much the features of the pre-microRNA secondary structures. There is a need to develop a more specific tool for pre-microRNA secondary structure prediction.

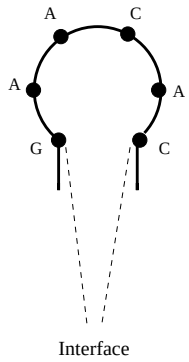
(2) While the currently available leading prediction tools achieve good accuracies on true positive cases, their accuracies on Matthews coefficient ratio [87] are relatively low.

Based on these observations, we decide to develop a new approach that will specifically deal with the pre-microRNA secondary structure prediction.

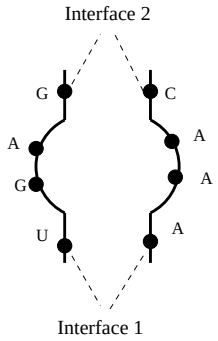
We propose a new pre-microRNA secondary structure prediction method based on Modified NCMs (MNCMs), which makes use of thermodynamics-based scoring function, implemented as a computer program: microRNAfold. MicroRNAfold employs a bottom-up algorithm to compute many local optimal solutions. The global optimal solution is produced by sorting these local optimal solutions. Our experimental results show that this algorithm is very efficient in predicting pre-microRNA secondary structure.

3.2 Prediction Methods

In this section, firstly, we demonstrate the use of MNCMs for RNA secondary structure prediction by showing how it arises as a natural extension of the recently developed NCMs. Secondly, we show how to convert from energy-based model to MNCMs, which is the hybrid model of traditional energy-based scoring schemes and MNCM structures. Finally, we introduce a global optimal algorithm which is based on the bottom-up local optimal solutions.



(a) Lone pair MNCM has one interface



(b) Double stranded MNCM has two interfaces

Figure 3.1: **The definition of interfaces for MNCMs.** The pair G·C is the only interface for the lone pair MNCM (a). The pair U·A is an interface for (b) and the pair G·C is another interface for (b).

3.2.1 Definition of NCMs

In the work of Major *et al.* [94], the NCM database contains lone pair NCMs and double-stranded NCMs. Lone pair NCMs are defined to be up to six nucleotides, which are denoted by the syntax "L- <sequence>", where L is the length of the loop and <sequence> is the sequence. There are 4 types and 5440 lone pair loops: 64 3-loops (3-AAA, 3-AAC, ..., 3-UUU); 256 4-loops; 1024 5-loops; and 4096 6-loops. On the other hand, they use the syntax "L1_L2-<sequence> " to denote double-stranded NCMs, where L1 is the length of the 5'-strand, L2 is the length of the 3'-strand, and <sequence> is the sequence. NCM-database contains 15 types and 407808 different double-stranded NCMs. For example, the 2_2-<sequence> NCMs represent 256 tandems: 2_2-AAAA, 2_2-AAAC, ..., 2_2-UUUU.

3.2.2 Definition of MNCMs

Compared to the definition of standard NCMs [94], we make some modifications (reasons are explained later in this section). We change the definition of NCMs based on the specific properties of microRNA precursors and the requirement of our algorithm. A valid lone pair MNCM must meet the following constraints:

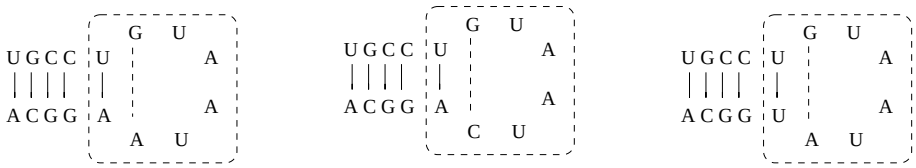
(1) The first nucleotide and the last nucleotide in a lone pair must be Watson-

Crick base pair or wobble base pair (G·U or U·G). This constraint is based on the requirement of the implementation of our algorithm. We extend the secondary structure of an miRNA by stacking MNCM blocks. We just consider canonical base pair as base pair. We propose a new term *interface*, which is the boundary pair between two MNCMs (MNCM is referred to as a specific nucleotide cyclic motif and to as a model as well). There is one interface for a lone pair MNCM. As shown in Fig. 3.1, G·C is the interface of a lone pair MNCM 5'GAACAC 3'. It is obvious that the first nucleotide and the last nucleotide in a lone pair form the interface. On the other hand, a double stranded MNCM has two interfaces. Consider a double stranded MNCM $\begin{smallmatrix}5'\text{UGAG}3'\\3'\text{AAAC}5'\end{smallmatrix}$ (see Fig. 3.1b) as an example, the pair U·A is interface1 and the pair G·C is interface2. In order to effectively employ our algorithm, we assume that all the interfaces should be a canonical base pair.

(2) The second unpaired pair is considered as the first mismatched pair of traditional minimal free energy algorithm. The second pair of a lone pair MNCM is the first mismatched pair of the hairpin loop according to the traditional thermodynamics-based models. The second constraint is based on the requirement of applying the traditional thermodynamics-based models.

(3) The length of a lone pair ranges from 3 to a half of the length of the given sequence. In fact, the hairpin loop of a pre-miRNA may be very long and it contains far more than 6 nucleotides. The third constraint is formulated based on the pre-miRNA feature, computational-based experiments by us and experimental results by the other research groups [131].

The definition of a lone pair MNCM is different from the one in MC-FOLD [94], and is not the same as the hairpin loop in traditional Minimal Free Energy (MFE) either. Fig. 3.2 depicts the selection of a valid lone pair. Let us focus on the parts within the dotted line rectangle (hairpin loop). (a) is a valid lone pair according to our preset rules. (b) is an invalid lone pair because the second pair G·C behind the



(a) (b) (c)

Figure 3.2: **The selection of valid lone pair.** (a) is a valid lone pair. (b) is an invalid lone pair. (c) is an invalid lone pair.

first pair U·A is a Watson-Crick base. (c) is an invalid lone pair because the first pair U·U is not a Watson-Crick base pair.

We use the same syntax "L1_L2-<sequence> " to denote the double-stranded MNCMs as Major *et al.* [94] do. But a valid double stranded MNCM structure must meet one constraint: Each interface of a double stranded MNCM must be a canonical base pair. The 2_2-<sequence> represents a base pairing tandem. For example, 2_2-CGCG represents a double stranded MNCM$^{5'\text{CG}3'}_{3'\text{GC}5'}$. The 3_2-<sequence> represents a 5'-strand single-nucleotide bulge, and the 2_3-<sequence> represents a 3'-strand single-nucleotide bulge. Similarly, the 4_4-<sequence> represents 2x2 internal loop. For example, 4_4-CAAGCGGG represents a double stranded MNCM$^{5'\text{CAAG}3'}_{3'\text{GGGC}5'}$.

3.2.3 From energy-based models to *MNCMs*

In order to describe the traditional energy-based model, we use an example.

Fig. 3.3 depicts the computation and model of free energy. This example is from Mathews *et al.* [84] and a similar strategy was adopted by Xia *et al.* [128]. The hairpin loop of four nucleotides AACA has an initiation of 5.6 kcal/mol. The dangling end (3'-most G) provides -1.3 kcal/mol of stability. The first mismatched pair A·A within dotted line in the hairpin loop is worth -1.1 kcal/mol. The 2x2 internal loop here destabilizes the structure, and its score is positive 1.0 kcal/mol.

The model that we use for our microRNAfold program is MNCMs. However, we use the experimentally measured thermodynamic parameters as scores instead

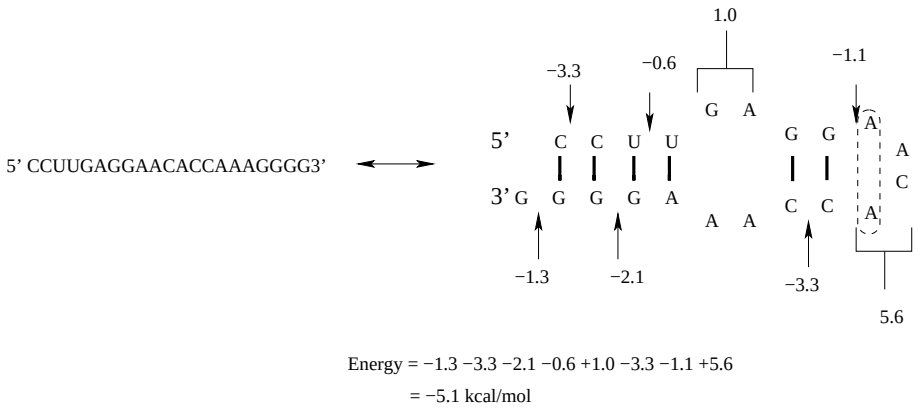


Figure 3.3: **Prediction of conformational free energy for an RNA.** The total free energy is the sum of each increment.

of the probabilities of each motif or their operations. We use the same example to show how to convert an energy-based model to an MNCM. Fig. 3.4 describes the procedure of constructing cycles (stems) for the structure of an RNA. We use this example to show how to construct the structure by MNCM model, and to show that the way the structure is constructed by using MNCM model is more natural and easily understood.

(1) Construct a lone pair (cycle m): 5'GAACAC 3'. As we mentioned earlier, the definition of our lone pair is different from that in the traditional energy-based model. The scoring function for a lone pair [86][128] is still adopted by the MNCM model:

$$f(\text{ lone pair}) = \begin{cases} \Delta G^o_{37initiation(n)} + \Delta G^o_{37}(\text{ stacking of the first mismatch}) \\ +\Delta G^o_{37bonus}(\text{ U·U or G·A first mismatch, but not A·G}) \\ +\Delta G^o_{37bonus}(\text{ special G·U closure}) \\ +\Delta G^o_{37penalty}(\text{ oligo-C loops}) \end{cases}$$

The only difference is that the first mismatch in traditional energy-based model is referred to as the second unpaired pair in MNCM model (see the 2nd constraint). (2) Construct a double stranded MNCM (cycle n): $\begin{matrix} 5'\text{GG}3' \\ 3'\text{CC}5' \end{matrix}$. For this double helix, we use the identical scoring rule as Mathews *et al.* [86] do.

(3) Merge cycle m and cycle n into cycle p: 5' GGAACACC 3'. We update the total score.

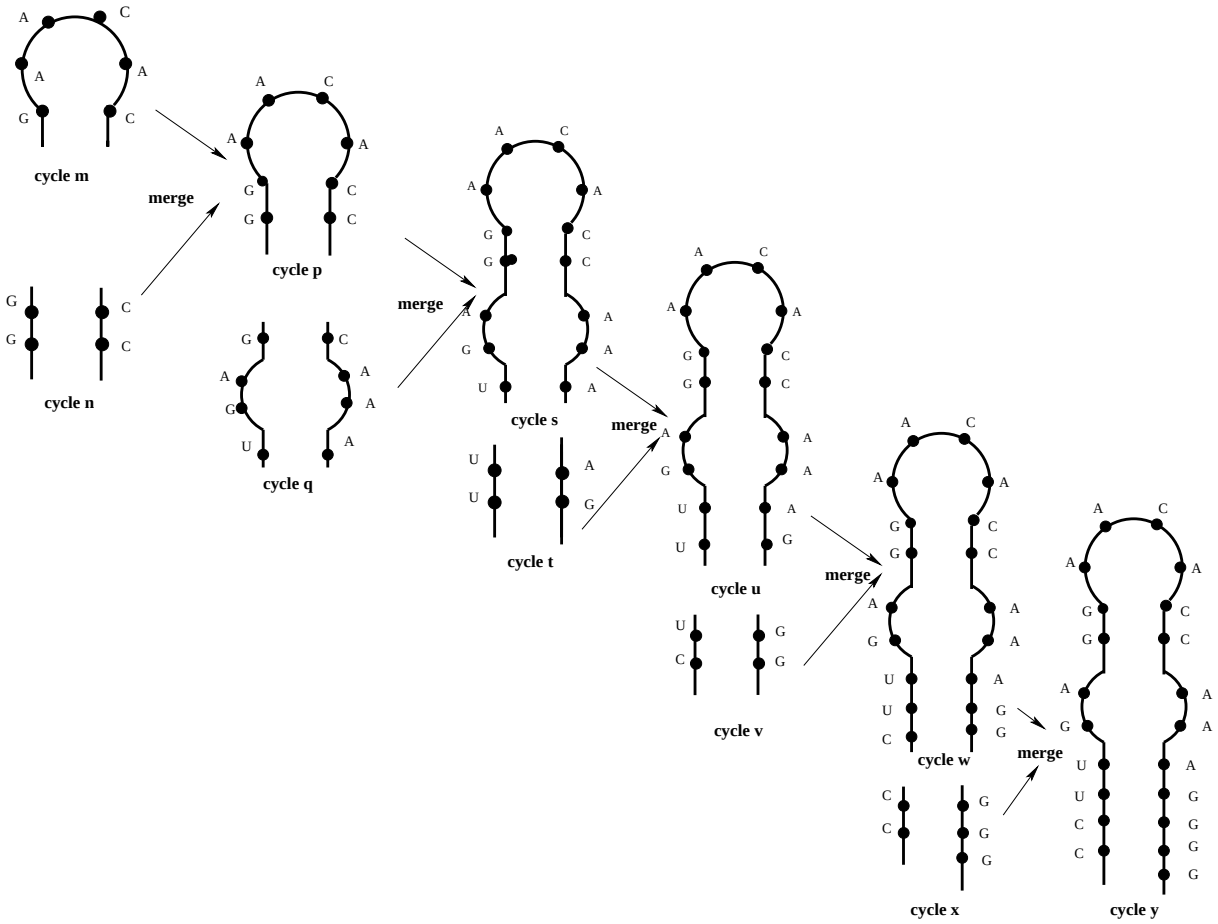


Figure 3.4: **The construction of cycles in the MNCM model.**

(4) Construct a double stranded MNCM (cycle q): $\begin{matrix} 5'\text{UGAG}3' \\ 3'\text{AAAC}5' \end{matrix}$. We use the same scoring function as the one used by Xia *et al.* [128] and Mathews *et al.* [86] for this tandem mismatches.

(5) Merge cycle p and cycle q into cycle s: 5' UGAGGAACACCAAA 3'. We update the total score. The similar scoring strategies are applied from step (6) to step (11).

(6) Construct a double stranded MNCM (cycle t): $\begin{matrix} 5'\text{UU}3' \\ 3'\text{GA}5' \end{matrix}$.

(7) Merge cycle s and cycle t into cycle u: 5' UUGAGGAACACCAAAG 3'.

(8) Construct a double stranded MNCM (cycle v): $\begin{matrix} 5'\text{CU}3' \\ 3'\text{GG}5' \end{matrix}$.

(9) Merge cycle u and cycle v into cycle w: 5' CUUGAGGAACACCAAAGG 3'.

(10) Construct a double stranded MNCM (cycle x): $\begin{matrix} 5'\text{CC}3' \\ 3'\text{GGG}5' \end{matrix}$.

(11) Merge cycle w and cycle x into cycle y: 5' CCUUGAGGAACACCAAAGGGG 3'.

3.2.4 Features of microRNAfold

The microRNAfold program applies MNCMs for pre-microRNA secondary structure prediction. The features in microRNAfold include:

(1) base pairs,

(2) helix closing base pairs,

(3) loop lengths,

(4) bulge loop lengths [36][42][80],

(5) internal loop lengths,

(6) internal loop asymmetry,

(7) terminal mismatch interactions, and

(8) dangling end.

Based on the features of the pre-microRNA structures, we do not deal with pseudo knots and multi-branch loops.

Generic base pairs

In order to shorten our parameter tables and simplify our model, we merge canonical base pairs and terminal mismatches into one category: base pair. In fact we just consider mismatches as non-canonical base pairs.

The 2x2 internal loops

We merge A·U/U·A cases and G·U/U·G cases together. We construct the table according to the publicly available data, and in other cases, just give the estimated value 2.8 [86] [128][129].

3.2.5 Soundness of combination of MNCM and MFE

Based on the specific features of pre-microRNA structure and the requirement of our algorithm, we modify the definition of NCMs and obtain MNCMs. For example, the hairpin loop of a pre-miRNA may be very long and it contains far more than 6

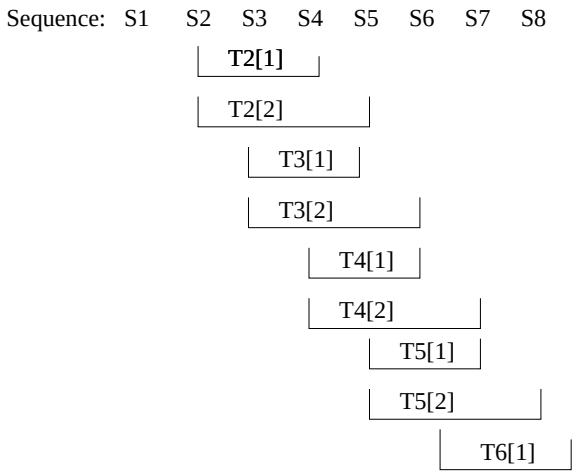


Figure 3.5: **The possible hairpin loops**. A sequence of nucleotides: S1, S2, .., S8. T2[1] represents one loop which starts with S2 and ends with S4. T2[2] represents another loop which starts with S2 and ends with S5.

nucleotides. In order to effectively implement our algorithm, we assume that all the interfaces should be a canonical base pair. By using MNCM, we deal with a block (at least four nucleotides) instead of a single nucleotide each time. In addition to its efficiency, construction of the structures can be viewed more naturally and much easier to understand. Compared to probabilistic-based methods, the MFE methods for modeling RNA structures have obtained higher accuracy. Furthermore, the thermodynamic parameters for free energy are publicly available. For these reasons, we propose to use MNCM model with MFE's scoring strategy to predict the secondary structure of microRNA precursors.

3.2.6 Global optimal algorithm based on bottom-up local optimal solutions

We implement our microRNAfold using a new recursive algorithm instead of the Waterman-Byers algorithm [127], which lists all near-optimal policies. We solve the problem by a backtracking method.

Construction of secondary structures

To generate a possible structure, we first determine the hairpin loop, which can be assigned a lonepair MNCM (see Fig. 3.5). The number of possible hairpin loops is the number of possible structures, which is $O(N^2)$, where N is the length of the sequence. In Fig. 3.5, there are eight nucleotides. There are two possible hairpin loops, which are T2[1] and T2[2], in the case that the starting nucleotide is S2. If the starting nucleotide is S3, S4, or S5, there are two possible hairpin loops. But if the starting nucleotide is S6, there is only one possible hairpin loop T6[1]. Then, we randomly select a valid double stranded MNCM from the rest of the sequence to merge it into the hairpin loop (lonepair MNCM) and get an extended structure. We do not use the optimal stem to construct the structure because we do not need to construct the minimum energy structure at this step. We update the scores after we merge, add a new stem each time, and repeat this process until we obtain a whole complete structure. That means there is no nucleotide in the rest of the sequence. This is the first structure generated with the input sequence. We obtain the optimal structure by backtracking over the stem variables in the first structure. The first structure is composed of many levels of stems. The top level of stems is the hairpin loop, and the bottom level of stems is the last stem of the structure. For the bottom level, we pick different candidate stem from a list P to substitute the old stem to reconstruct the structure. If it is possible to add this stem into the current structure and obtain another new whole structure, we continue the current construction and keep going toward the bottom; otherwise we try the next candidate stem or if no more stem in the list is available, we backtrack to the previous level, and then we go deeper. We repeat the process and generate another lists of structures. We stop until we reach the top level (hairpin loop). A schematic algorithm is given as follows in Algorithm 4.1.1.

Description of algorithm

ALGORITHM **3.2.1.** *Global optimal algorithm based on bottom-up local optimal solutions [1]*

1. for all possible starting stems
2. Do:
3. select one starting stem (hairpin loop)
4. while <current structure is not a valid complete structure ?>
5. Do:
6. construct a cycle as the next MNCM
7. if this cycle satisfies the preset rules, the current cycle is added into the current structure, else try the next cycle
8. endDo
9. construct a complete structure
10. apply a bottom-up algorithm to enumerate all the candidate solutions recursively sort them to obtain a local optimal solution

    Note: The local optimal solution is based on the specific hairpin loop
11. endDo
12. sort the sub-optimal structures according to their scores
13. obtain a global optimal solution from many possible sub-optimal structures

In Step 7, if the current cycle is satisfied with our preset rules which are different from the Waterman-Byers condition, we will add this current candidate cycle into the current structure. Preset rules specify that the interface of each cycle must be valid (canonical base pair). A searching program is used to guarantee that the path created from hairpin loop to current stem is a new one. Waterman-Byers condition

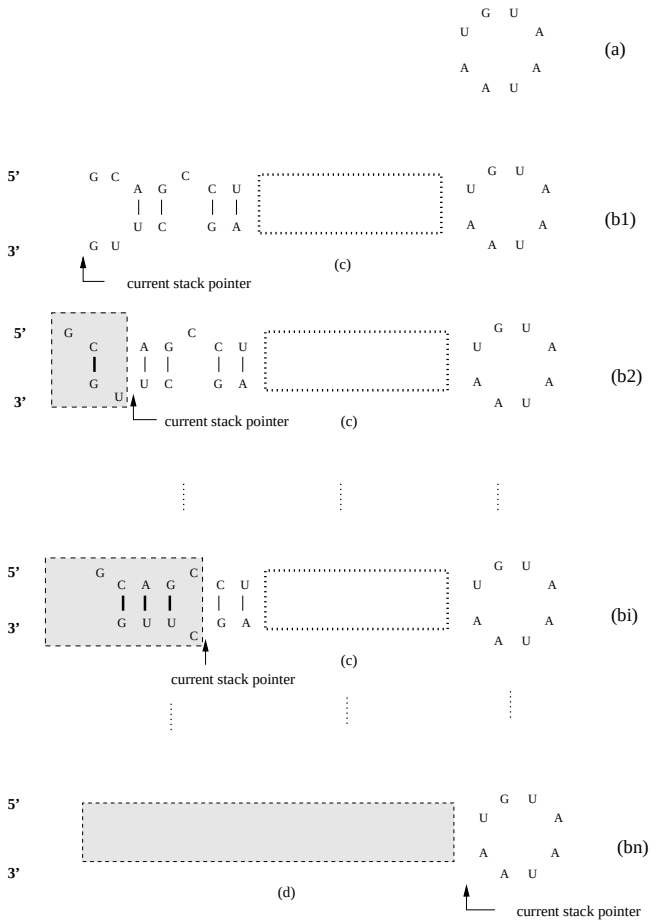


Figure 3.6: **The bottom-up algorithm**.(a) denotes a lone pair (the hairpin loop). (b1) displays the first initial structure that the program constructs with a given input sequence. (b2) denotes another structure when we backtrack the stack pointer. (bi) denotes that at the ith step, this structure is produced by the program. (bn) denotes the last structure that is built by the program. (c) shows the part of structure that remains unchanged from (b1) to (bi). (d) is the stem part of the last structure based on the current lone pair. Compared to the previous structures, the modified part is shown by the shadowed area.

requires an *a priori* threshold ε and reports all path of score less than $E_{min} + \varepsilon$. E_{min} is $E(1, N)$, which is the minimal energy, and is determined by dynamic programming. The Waterman-Byers strategy is to trace back all paths from the sink to the source in a recursive fashion. The essential idea of the Waterman-Byers algorithm is to limit the traceback to only those paths of score not greater than $E_{min} + \varepsilon$. One new cycle (stem) is added each time. Meanwhile, the total score needs to be adjusted. A backtracking technique is used here, which is a bottom-up algorithm.

As shown in Fig. 3.6, the bottom-up (BU) algorithm is introduced by using an example. We starts with a lone pair MNCM depicted in Fig. 3.6a. Then we repeatedly select a valid double stranded MNCM and add this MNCM to the current structure until we construct a complete structure (see Fig. 3.6b1). At this moment, the stack pointer is at the beginning. We consider the beginning as the bottom and the lone pair as the top or head. We backtrack to the previous MNCM and rebuild the next possible structure (see Fig. 3.6b2). When we compare (b2) to (b1), we notice that the shadowed part is modified. When we go deep toward the lone pair, we can construct the structures shown in Fig. 3.6bi to Fig. 3.6bn. The local optimal structure with the minimum score among the candidates (b1, b2, ..., bi, ..., bn) is chosen. Based on the different lone pairs, we obtain many different local optimal structures. The global optimal solution is obtained by applying the insertion-sort algorithm. The structure with the minimal score is the global optimal solution. Among the sorting algorithms, the insertion-sort algorithm is relatively easy to implement and seems better for small set. So we choose the insertion-sort algorithm.

3.2.7 Accuracy metrics

In order to precisely assess the predictive power of prediction methods, we use some typical measures, which have been extensively applied in the field of bioinformatics. Measures used in our study include True Positive rate, False Positive rate, True Negative rate, and False Negative rate, in addition to some more important metrics $Matthews$, $Sensitivity$, and $Specificity$. $Sensitivity$ is defined as

$$Sensitivity = \frac{\text{number of correct base pairings}}{\text{number of true base pairings}} \quad . \tag{3.1}$$

We see that sensitivity is equal to True Positive rate here.

$Specificity$ is defined as

$$Specificity = \frac{\text{number of correct base pairings}}{\text{number of predicted base pairings}} \quad . \tag{3.2}$$

Matthews coefficient ratio [87] is written as:

$$Matthews = \sqrt{\frac{TP \times TP}{(TP + FN) \times (TP + FP)}} \quad , \tag{3.3}$$

where FP is the number of false positive cases, FN is the number of false negative cases, and TP is the number of true positive cases.

3.2.8 Our microRNAfold accessibility

Web service is freely available at http://www.cs.uky.edu/∼ dianweih/rnaprediction/server.html.

3.3 Experimental Results

We evaluated the predictive power of microRNAfold by using known secondary structures of non-coding RNA taken from the miRBase database [39][40][41]. Our testing data set came from *Arabidopsis thaliana, Brassica napus, Triticum aestivum, Homo sapiens, Gallus gallus, Glycine max, Apis mellifera, Drosophila melanogaster, and Physcomitrella patens*. The pre-microRNAs' names are listed in Fig. 3.8. The sequence lengths of the testing data set range from 63 to 184. We implemented the microRNAfold by using ANSI C code and the program was run on a Linux-based machine. We used Pseudoviewer to view our structures (http://pseudoviewer.inha.ac.kr/) [98]. Our best solution was from the first one among several hundreds of sorted possible structures (see Fig. 3.7).

3.3.1 Predictive power of microRNAfold associated with different sequence lengths

In our study, we considered only AU, GC, and GU base pairs because there is no sufficient knowledge concerning the non-canonical base pairs even though non-canonical base pairs might be important and play some roles in determining 3D structures of RNAs [94].

> MicroRNAfold (input file is from hsa−let−7a sequence)

Top 5 scores:

---------- **RESULT** ----------

Sequence : UGGGAUGAGGUAGUAGGUUGUAUAGUUUUAGGGUCACACCCACCACUGGGAGAUAACUAUACAAUCUACUGUCUUUCCUA

```
(((((.(((((((((((((((((((.........................)))))))))))))))))))))))) -14.037429

(((((.(((((((((((((((((((.(.....................).)))))))))))))))))))))))) -13.846993

(((((.(((((((((((((((((((.((.....................))..)))))))))))))))))))))))) -10.686993

(((((.(((((((((((((((((((.........................).)))))))))))))))))))))))) -7.517427

(((((.(((((((((((((((((...((.......................))..)))))))))))))))))))))) -7.516992
```

Figure 3.7: **microRNAfold predictions for hsa-let-7a.** The top five structures generated by microRNAfold for hsa-let-7a. The structures are shown in dot-bracket notation. A parenthesis represents a canonical base pair; a dot represents an unpaired nucleotide. A dot-bracket can be converted to a secondary structure representation. Negative floating point numbers on the right hand side denote the corresponding scores.

Fig. 3.9 shows the specific performance of microRNAfold based on different sequence lengths. It shows that our microRNAfold does best when the lengths of the input sequences range from 85 to 110 which are the average lengths for normal microRNA precursors. The Matthews value in case (b) increases to 95% from 94%, compared to case (a). Compared to case (a) and case (b), the Matthews value in case (c) drops dramatically and the True Positive rate drops slightly. It makes sense because it is always a challenge for a prediction approach when the input sequence is very long.

3.3.2 Predictive power of microRNAfold associated with different hairpin loop lengths

Fig. 3.10 depicts the comparison of the microRNAfold performance based on the hairpin loop lengths. We just used the Matthews value as the metric to assess the effectiveness and the performance of our method. It seems that microRNAfold obtained the best Matthews coefficient ratio when the hairpin loop is relatively small.

Human (10)	**Arabidopsis (12)**	**Soybean (4)**	**Moss (4)**
hsa−let−7a	ath−mir−157a	gma−mir−156d	ppt−mir−156a
hsa−mir−24−1	ath−mir−158a	gma−mir−396a	ppt−mir−1048
hsa−mir−31	ath−mir−165a	gma−mir−482	ppt−mir−1215
hsa−mir−101−2	ath−mir−166a	gma−mir−1513	ppt−mir−2084
hsa−mir−139	ath−mir−168a		
hsa−mir−151	ath−mir−171a	**Colza (4)**	**Honeybee (3)**
hsa−mir−196a	ath−mir−393a	bna−mir−161	ame−let−7
hsa−mir−299	ath−mir−396a	bna−mir−166a	ame−mir−282
hsa−mir−491	ath−mir−771	bna−mir−169a	ame−mir−317
hsa−mir−518a	ath−mir−778	bna−mir−1140a	
	ath−mir−782		
	ath−mir−1886		
D.melanogaster (4)	**Fowl (Gallus gallus) (4)**	**Wheat (4)**	
dme−mir−3	gga−mir−16−1	tae−mir−159a	
dme−mir−5	gga−mir−18a	tae−mir−1121	
dme−mir−31a	gga−mir−24	tae−mir−1131	
dme−mir−284	gga−mir−26a	tae−mir−1136	

Figure 3.8: **The pre-microRNAs used in our test set.**

For example, in case (a), TP is 769, FN is 52 , and FP is 29, so the Matthews value is $\sqrt{\frac{TP \times TP}{(TP+FN)\times(TP+FP)}} = 95\%$.

3.3.3 Comparison to other methods

We compared the performance of microRNAfold with the other three leading methods: two adopt the probabilistic-based strategy, and the other chooses the free energy minimization strategy. For benchmarking experiments, we used MC-FOLD [94], CONTRAfold [31], and Mfold (http://mfold.bioinfo.rpi.edu/cgi-bin/rna-form1.cgi) [134], with default parameters for each program. All benchmarks were conducted on Intel-based servers running a GNU/Linux operating system. Whenever a program returned multiple possible structures (e.g., Mfold and MC-FOLD), we chose the structure with the minimum score.

Fig. 3.11 shows the comparison of the predictive power of different methods. Compared to the thermodynamic approach and the probabilistic methods, MicroRNAfold obtained a higher Matthews coefficient ratio, a higher True Negative rate and a lower

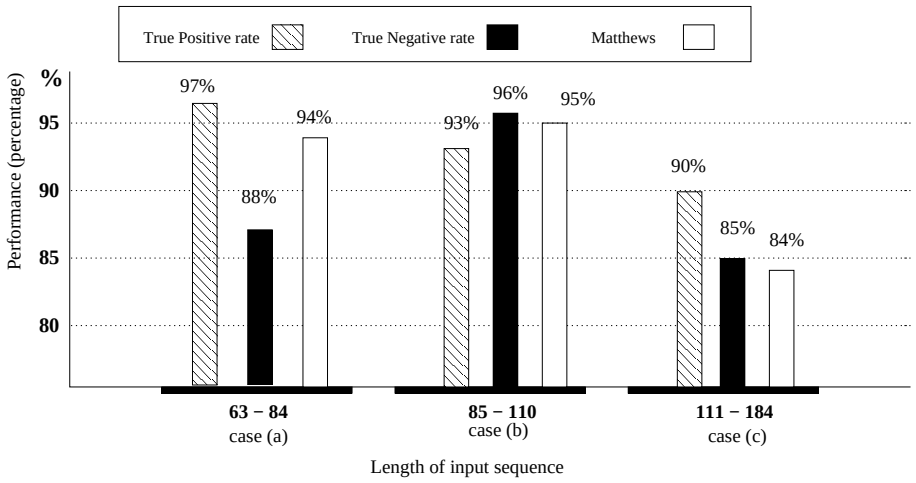


Figure 3.9: **The specific performance of microRNAfold based on different sequence lengths.** We divide the sequence lengths into three groups: case(a) (63-84), case(b) (85-110), and case(c) (111-184).

False Negative rate, despite a lower True Positive rate and a higher False Positive rate. In particular, microRNAfold achieved statistically significant improvements of over 11% in specificity relative to the best current method, Mfold. In the aspect of the True Positive rate, MC-FOLD, CONTRAfold, and MFold worked better than microRNAfold.

A paired t-test was performed to determine if the difference between the specificity of microRNAfold and that of MFOLD is significant. The mean difference (M=0.1147, SD =0.1588, N= 49) was significantly greater than zero, t(49)=5.06, two-tail p = 0.00007, providing evidence that microRNAfold achieved statistically significant improvements in specificity relative to MFold. The confidence level is 95%.

We also constructed an auROC plot shown in Fig. 3.12. The best possible prediction method would yield a point in the upper right corner or coordinate (1,1) of the ROC space. A completely random guess would give a point along the diagonal line from the right bottom to the top left corners. The microRNAfold method clearly shows the best among MC-FOLD, CONTRAfold, Mfold, and microRNAfold.

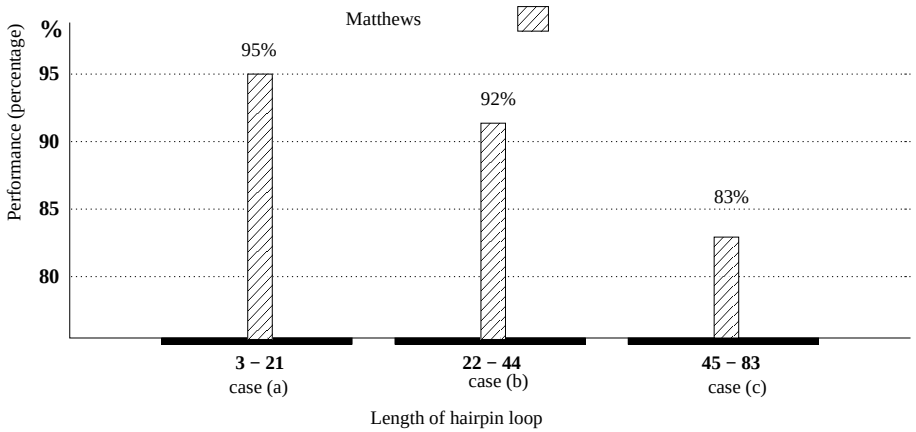


Figure 3.10: **The Matthews coefficient ratio performance of microRNAfold based on different hairpin loop lengths.** We divide the loop lengths into three groups: case(a), case(b), and case(c). The case(a) indicates that the length range is 3-21, the case(b) indicates that the length range is 22-44, and the case(c) indicates that the length range is 45-83.

3.4 Discussion

Although we have obtained encouraging results compared to other prediction approaches, there are still some issues that need to be discussed in detail. The first thing that we would like to mention is the auxiliary information. As we know, all the parameters and understanding of RNA secondary structure come from experimental results. Experimental results and the related analysis based on the experimental facts may help us design a more accurate model and prediction algorithm. How to get this knowledge is still a challenge for us. The second thing is the scoring strategy. During our testing phase, we found some proposed structures from the database could not be generated from our results based on the current scoring function.

3.4.1 Taking into account auxiliary information and more parameters

Sometimes we could not successfully predict the secondary structure of an RNA because "our knowledge of the contributions of various RNA motifs to the total free energy of RNA structures is still incomplete" [128]. Due to the limitation of this

Prediction Methods:	MC−FOLD (NCM)	CONTRAfold (probabilistic_based)	MFold (Thermodynamics)	microRNAfold (Modified NCM plus thermodynamics)
True Positive rate = Sensitivity	93.91 %	96.51 %	**97.99 %**	92.61 %
False Positive rate	6.09 %	3.49 %	**2.01%**	7.39 %
True Negative rate	41.18 %	69.20 %	70.76 %	**89.13 %**
False Negative rate	58.82 %	30.80 %	29.24 %	**10.87 %**
Specificity	61.92 %	75.25 %	78.22 %	**89.69 %**
Matthews coefficient ratio	73.11 %	83.42 %	84.82%	**90.04 %**

Figure 3.11: **Comparison of the predictive power with other prediction methods.** The predictions are compared over 1651 base pairs. For each approach, the best predicted structures are analyzed. In each row, we use bold font to represent the best value. MC-FOLD software is available at http://www.major.iric.ca/MC-Tools.html. CONTRAfold is available at http://contra.stanford.edu/contrafold/server.html.

kind of knowledge, we could not give all the thermodynamic parameters concerning free energy. Thus it could affect our prediction negatively. For example, when we predicted the microRNA precursor hsa-mir-196a, we failed to achieve the proposed structure of $\begin{matrix} 5'\text{UUAG}3' \\ 3'\text{AGCC}5' \end{matrix}$.

Fig. 3.13 shows the comparison of the predicted structure by microRNAfold with the structure proposed by the database. According to our current scoring function, the sum of the two parts (b1) and (b2) is -0.07 and the score of the structure (c) is 0.95. Therefore, we took (b) as the solution. When we calculated the score for the structure (c), we used the following formula proposed by Xia *et al.* [129]:

$$\Delta G^o_{37predict}(c) = [\Delta G^o_{37loop}(c1) + \Delta G^o_{37loop}(c2)] * \frac{1}{2} + \Delta. \tag{3.4}$$

In order to solve this problem, we need to refine scoring function [4][31] or incorporate auxiliary information. Based on the statistical and theoretical analysis of the experimental data, we may incorporate biological constraints to help the prediction [135].

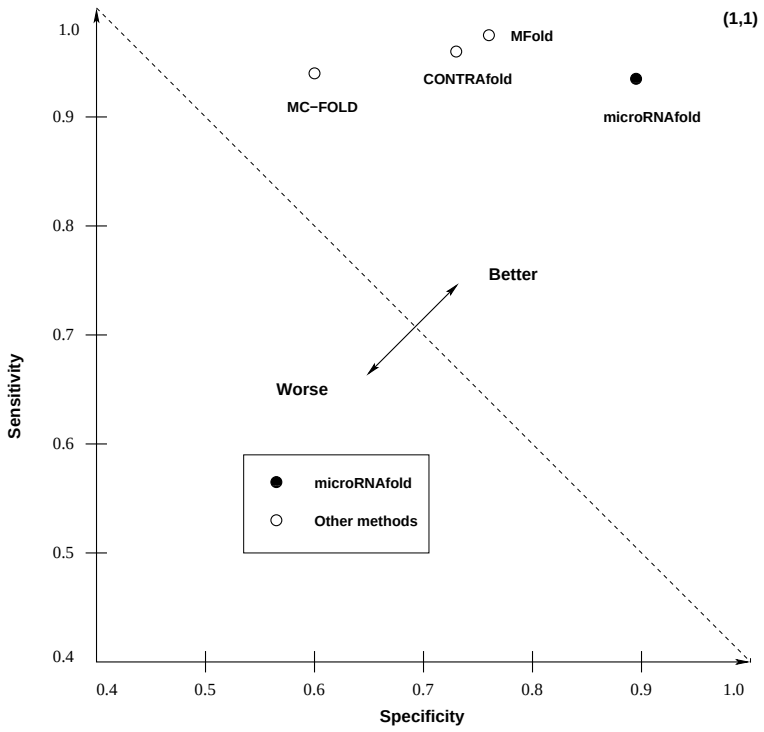


Figure 3.12: **ROC plot comparing sensitivity and specificity for several RNA structure prediction methods.**

3.4.2 Some issues with scoring strategy

During our study, we found that some of the best structures did not come from the first structure whose score is minimal. For example, according to the miRBase database, the structure of pre-microRNA dps-mir-6-3 should be (b) in Fig. 3.14. But the result from microRNAfold showed that the proposed structure should be (a) in Fig. 3.14.

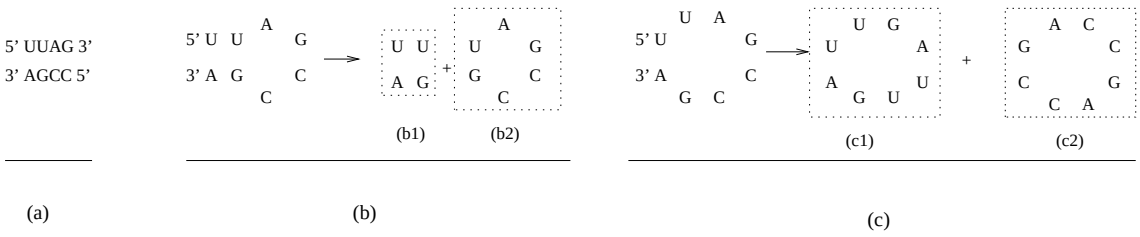


Figure 3.13: **Prediction of a specific structure.** (a) is the sequence of the structure, (b) is the predicted structure by microRNAfold, and (c) is the proposed structure by the database.

MicroRNAfold's result on dps−mir−6−3

Sequence: CAAAAAGAAGGGAACGGUUGCUGCUGAUGUAGUUCAAGUUUUGCACAAUUUAUAUCACAGUGGCUGUUCUUUUUUGUUUG

```
(.(((((((((((((((((((.(((((((((.................)))))))))))))))))))))))))).))))        (a)

((((.((((((((((((..(((.(((((((.....(.....).....)))))))))))..))))))))))))).))))        (b)
```

(c)

Figure 3.14: **The result of microRNAfold with the input of the pre-microRNA dps-mir-6-3.** (a) shows the best structure predicted by microRNAfold. (b) is the proposed structure by the database. (c) shows the hairpin loop of (a) and the corresponding area of (b).

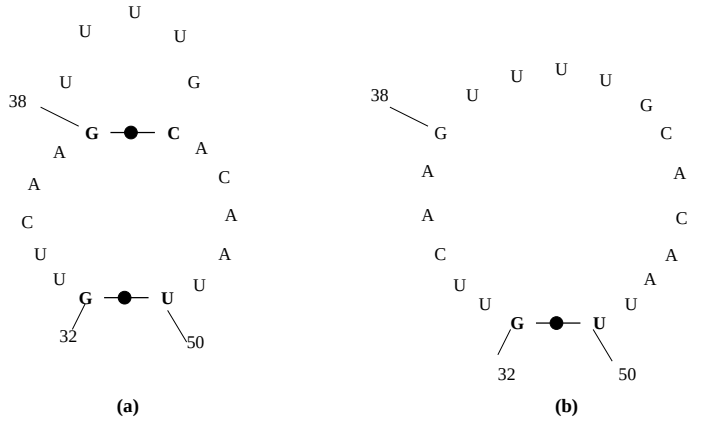


Figure 3.15: **Comparison of the predicted hairpin loop of the pre-microRNA dps-mir-6-3 with the corresponding area from the database.** (a) is the result of the database. (b) is the result of microRNAfold.

Fig. 3.15 displays the different hairpin loops between the database and our prediction approach. Let us see our scores. According to our strategy, the score for the hairpin loop of (b) (in Fig. 3.15) is 6.2 while the score for the hairpin loop and that 5x5 internal loop of (a) (in Fig. 3.15) is 6.66. So, we chose (b) as the structure of the hairpin loop based on our current scoring function. In order to improve our prediction algorithm we have to refine the scoring function [4][31].

3.5 Conclusion

The experimental results show that microRNAfold provides an alternative approach to the prediction of miRNA secondary structure, which has the following advantages: (1) it achieved higher Specificity, (2) it obtained higher Matthers coefficient ratio, and (3) receiver operating characteristic (ROC) plot showed that microRNAfold is the best among MFold, MC-FOLD, CONTRAFold, and microRNAfold. The predictive power of microRNAfold was evaluated in terms of input sequence lengths as well. Our model obtains the best performance when the length of the input sequence is average. In addition, we conducted experiments to assess the prediction performance with the different hairpin loop lengths. It seems that the model is ideal when the hairpin loop is small. If domain knowledge could be incorporated into our model it would improve the prediction a lot. Knowledge of secondary structure will provide enough structural constraints to the building of three-dimensional structure. In the future, we can consider formulating a hybrid statistics/thermodynamic model, which could use the statistical frequencies as *a priori* for selecting competing thermodynamically favorable configurations.

4 Predicting the Secondary Structure of Polycistronic MicroRNAs

There are some differences between plant miRNAs and animal miRNAs. Animal miRNAs are often encoded within introns of protein genes [1, 56, 89, 102], while most plant miRNAs are encoded in intergenic loci. In plants, miRNAs are mainly generated from independent transcripts [1, 89]. On the other hand, about 40 to 50% miRNAs in human, zebrafish and mammals, are located within clusters and encoded in independent hairpins or in both arms of the same hairpin [3, 1]. To our knowledge, miRNA clusters in plants have not been analyzed in detail. However, recently, a few miRNA clusters in plants have been reported [23, 43]. It has been shown that clustered miRNAs can be either encoded in a single polycistronic transcriptional unit or independently transcribed [3, 7, 89].

Based on these observations, an efficient algorithm should be available to predict the structures of poly-cistronic miRNAs. Both the classes of prediction methods based on phylogenetic sequence comparison [31, 33, 105] and the classes of prediction methods based on thermodynamics [11, 88, 134], however, face the same challenge: predicting the secondary structure of a single RNA is usually very time-consuming. We extend our previous work [44] by using a parallel algorithm to predict the secondary structure of multiple microRNAs located on a single polycistronic transcript.

In this chapter, we propose a new master-slave algorithm to tackle this problem. Among the parallel architectures, the master-slave architecture is easy to implement [50]. So we choose the master-slave architecture. First, the master processor partitions the input sequence into subsequences. Then the master processor distributes the subsequences to the slave processors and the slave processors begin predicting the structure of miRNAs. Afterwards, the master processor receives the returned structures from these slave processors, and merges partial structures into a list of whole structures. Finally, the master processor sorts these structures according to

their scores and obtains the optimal solution.

The organization of the chapter is as follows: In Section 4.1, we introduce a parallel algorithm, analyze the time complexity for sequential approach and parallel approach, and derive the speedup by using the proposed parallel method. The experiments and the results are presented in Section 4.2. A brief discussion is given in Section 4.3, followed by some concluding remarks in Section 4.4.

4.1 Parallel prediction of secondary structure of poly-cistronic miRNAs

In this section, we first describe how to predict the structure of multiple microRNAs by using the single microRNA prediction method. Next, we present a master-slave parallel algorithm for predicting the structure of multiple microRNAs. Finally, we analyze the time complexity of both sequential way and parallel way and derive the theoretic speedup under the ideal conditions.

4.1.1 Predicting multiple microRNAs' structure by using single microRNA prediction method

When predicting a single microRNA secondary structure, the prediction program is executed only once because the input sequence contains a single microRNA primary structure. On the other hand, when predicting the structure containing multiple microRNAs, the situation becomes more complicated. The input sequence contains more than one microRNA primary structures and other nucleotides. So the first task is to partition the input sequence into subsequences, and each subsequence contains a single microRNA primary structure. Then the subsequences (subtasks) are predicted respectively. When every subtask has been done, multiple partial results are merged into a candidate structure. So the single prediction algorithm could be run multiple times. There are many such partitions. So the big issue to the practical implementation of predicting multiple microRNAs' structure is that the number of

partitions may be large. We propose an effective algorithm to solve this problem by using parallel processing techniques.

4.1.2 Parallel algorithm

The master processor will partition a task (it is a sequence in our study) into subtasks and distribute them to the slave processors. The slave processors compute the secondary structures based on the subtasks received. The optimization processes, such as merging substructures and sorting the candidate structures, are performed by the master processor. The schematic algorithm is given in Algorithm 4.1.1.

ALGORITHM **4.1.1.** *Parallel algorithm based on the master-slave architecture for predicting the structure of multiple microRNAs. [1]*

Master Processor:

1. partition the input sequence into subsequences;
2. distribute the subsequence into multiple slave processors;

3. ... wait ...

7. receive the results from multiple slave processors;
8. merge partial structures into a whole structure with minimal score;
9. sort the candidate structures based on their scores;
10. obtain a global optimal solution.

Slave processors:

4. receive task from master processor;
5. compute the local optimal structure with score based on the assigned task;

Global optimal algorithm based on bottom-up

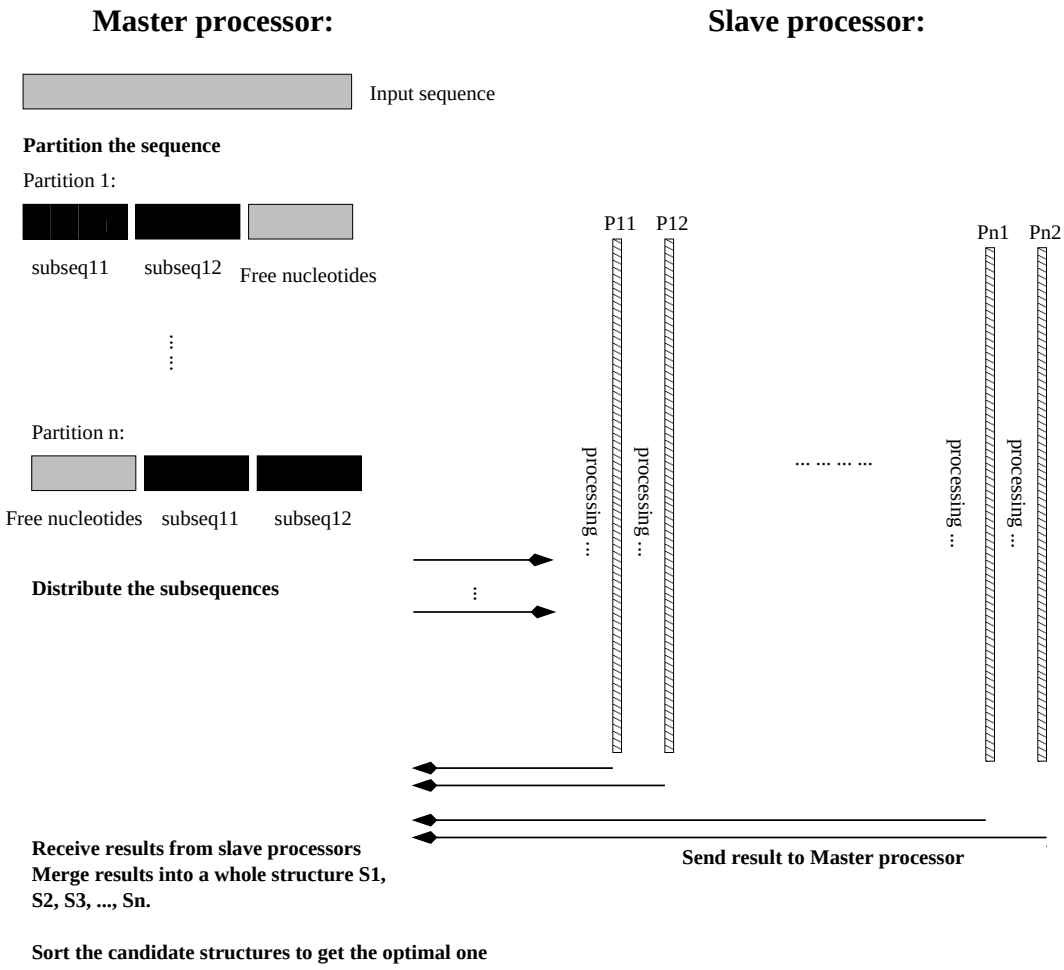


Figure 4.1: **Master processor and Slave processors.**

6. return the result to master processor;

It is easily seen that Steps 4, 5, and 6 are executed by the slave processors and must be finished before Step 7. The global optimal algorithm based on the bottom-up strategy, which is used by the slave processors, is briefly described in Chapter 3 (See [44] for detail).

Partitioning sequence

Basically, a sequence is a discrete structure used to represent an ordered list. The RNA sequence used in our study is referred to as a function from a subset of the

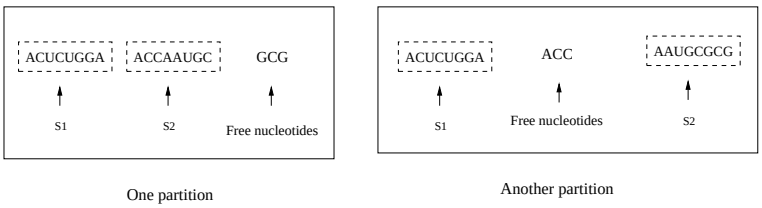


Figure 4.2: **An examples of two partitions.**

set of A, U, C, G. For example, sequence 1 can be denoted as AAUGC, sequence 2 AAGUC, and sequence 3 GCUAA. In Fig. 4.1, the first step is to partition the input sequence into multiple subsequences by the master processor. Suppose there is a sequence ACUCUGGAACCAAUGCGCG with the length of 19. We can partition it into two subsequences $s1$, and $s2$. And we consider the rest of nucleotides as free nucleotides. We partition the sequence with two constraints: The length and the number of subsequences are 8 and 2, respectively. Two partitions are shown in Fig. 4.2. All possible partitions will be examined by the algorithm.

Cost of partitioning sequence

Let n be the length of sequence, m be the number of subsequences, q be the subsequence length. The number of free elements will be $n - mq$.

We use NP to denote the number of partitions, which is:

$$
NP = \begin{cases}
(n - mq + 1) \\
+(n - mq + 0) \\
+(n - mq - 1) + ... + (n - mq - (n - mq - 1)) \\
= \sum_{i=-1}^{n-mq-1} (n - mq - i) \\
= (n - mq + 1)(n - mq) + 1 + 0 - 1 - 2 - 3.. \\
-(n - mq + 1) \\
= (n - mq)^2 + (n - mq) + 1 - (0 + 1 + 2 + 3 \\
+... + (n - mq + 1)) \\
= (n - mq)(n - mq + 3)/2 \\
\approx (n - mq)^2
\end{cases}
$$

4.1.3 Soundness of parallel strategy

With respect to the computational methods for RNA secondary structure prediction, generally there are two classes of algorithms, which are represented by combinatorial and recursive folding [32] and by dynamic programming [86]. The combinatorial and recursive methods are extremely time consuming and generally are limited to folding about 200 nucleotides. Its advantage is that it considers all the possible structures. The dynamic programming method is much better than the combinatorial and recursive algorithm in terms of time cost. More challenges will be encountered and the time cost is still a big issue when we deal with the prediction for multiple RNAs.

Due to the dramatic increase in available computing power over the last decade, prediction or simulation with parallel processing techniques has become a feasible way of overcoming the problem of expensive time cost. A number of studies have reported on the success of such techniques in bioinformatics and computational biology [19, 133]. Considering the problem itself, the proposed algorithm will certainly benefit from parallelization. In order to obtain the optimal secondary structure, we need to partition the input sequence. There are altogether $(n-mq)^2$ partitions, where n is the length of the input sequence, m is the number of microRNAs, and q is the length of a single microRNA. If sequential approach is applied the total prediction cost should be $(n-mq)^2$ times of the prediction time for a single microRNA. It is definitely very large if $n-mq$ is large. On the other hand, the problem itself is suitable for parallel computation, as each prediction is independent of the other predictions. Therefore, if we perform prediction by applying parallel processing strategy the computational cost could be largely reduced.

The parallel strategy we proposed combines the advantage of combinatorial and recursive method and the advantage of dynamic method. It considers all the possible partitions of sequence. The algorithm that the slave processor uses is presented in

Chapter 3.

4.1.4 Time complexity analysis

We first analyze the time complexities of both sequential and parallel algorithms. The speedup of the parallel method is then derived. We define some related notations as follows:

n : the length of input sequence;

m : the number of microRNAs;

q : the length of a single microRNA;

NP : the number of partitions;

p : the number of slave processors;

$n - m * q$: the number of free nucleotides in one partition;

t_{single}^{ave} : the average execution time of a single microRNA prediction;

t_{comm}^{ave} : the average communication time between the master processor and a slave processor; Communication is needed when the master processor distributes subsequences and receives results from the slave processors.

$t_{partition}$: the time for the master processor to partition sequence;

t_{sort} : the time for the master processor to sort candidate structures;

T_s^{ave} : the average time complexity of the sequential algorithm;

T_p^{ave} : the average time complexity of the parallel algorithm;

S_{ave} : the average speedup.

With respect to the sequential way, the program partitions the input sequence into subsequences. Then the subsequences are distributed and computed. As we mentioned early, the total number of partitions is NP; In each partition, m microRNAs will be predicted. This is the most time consuming part in the whole program. When every subsequence is constructed into structures the program sorts them and gets the optimal structure. Based on this procedure, we derive the average time

complexity of the sequential algorithm.

The average time complexity of the sequential algorithm is:

$$\begin{aligned} T_s^{ave} &= t_{partition} + NP * m * t_{single}^{ave} + t_{sort} \\ &= t_{partition} + (n - mq)(n - mq + 3)/2 * m * t_{single}^{ave} + t_{sort} \\ &\approx t_{partition} + (n - mq)^2 * m * t_{single}^{ave}/2 + t_{sort} \end{aligned}$$

The parallel algorithm can distribute the subsequence tasks to the slave processors. Therefore, The whole prediction time could be significantly reduced. However, the parallel algorithm needs additional communication time $NP * m * t_{comm}^{ave}$ between the master processor and the slave processors. We derive the average time complexity of the parallel algorithm.

$$\begin{aligned} T_p^{ave} &= t_{partition} + \frac{NP * m * t_{single}^{ave}}{p} + NP * m * t_{comm}^{ave} + t_{sort} \\ &\approx t_{partition} + (n - mq)^2 * m * t_{single}^{ave}/2p \\ &+ (n - mq)^2 * m * t_{comm}^{ave}/2 + t_{sort} \end{aligned}$$

The average speedup is:

S_{ave}

$$= \frac{T_s^{ave}}{T_p^{ave}}$$

$$\approx \frac{t_{partition} + (n - mq)^2 m t_{single}^{ave}/2 + t_{sort}}{t_{partition} + (n - mq)^2 m t_{single}^{ave}/2p + (n - mq)^2 m t_{comm}^{ave}/2 + t_{sort}}$$

In this study, the average time of partitioning the sequence and the sorting time are much smaller than the average time of single microRNA prediction. In addition, since the master processor only distributes the subsequences to the slave processors and receives the returned structures from the slave processors, the communication time is

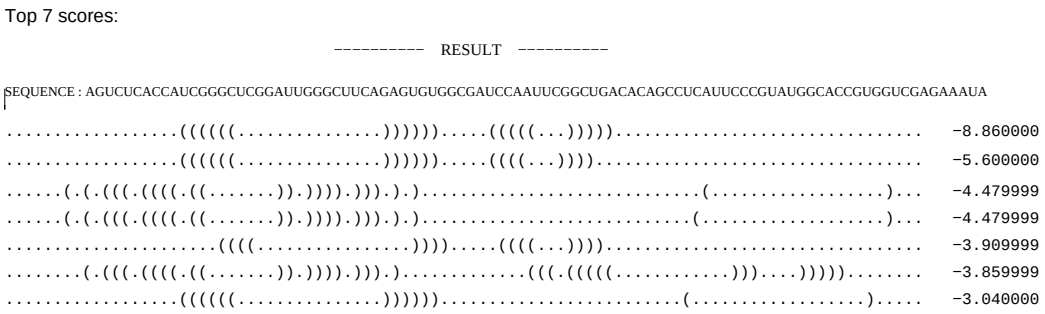


Figure 4.3: **Prediction of synthetic data.**

relatively small. So, the speedup can be simplified as follows:

$$\begin{aligned} S_{ave} &\approx \frac{(n-mq)^2 * m * t^{ave}_{single}/2}{(n-mq)^2 * m * t^{ave}_{single}/2p + (n-mq)^2 * m * t^{ave}_{comm}/2} \\ &= \frac{t^{ave}_{single}}{t^{ave}_{single}/p + t^{ave}_{comm}} \\ &= p[\frac{1}{1 + p\frac{t^{ave}_{comm}}{t^{ave}_{single}}}] \end{aligned}$$

When $\frac{t^{ave}_{comm}}{t^{ave}_{single}}$ is small enough (t^{ave}_{comm} is much smaller than t^{ave}_{single}), S_{ave} is near p. At this moment, the speedup is almost linear.

4.2 Experiments and Results

We conducted some experiments to verify the effectiveness of our parallel algorithm. First, we used some synthetic dataset. It is a sequence of 95 nucleotides. Two miRNAs with the length of 45. Second, we used a sequence of 183 nucleotides which include two endogenous poly-cistronic miRNAs osa-MIRNA395n and osa-MIRNA395o. Third, we used a sequence of 328 nucleotides which include two artificial miRNAs that were validated to be expressed to target the Arabidopsis AGO2 and AGO3 gene transcripts. Finally, we performed some experiments to test the performance of our approach with respect to speedup. Fig. 4.3 shows the first 7 structures with the top score in our first set of experiments.

4.2.1 Synthetic dataset

Fig. 4.4 shows the optimal structure with the minimal score -8.86. Another structure in Fig. 4.5 has one shorter microRNA which starts with the 51st nucleotide and ends with the 61st nucleotide. This structure is less stable than the optimal one just because its second microRNA is less stable than the corresponding one in the optimal structure.

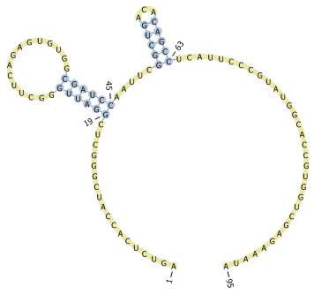


Figure 4.4: The optimal structure with the score -8.86.

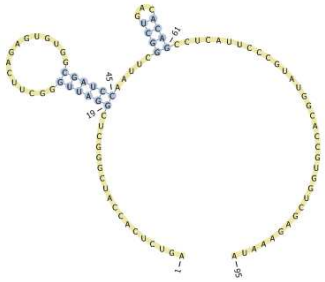


Figure 4.5: Another structure with the score -5.6.

The third structure and the fourth structure are shown in Fig. 4.6 and Fig. 4.7 respectively. Both of them have the same microRNA stem-loop structure (7-43 nucleotides). However, they are a little different in the second microRNA structure. We ranked them randomly.

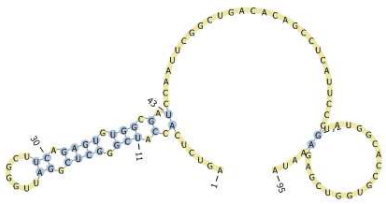


Figure 4.6: The third structure with the score -4.479.

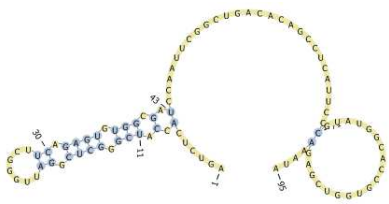


Figure 4.7: The fourth structure with the score -4.479.

4.2.2 Real world dataset

We performed an experiment to predict the secondary structure of putative polycistronic clustered pre-miRNAs (osa-MIRNA395n and osa-MIRNA395o). 990 struc-

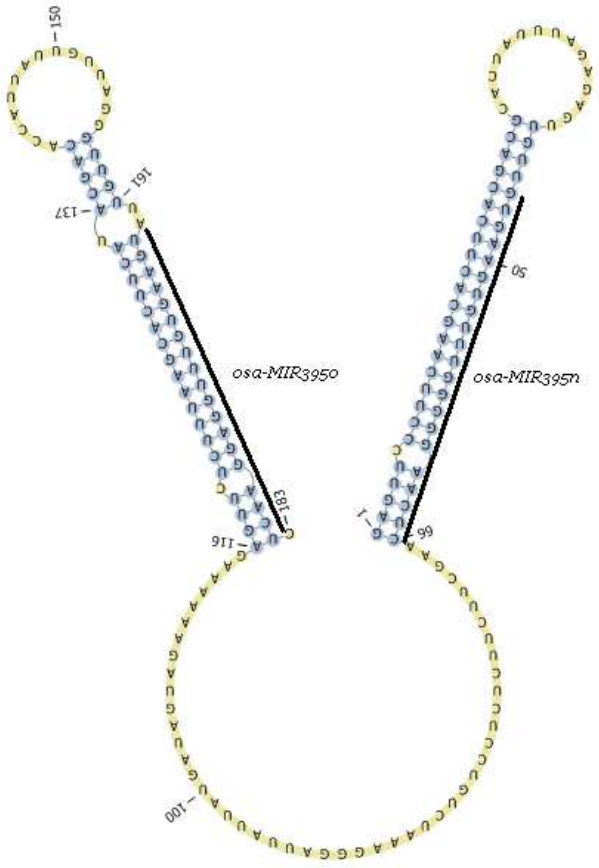


Figure 4.8: **Representative RNA secondary structure of polycistronic clustered miRNAs' precursors.** Its score is -41.50, osa-MIRNA395n is from 46 to 66, and osa-MIRNA395o is from 163 to 182 .

tures with scores were obtained. The optimal structure was shown in Fig. 4.8 [43].

AGO2 and AGO3 are two important proteins that constitute small RNA pathways in Arabidopsis. To study their functions, two artificial miRNAs (amiRNAs) can be designed to target them for silencing. We conducted some experiments to evaluate the performance of our algorithm in prediction of the secondary structures of the two amiRNAs in a transcript. 1225 structures with scores were obtained. The algorithm obtained the same results when different number of slave processors were used. Some of the obtained structures were listed as follows. MicroRNA A in Fig. 4.9 is AGO2 amiRNA. MicroRNA B consisted of a big loop and a short stem.

Compared to the microRNA B in Fig. 4.9, the microRNA B in Fig. 4.10 is rel-

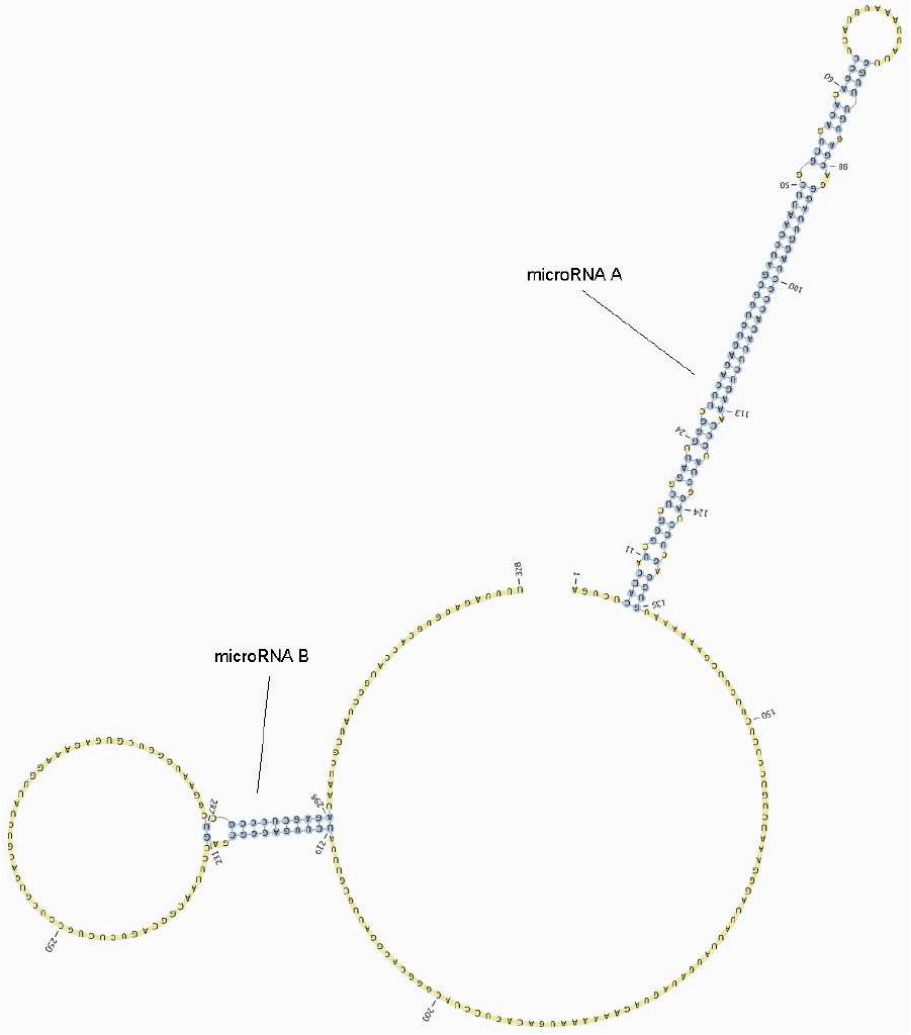


Figure 4.9: **Prediction of AGO2 and AGO3 amiRNAs.** Its score is -66.44, microRNA A (from 6 to 135) is AGO2 amiRNA, and microRNA B (219 - 298) .

atively more stable than that in Fig. 4.9 because it has a smaller loop and a longer stem which is consisted of more canonical bases.

We focus on the microRNA B (because the microRNA A is the same as that in other cases) in Fig. 4.11 as well. Compared to the two previous examples, this microRNA has a longer stem and a smaller loop. This structure is relatively stable.

The optimal structure is shown in Fig.4.12. It is clear that the miRNA A (i.e., AGO2 amiRNA) structure is the same as the ones in the previous three figures. In addition, the program successfully predicted the secondary structure of AGO3

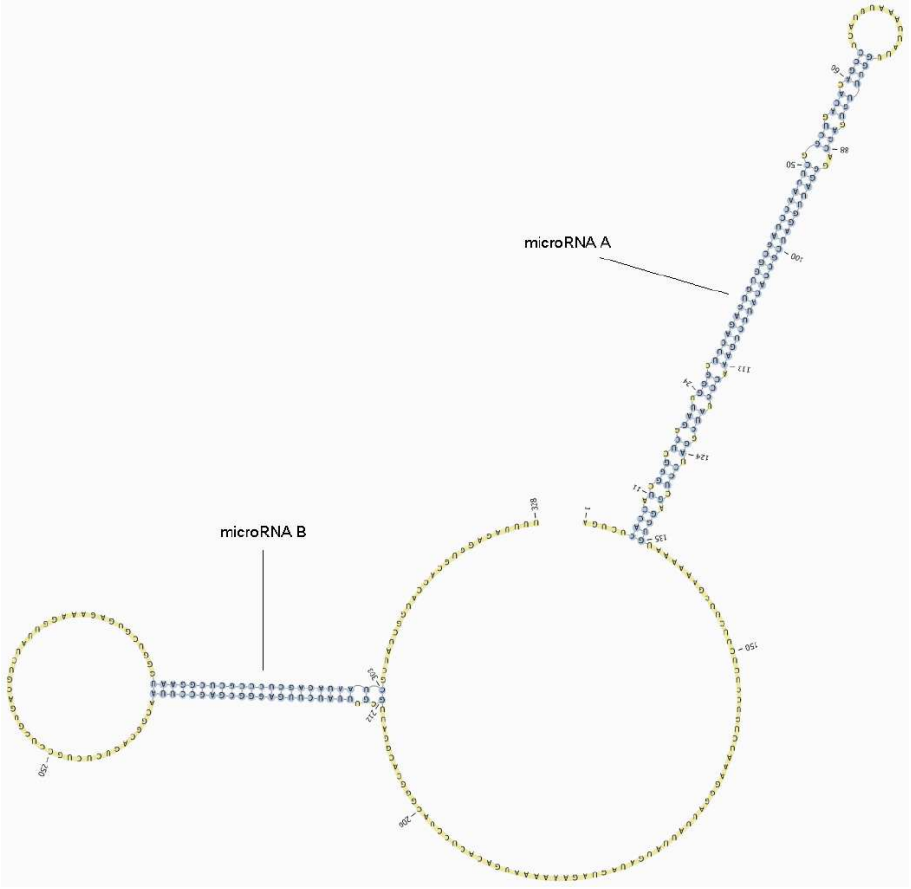


Figure 4.10: **Prediction of AGO2 and AGO3 amiRNAs.** Its score is -82.78, microRNA A (from 6 to 135) is AGO2 amiRNA, and microRNA B (212 - 303) .

amiRNA.

4.2.3 Speedup trend

Experiments were performed to show the performance with respect to speedup of the proposed approach. They were simulated in C programming language on a supercomputer. The supercomputer is an IBM HS21 blade cluster, which has 340 blades with 4 processor cores per blade. It uses an intel processor (3.0 GHz, 8 GB/node RAM) running a linux OS (2.6.5-7.244-smp). Table 4.1 shows the different time cost with different number of slave processors for predicting the structure of AGO2 and AGO3 amiRNAs. Fig. 4.14 displays the speedup when running simulations in parallel approaches. Because of the overhead associated with running a simulation in parallel,

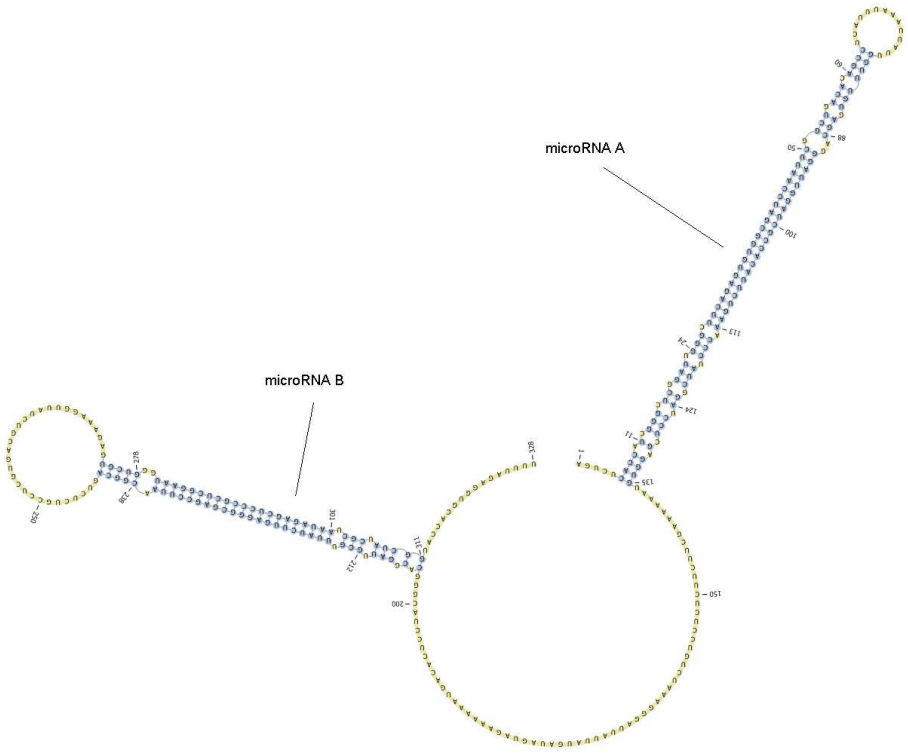


Figure 4.11: **Prediction of AGO2 and AGO3 amiRNAs.** Its score is -101.52, microRNA A (from 6 to 135) is AGO2 amiRNA, and microRNA B (204 - 311) .

the actual values (shown in hexagons) are different from the theoretic values (shown in circles).

Fig. 4.13 shows the different efficiencies with the number of slave processors. When the number of slave processors is 1 the speedup is 1.00 and the efficiency is $1.00/2 = 50\%$. Then with the increase of number of slave processors, the efficiency increases as well. When the number of slave processors is 4, the efficiency reaches 76%, which is the highest. However, after that, the efficiency decreases with the number of slave processors increasing.

The results (Fig. 4.14) demonstrate clear benefits from running simulations in parallel. The program obtained the largest speedup when the number of slave processors was 128. The trend is correct. Moreover, it can be easily observed that the speedup increased almost linearly along with the number of slave processors when the number of slave processors is less than 32. The speedup increases slowly when the number of

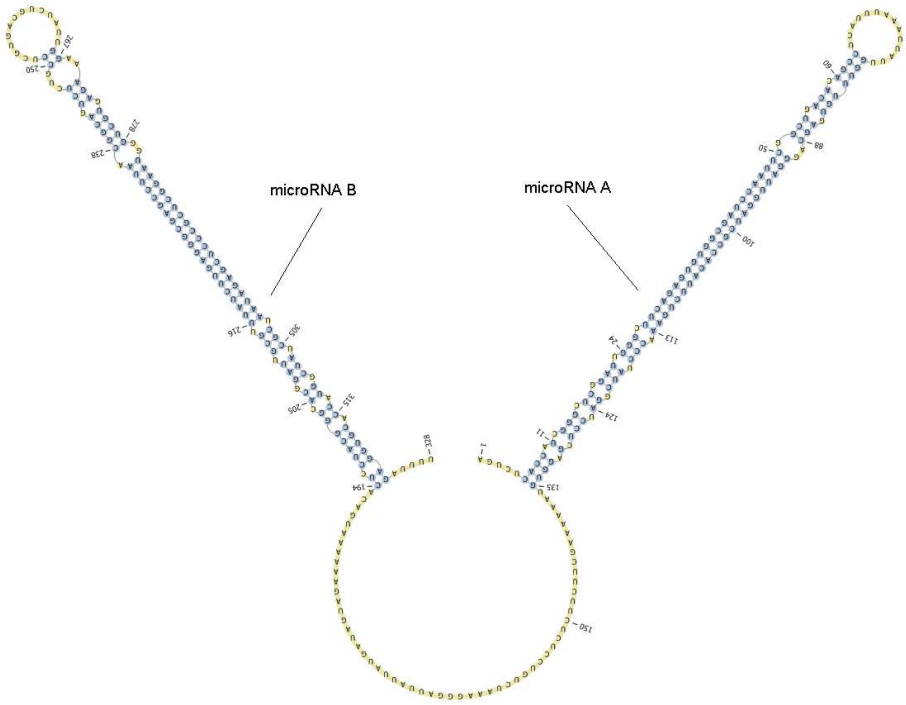


Figure 4.12: **Prediction of AGO2 and AGO3 amiRNAs.** Its score is -124.18, microRNA A (from 6 to 135) is AGO2 amiRNA, and microRNA B (194 - 324) is AGO3 amiRNA.

slave processors is greater than 32.

4.3 Discussion

4.3.1 When to get benefit from parallel computing?

Since the number of partitions is $(n-mq)(n-mq+3)/2$, it is obvious that the number of partitions is directly related to the number of free nucleotides (see the partitioning sequence cost). Apart from the number of free nucleotides, the overall prediction time is much influenced by the length of a single microRNA m. That means the prediction time will be much longer if the single microRNA sequence is long. So the length of a single microRNA m and the number of free nucleotides $n - mq$ are the main factors that affect our judgement: use parallel approach or sequential approach?

In general, if the subsequence is short (m is moderately small) and the number of free nucleotides $n - mq$ is not very large, the benefit of using the parallel algorithm

Table 4.1: The time cost with different number of slave processors (in minutes).

Num of slave processors	Total
1	137.5
2	72.5
4	36.8
8	21.3
16	12.13
32	8.9
64	8.7
128	8.3

to predict the structure of RNAs is relatively small.

4.3.2 Linear speedup

Many factors influence the effect of parallel computing on speedup. In general, the communication overhead between the slave processors and the master processor can be omitted because it is very small compared to computing time used by a single slave processor. The big issue here is the load balancing. Speedup is generally limited by the speed of the slowest node. So an important consideration is to ensure that each node performs approximately the same amount of work, i.e., the system is load balanced. In our study, it is almost impossible to guarantee that each node uses the similar or same amount of time. Let us do some theoretic analysis in the first place. The average speedup can be written as follows under the ideal conditions.

$$Speedup \approx \frac{t_{partition} + (n - mq)^2 * m * t_{single}^{ave}/2 + t_{sort}}{t_{partition} + (n - mq)^2 * m * t_{single}^{ave}/2p + t_{sort}}$$

To be simple, we partition the computational time as two parts: T_{serial} and $T_{parallel}$. So, the Speedup can be rewritten approximately as

$$\frac{T_{serial} + T_{parallel}}{T_{serial} + T_{parallel}/p}$$

Assume that the program uses $\mu/128$ seconds on the serial part and μ seconds on

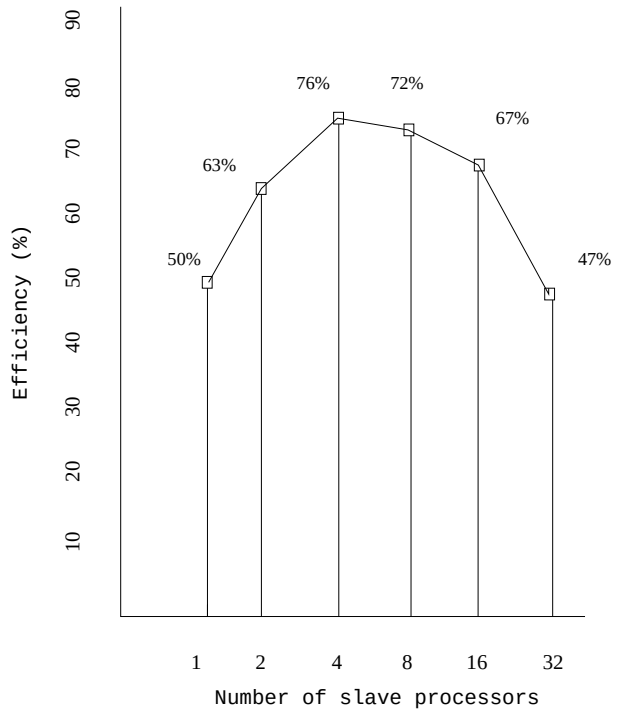


Figure 4.13: **The experimental results for efficiency.**

the parallel part. So when $p = 1$, the speedup is

$$\frac{\mu/128 + \mu}{\mu/128 + \mu} = 1$$

We get a series of theoretical speedups when p is different (see Fig. 4.14). We can see that the speedup line (dotted line) becomes flatter when p becomes bigger (bigger than 32 in our case). Actual experimental results also indicate this trend (see Fig. 4.14).

4.4 Conclusion

The experimental results show that our algorithm is able to produce the optimal secondary structure of poly-cistronic microRNAs. When there are many possible partitions of sequence it may be beneficial to use parallel algorithm instead of sequential algorithm. The experimental results show that sometimes multiple structures with the same score could be obtained. In the future, we may apply parallel algorithm to the endogenous poly-cistronic miRNAs from plants and animals for their secondary

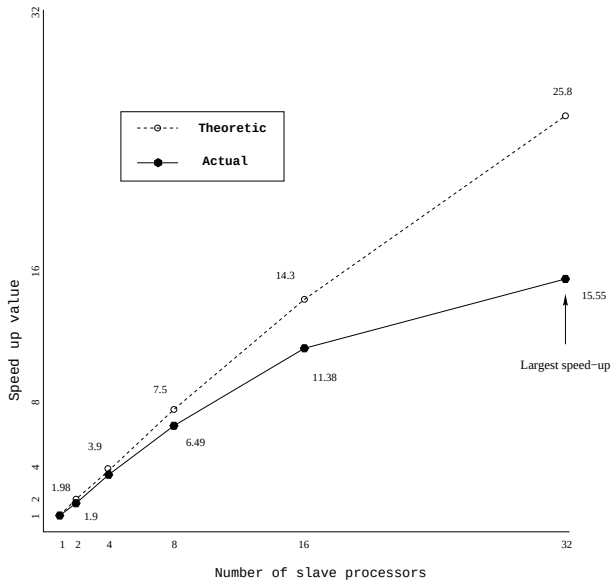


Figure 4.14: **The experimental results for speedup.** Theoretic values are represented in circles, and the actual values are denoted in hexagons.

structure prediction and validation. The trend of speedups of our parallel algorithm matches that of theoretical speedups. Knowledge of secondary structure will provide more structural constraints to the building of 3D structure. We can consider building a 3D structure model based on the secondary structure prediction in the future.

5 Classification of real and pseudo human pre-microRNAs based on structure's characteristics with SVM

5.1 Introduction

According to the current understanding, miRNAs are initially expressed as part of transcripts termed primary miRNAs (pri-miRNAs) [67]. The long primary miRNA is processed into $60\tilde{7}0$ nt miRNA precursor by nuclear RNase III Drosha [67, 68]. The pre-miRNA then is cleaved into $\tilde{2}2$ nt deplexes [7]. The important characteristics of pre-miRNAs is the stem-loop hairpin structure. The hairpin structure of pre-miRNA acts as not only the structure motif for Exportin-5 in nuclear-cytoplasm transportation, but also a substrate for Dicer enzyme [132]. It indicates that the secondary structures are very important in the miRNA biogenesis processing [130].

Computational methods are becoming more and more important due to the fact that it is difficult to systematically detect miRNAs from a genome by experimental approaches. Many computational methods used comparative genomics information to identify miRNAs [63, 64]. The general idea is to use comparative genomics to filter most of hairpins that are not conserved in related species [130].

In order to study the important features that can be used to differentiate the real pre-miRNAs from other hairpin sequences with similar stem-loops, we want to carry out some experiments to classify the real pre-miRNAs and pseudo pre-miRNAs. To our best knowledge, some other groups [130] performed research in this study and reported some good results. But they used a relatively large feature space and the data structure depends on the sequence data. Therefore, we want to use the characteristics from the secondary structures of pre-miRNAs as the features to do the classification with support vector machine (SVM).

This chapter is organized as follows: We briefly review the classification algorithms used in our prediction system in Section 5.2. The proposed prediction method is

described in Section 5.3. The computational experiments are carried out and the results are discussed in Section 5.4. A discussion on the results is introduced in Section 5.5. We sum up this chapter in Section 5.6.

5.2 Classification

In this section, we briefly review the classification algorithms we use in the system.

5.2.1 SVM Classification

The SVM (Support Vector Machine) is an effective classification method based on a structural risk minimization theory [119]. It has been successfully applied to many applications like face identification, text categorization, bioinformatics, etc [13, 14, 71].

In SVM classification the goal is to find a hyperplane that separates the examples with maximum margin. Given l examples (x_1, y_1), ..., (x_l, y_l), with $x_i \in R^n$ and $y_i \in \{-1, 1\}$ for all i, SVM classification can be stated as a quadratic programming problem:

$$\text{minimize } \frac{1}{2} \parallel w \parallel^2 + C \sum_{i=1}^{l} \xi_i$$

$$\text{subject to } \begin{cases} y_i(< w, x_i > + b) \geq 1 - \xi_i \\ \xi_i \geq 0 \\ C > 0 \end{cases}$$

where C is a user-selected regularization parameter, w is the weight vector, and ξ_i is a slack variable accounting for errors. After solving it, we can get the following decision function:

$$f(x) = \sum_{i=1}^{l} \alpha_i y_i < x_i, x > + b. \tag{5.1}$$

where $0 \leq \alpha_i \leq C$.

For the nonlinear case, we apply a kernel function, $K(x, x_i) =< \Phi(x), \Phi(x_i) >$, which maps the input space into some reproduced kernel feature space. Then Equa-

tion (5.1) can be rewritten as:

$$f(x) = \sum_{i=1}^{l} \alpha_i y_i K(x, x_i) + b. \tag{5.2}$$

The above methods only work for classifying two classes. If the matrix features are to be classified into more than two classes, we have to use one of the multi-class classification methods [51, 96, 121]. Two of the commonly used multi-class methods are "one-against-one" and "one-against-all". Suppose there are n classes, "one-against-all" method constructs n classifiers. Classifier i divides the data into the class belonging to class i and those not belonging to class i. The "one-against-one" method constructs classifiers for each of the class pairs, thus totally constructing $n(n-1)/2$ classifiers. Hsu *et al.* showed in [51] that "one-against-one" is more suitable for practical use. However, a recent publication [100] argued that "one-against-all" is as accurate as any other approaches. Since using "one-against-all" can save more training time, we adopt the "one-against-all" method in this chapter.

5.3 Prediction Method

Before we introduce our prediction method we would like to introduce aother method proposed by aother group.

5.3.1 Classification based on local structure-sequence features

In 2005, Xue *et al.* proposed a method to classify the real human pre-miRNAs and pseudo pre-miRNAs. They proposed a set of features that combine the local contiguous structures with sequence information to characterize the hairpin structure [130]. The features focus on the information of every 3 adjacent nucleotides. There are 8 possible structure compositions: (((, ((., (.., (.(, .((, .(., ..(, and .... There are 32 (4*8) possible structure-sequence combinations, which they denoted as U(((, A((., etc. Fig. 5.1 shows how triplet method extracts the features of pre-miRNAs.

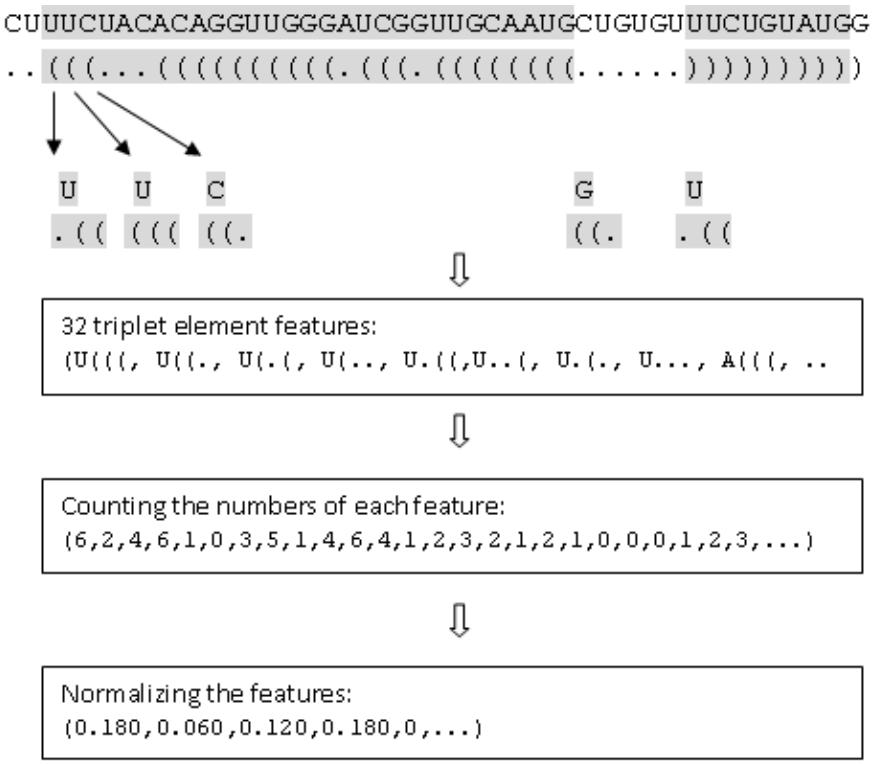


Figure 5.1: Triplet Method.

5.3.2 Classification based on conserved characteristics

In order to make use of the conserved characteristics of human microRNAs, we will perform the classification of real and pseudo human microRNA precursors. We will take the following steps:

Test all the known microRNAs in human

In this step, test all the structures in human except for the structures that represent the similar RNA sequence or predicted energy score is higher than a preset threshold value.

Extract features from each subject

Define some features according to the conserved characteristics. For example, Pre-microRNAs names, Length of pre-microRNAs, Number of base pairs, Number of mis-matched, Size of loop, Degree of base, Ratio of loop to whole sequence, Distance

from the 5 strand to the loop, energy score. We then form a matrix.

Normalize the feature matrix

If there are multiple features in a sample and the scales of some features are different, normalization process is needed. Normalization processing allows underlying characteristics of the data sets to be compared: this allows data on different scales to be compared, by bringing them to a common scale. There are different formula to get the normalized data. Our formula is listed as follows.

$$sum = \sqrt{\sum_{i=1}^{n} x_i^2}$$

$$x_i = \frac{x_i}{sum}$$

Train the training set and build the learning model

Support Vector Machine (SVMs) model will be used in this study. SVM is a relatively new learning process influenced highly by advances in statistical learning theory [120].

Predict the testing set

The LibSVM package [18] will be used. To obtain SVM classifier with optimal performance, the penalty parameter C and RBF kernel parameter will be tuned based on the training set.

5.4 Experiments and Results

5.4.1 Human miRNA precusor and pseudo miRNA datasets

Sets of human pre-miRNAs and pseudo-miRNA hairpins are collected to train SVMs and to evaluate the classification performance.

Human pre-miRNAs

The sequences of human pre-miRNAs are downloaded from the mirBase database (http://www.mirbase.org/cgi-bin/mirna_summary.pl?org=hsa), which contains 510

reported pre-miRNA entries from Homo Sapiens. Only the pre-miRNAs whose secondary structures do not contain multiple loops, the number of bases are greater than 15, their scores are less than -14.50, and the lengths are between 65 and 110, 281 pre-miRNAs are considered.

Human pseudo pre-miRNAs

The dataset of human pseudo pre-miRNA hairpins are built. They are sequence segments that have similar stem-loop structures as real pre-miRNAs. And the dataset is collected from protein coding regions. The protein coding sequences (CDs) of human RefSeq genes with no known alternative splice events are collected. The CDs are extracted according to the UCSC refGene annotation tables [53, 97]. We join the CDs together and extract non-overlapping segments from them. And we make the length distribution of the chosen segments identical with that of human real pre-miRNAs. The secondary structures of the extracted segments are predicted using microRNAFold [44]. The criteria for selecting the pseudo-microRNAs are: the minimum number of base pairs on the stem is 15, their score is less than -14.5, there is no multiple loops, the lengths range from 65 to 110. As far as we know that all reported micorRNAs are located in non-coding regions or intergenic regions, we take the hairpins collected from CDs as correct examples of pseudo pre-miRNAs.

5.4.2 Training and test sets for classification experiments

For the classification experiments, one training set and one test set are built using the data set explained above. The training set includes 224 real human pre-miRNAs (positive samples) and 144 pseudo human pre-miRNAs (negative samples) randomly selected from 281 real human pre-miRNAs and 180 pseudo human pre-miRNAs. The test set consists of the remaining real human pre-miRNAs and pseudo human pre-miRNAs.

5.4.3 Extract features from each subject

Some important features, which include Pre-microRNAs names, Length of pre-microRNAs, Number of base pairs, Number of mis-matched, Size of loop, Degree of base, Ratio of loop to whole sequence, Distance from the 5 strand to the loop, and energy-score, are extracted from each subject. Table 5.1 lists the part of the feature matrix of real human pre-miRNA, and Table 5.2 lists the part of the feature matrix of pseudo human pre-miRNA.

Table 5.1: Extracted features from real human pre-miRNAs.

Pre-Name	Pre-len	BP #	Mis #	Lp SZ	Deg-BP	Dist	Ratio L/Seq	Score
hsa-let-7b	83	26	31	30	0.627	26	0.361	-26.49
hsa-let-7c	84	27	30	17	0.643	34	0.202	-17.84
hsa-let-7d	87	27	33	26	0.621	31	0.299	-21.17
hsa-let-7e	79	24	31	23	0.608	28	0.291	-19.22
hsa-let-7f-2	83	28	27	26	0.675	29	0.337	-21.7
hsa-let-7i	84	30	24	9	0.714	35	0.310	-28.11
hsa-mir-7-1	110	40	30	16	0.727	47	0.255	-23.76
Hsa-mir-9-2	87	34	19	10	0.781	39	0.115	-20.18
hsa-mir-19a	82	30	22	12	0.732	34	0.146	-18.16
hsa-mir-20a	71	27	17	11	0.761	30	0.155	-22.84
Hsa-mir-21	72	27	18	13	0.750	30	0.181	-20.53
hsa-mir-23a	73	27	19	10	0.740	31	0.137	-17.69
Hsa-mir-24-1	68	23	22	13	0.676	28	0.191	-21.32
hsa-mir-26a-1	77	28	11	7	0.727	32	0.091	-20.8
Hsa-mir-28	86	31	24	16	0.721	35	0.186	-39.6
hsa-mir-29b-1	81	31	19	13	0.765	35	0.160	-21.11
hsa-mis-30e	92	34	24	9	0.739	41	0.098	-18.14
Hsa-mir-31	71	27	17	8	0.761	31	0.113	-31.32
Hsa-mir-32	70	27	16	16	0.771	28	0.229	-17.3

We performed statistical analysis on the feature matrices of each category. The mean, median, and standard deviation values are listed in Fig 7.4.

Table 5.2: Extracted features from pseudo human pre-miRNAs.

Pre-Name	Pre-len	BP #	Mis #	Lp SZ	Deg-BP	Dist	Ratio L/Seq	Score
$NM_0$32291	68	17	34	6	0.500	32	0.088	-14.7
$NM_0$52998	69	17	35	11	0.493	37	0.159	-22.5
$NM_0$13943	65	17	31	4	0.523	23	0.062	-25.8
$NM_0$01145278	63	19	25	3	0.603	34	0.048	-20.0
$NM_0$01145277	72	25	22	3	0.694	34	0.042	-24.9
$NM_0$32785	67	15	37	6	0.448	28	0.090	-17.4
$NM_0$01080379	66	18	30	6	0.545	32	0.091	-28.1
$NM_0$18090	69	22	25	3	0.638	34	0.043	-22.9
$NM_0$01918	68	20	28	8	0.588	23	0.118	-22.4
$NM_0$03243	69	19	31	14	0.551	27	0.203	16.4
$NM_0$30806	72	19	34	5	0.528	32	0.069	-14.9
$NM_0$21222	73	17	39	8	0.466	26	0.110	-15.3
$NM_0$22457	89	22	45	6	0.494	48	0.067	-14.5
$NM_0$01001740	67	17	33	5	0.507	24	0.075	-18.8
$NM_1$73083	71	18	35	4	0.507	24	0.056	-22.2
$NM_0$12405	65	19	27	5	0.585	30	0.077	-25.0
$NM_0$22114	70	15	40	5	0.429	35	0.071	-14.5
$NM_0$17891	66	15	36	11	0.455	31	0.167	-15.1
$NM_0$15215	87	22	43	9	0.506	37	0.103	-24.8
$NM_0$01033581	64	16	32	7	0.500	40	0.109	-22.9

5.4.4 SVM classification

We test how accurate our SVM classifier is, compared with the other SVM classifer built with triplet model. SVM is chosen due to its good generalization [119]. We use 5-fold cross validation and choose the parameter C in SVM to be 10000 according to the results of the 5-fold cross validation.

Fig. 5.3 shows the accuracy comparison using a RBF kernel ($\gamma = 0.13$). The figure indicated that our method improved the accuracy by 5% in terms of Positive Correct and Negative Correct. Basically, that was statistically significant improvement.

Feature	*Average*		*Median*		*SD*	
	Real	*Pseudo*	*Real*	*Pseudo*	*Real*	*Pseudo*
Pre−miRNAs Length	87.80	78.70	87.00	77.00	10.6	11.13
Number of base pair	31.65	19.91	32.00	18.00	4.30	4.70
Number of mis−matched	24.50	38.90	24.00	39.00	7.40	7.87
Loop size	10.60	6.74	10.00	6.00	5.20	4.09
Degree of base pair	0.72	0.504	0.73	0.49	0.06	0.08
Distance from end of 5' miR to loop	38.23	34.68	38.00	34.00	6.62	8.03
Ratio of loop	0.124	0.086	9.118	0.07	0.06	0.05
Score	−27.33	−22.11	−24.80	−20.40	9.30	9.96

Figure 5.2: Statistics of each feature.

Approach	Samples	Accuracy (%)
Triplet	Positive Correct	93%
	Negative Correct	87%
Our Method	Positive Correct	98%
	Negative Correct	92%

Figure 5.3: SVM classification.

5.5 Discussion

We obtained better results on classification of real and pseudo human pre-miRNAs using our approach compared with other method, such as triplet-SVM. Through a careful observation and study, we found the reasons why our model outperforms triplet-SVM model. Firstly, our approach uses many fewer features than triplet-SVM model. In triplet-SVM model, all 32 features get involved in prediction, while in our model, only 8 features are included instead and the feature name is extracted only for statistical purpose. Triplet-SVM approach does not use feature selection algorithm to filter out the features that are considered less important because all of

them are equally important. On the other hand, our model uses only 8 features so it is not necessary to use feature selection. Our model using fewer features makes it more accurate. Secondly, our model is much better than triplet-SVM in terms of generalization since data set is from characteristics of structure itself. On the other hand, triplet-SVM uses sequence-structure as features. So it depends more on the sequence data than our model. In addition, this factor will have a negative impact on the accuracy of classification.

5.6 Conclusion

A major characteristics of pre-miRNAs is the hairpin structures. But there are so many similar hairpins that can be formed from segments in genomes. Therefore, an effective method that can distinguish the real pre-miRNAs from pseudo pre-miRNAs is important for identifying novel and specific miRNAs. Based on this reason, some conserved features are extracted and SVM classifier was built with these features to perform classification between real vs. pseudo pre-miRNAs. The experimental results showed that our method achieved 98% accuracy on positive correct and 92% accuracy on negative correct. Moreover, compared with triplet-SVM classifier, our method outperforms it, which uses sequence-structure as features in terms of accuracies. The other advantages that our method has include: our model uses fewer features (9 features) while triplet method used 32 features, and our model decreases the dependency on the sequence data because it just uses the characteristics from the secondary structures of pre-MiRNAs. Finally, our results indicated that there are some discriminative and conserved characteristics that separate the real pre-microRNAs from the pseudo ones effectively, such as the 8 features.

6 Improving prediction based on conserved microRNA characteristics in human

In this chapter we focus on improving the prediction based on the secondary structure characteristics itself. It shows that the conserved characteristics from the secondary structure can contribute positively to the prediction. So extracting and identifing these features are the first step, which is the most important step as well. Then we construct the knowledge base (KB) based on the rules from the characteristics combined with some other pre-microRNA secondary structure features. In order to make the KB robust, we incorporate fuzzy techniques when we construct it. Our aim in this chapter is to improve the prediction with knowledge base support whose rules are extracted from the secondary structure themselves.

This chapter is organized as follows: Section 6.1 introduces the research that had been performed by the other groups and the structure of our improved prediction system. The characteristics extracted from the human pre-microRNA secondary structures will be shown and explained in Section 6.2. The KB is constructed in Section 6.3. The experiments are carried out and the results are reported in Section 6.4. In Section 6.5, we discuss the issue of conflict set. Conclusion of this chapter is in Section 6.6.

6.1 Introduction

Computing the consensus structure common to several related RNA sequences and identifying conserved characteristics in RNA can drastically improve the prediction [5, 10, 94]. After the characteristics are extracted from the secondary structures of RNAs, corresponding patterns or rules could be dug out and used. To our best knowledge, some studies using conserved structure to improve prediction have been reported [9, 104, 130]. Even though they were trying to utilize the features from the secondary structures to enhance the prediction, we first time propose to use

the important features through knowledge base with fuzzy techniques to achieve our goal: improving the prediction accuracy. Here, We will focus on the pre-microRNAs in human.

Four steps required are described below.

(i) *Test all the pre-microRNAs in human:* In this step, we test all the structures (about seven hundred structures) in human except for the structures that represent the similar RNA sequence or its predicted score is higher than a preset threshold value.

(ii) *Obtain conserved characteristics:* After statistical analysis, we can get some very important conserved characteristics. Many other conserved characteristics should be extracted as well.

(iii) *Derive rules from the characteristics and further build a knowledge base:* Rules can be derived through these observed characteristics.

(iv) *Predict the testing set with the support of the knowledge base (KB)* Our prediction algorithm goes through the KB in order to try to find some helpful information. This information can be applied into the prediction so that the prediction time cost will be reduced. Testing set should be different from the data set from which characteristics are extracted. The performance of improved prediction algorithm will be measured by comparing the experimental results in this step (step iv) with the corresponding ones in step (i).

Our current method to solve this problem is to extract important features from the secondary structures and perform statistical analysis on the features. Based on these useful and conserved characteristics and other biological constraints regarding pre-miRNAs (human), we create an effective knowledge base. When we run the prediction program the KB provides support to speed up the prediction. The construction of effective KB poses a new challenge but it will make our algorithm more stable and efficient (see Section 6.3 for the detail).

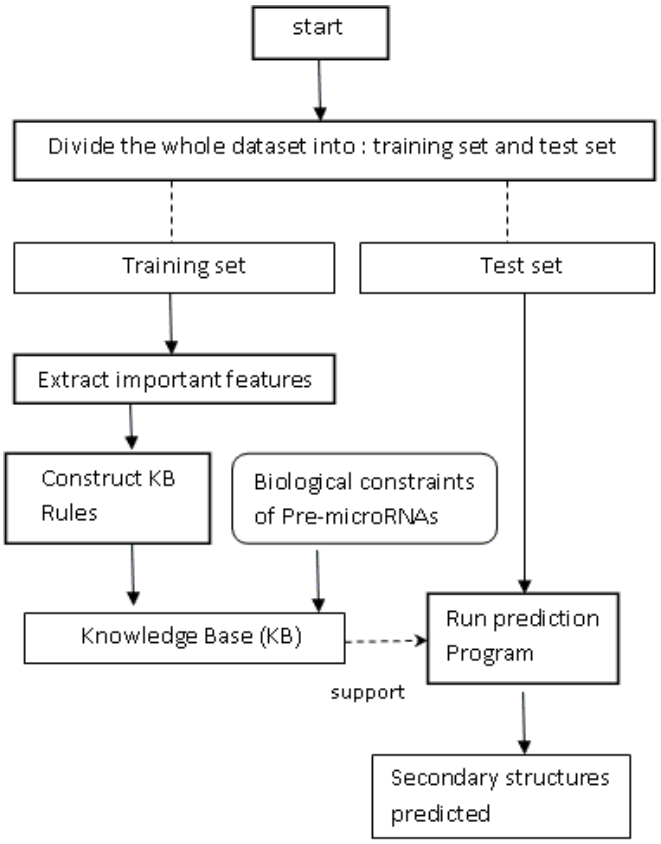


Figure 6.1: **Prediction based on KB support.**

6.2 Conserved characteristics

MicroRNAs are found to play an important role in regulating gene expression in plants and animals. They regulate gene expression by two ways: mRNA cleavage or translational repression. There are four steps for miRNAs to become mature. First, enzymatic processing of long non-coding hairpin transcripts (called primary miRNAs) yields pre-miRNA [30]. Second, pre-miRNAs are exported to the cytoplasm. Third, precursors are cut into double-stranded miRNA:miRNA* duplexes. Finally, the duplexes are unwounded, the miRNA strands are incorporated into RNA induced silencing complex (RISCs).

It is more likely that efficient biogenesis of miRNAs requires molecular signatures. For example, the enzymatic processing of pri-miRNAs may require certain character-

istic primary or secondary structure featurs from pri-miRNAs [104]. Similarly, the export efficiency may depend on the secondary structure features or primary features. Our goal is to find all the conserved structural characteristics that make miRNAs effective regulators *in vivo* and help identify the other pre-miRNAs. We attempted to use these characteristics to construct a Knowledge Base that supports effective prediction.

It has been shown that several characteristics are conserved in pre-miRNAs. First, the probabilities of unpaired base in the stem is conserved. Second, internal loops and bulges are more prevalent at specific positions. Finally, the double-stranded structure is conserved throughout the first 13 nt flanking the stem.

6.2.1 Probabilities of unpaired base

First, let us take a look at the probability distribution of mismatched bases. Fig. 6.2 indicated that the positions 2-9 are highly likely to be mismatched. This seed region is verified by many published reports [1, 9]. And nucleotide 1 is the most likely to be mismatched. Base-pairing at the first base in the precursor and at positions 10-12, corresponding to the middle area of the duplex, has a higher probability of being disturbed by the internal loops and bulges. This result matched the report by [104]. On the other hand, positions 13-21 have high probability of being base-pairing. And this result is supported by [104].

6.2.2 Relationship between the sequence length and loop size

Fig. 6.3 shows the relationship between the loop size and the sequence length. It is interesting to learn that almost all the loop sizes are around 10 and when the sequence length is between 65 and 69 the loop size increases along with sequence length increases. But after the sequence length is 70 the pattern is much like a saw tooth. Fig. 6.5 is more clear to demonstrate that all the means across every length segments are larger than 10, which matched the research result. The length segment

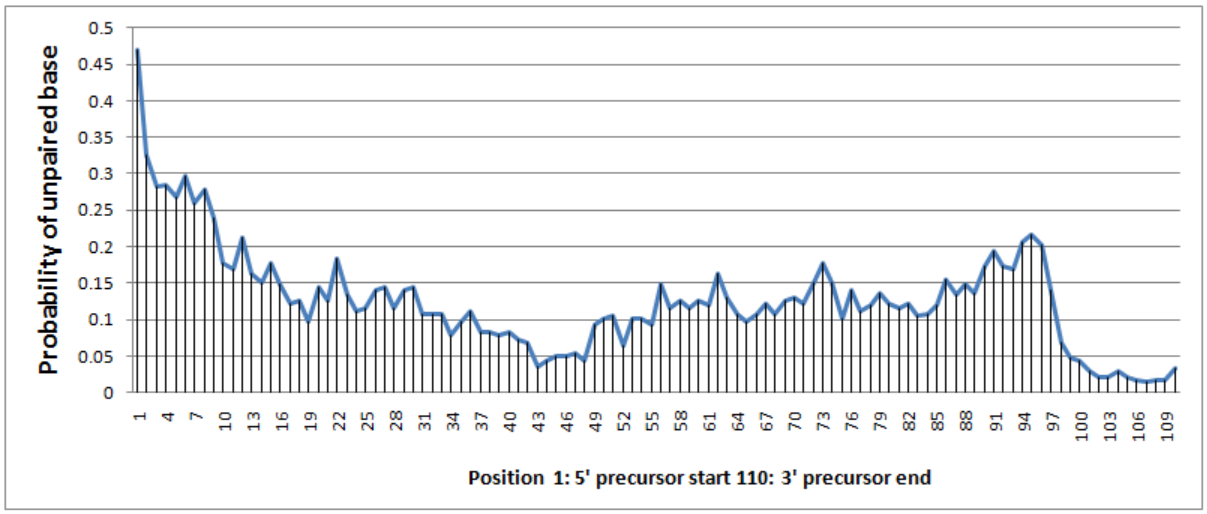


Figure 6.2: **Mismatch position.**

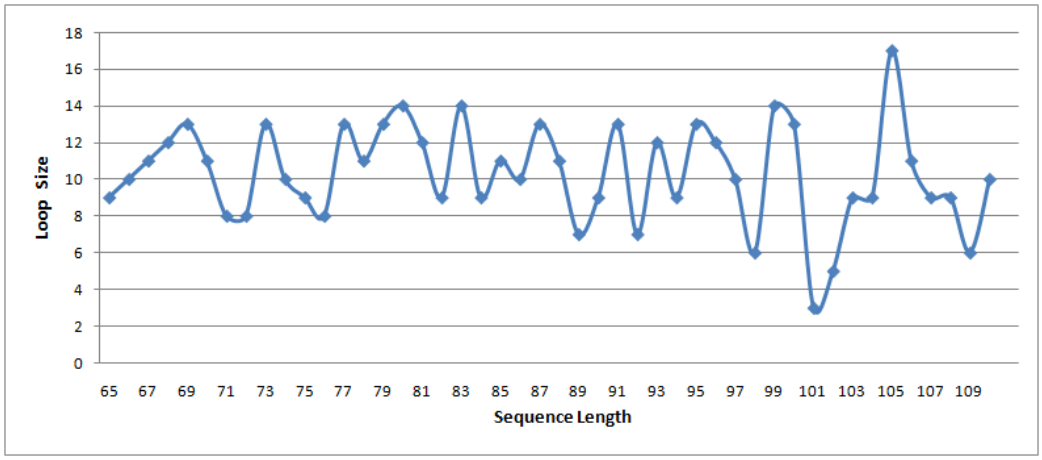


Figure 6.3: **Different sequence lengths and corresponding loop sizes.**

(71-80) has the highest mean in terms of loop size. With the bigger sequence length, the secondary structure is more likely to have a longer stem and so the loop is smaller.

6.2.3 Relationship between the sequence length and score

There are two high score areas (83-89) and (95-98), and their means are around -30.10 and -29.56 respectively. The area (78-80) is the low score area. In addition, when the sequence lengths are 65, 66, 70, 72, 102, or 106 their scores are as low as -20.00 or below that. Unfortunately, we could not apply this information in our knowledge base to improve our prediction.

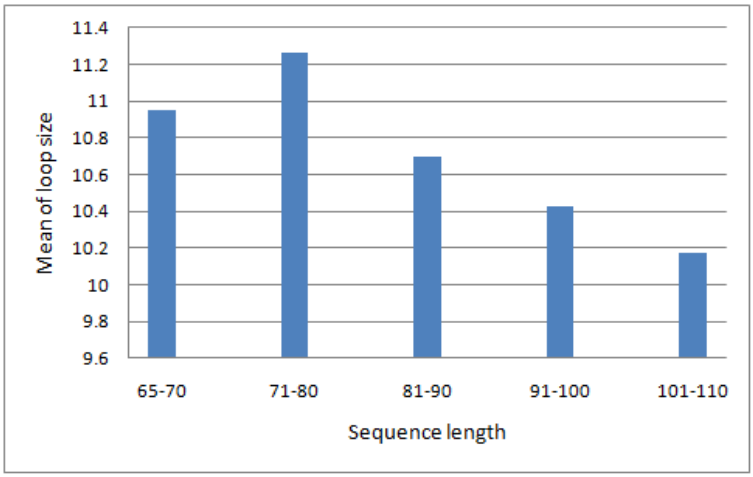


Figure 6.4: **Different sequence length ranges and corresponding loop sizes.**

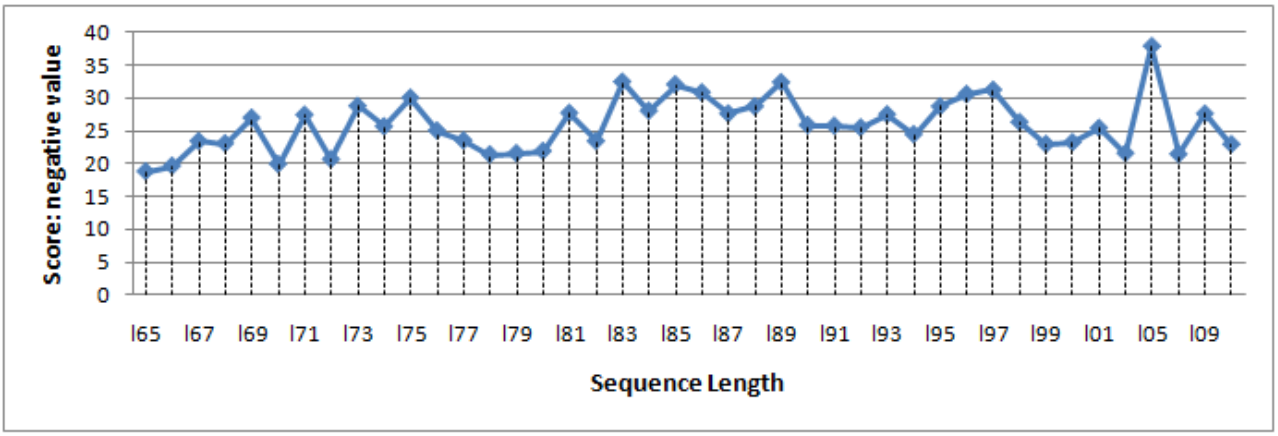


Figure 6.5: **Different sequence lengths and corresponding scores.**

6.3 Creation and building of Knowledge Base

6.3.1 The definition of Knowledge Base

A knowledge base (KB) is a special kind of database for knowledge management, providing the means for the computerized collection, organization, and retrieval of knowledge [60].

Knowledge bases are categorized into two major types: machine-readable knowledge bases and human-readable knowledge bases. Machine-readable knowledge bases store knowledge in a computer-readable form, usually for the purpose of having automated deductive reasoning applied to them. They contain a set of data, often in

the form of rules that describe the knowledge in a logically consistent manner [60].

Human-readable knowledge bases are designed to allow people to retrieve and use the knowledge they contain.

We want to build a small scale machine-readable knowledge base, which contains a set of data in the form of rules that are derived from conserved characteristics.

6.3.2 Difference between Knowledge Base and Database

(1) A database is an organized collection of data for one or more purposes, usually in digital form [29]. Database keeps structured related data. Knowledge base keeps knowledge. Data is extracted and displayed, but knowledge is learning and answering. Data is not information; information is not knowledge. Data is the collection of facts, figures and statistics related to an object [25]. Data can be processed to create useful information. Data is information that has been translated into a form that is more convenient to move or process [26]. Knowlege is information with guidance for action based upon insight and experience [20].

(2) A knowledge base is not a static collection of information, but a dynamic resource that may itself have the capacity to learn, as part of an artificial intelligence expert system [59]. Knowledge can be used to change the intelligence agent's status because of the learning process involved, but data cannot. Data-based systems only process data and don't output information. Knowledge base poses challenges.

6.3.3 Soundness of using KB instead of database

Using KB to support the prediction is due to the reasons as follows.

First of all, the very useful and conserved features can be extracted from the characteristics. It is easy and convenient to construct rules from these important features. Our prediction will be improved based on the support from KB in terms of CPU time and accuracy. Second of all, we generally use query to retrieve data from databases and then we can perform some statistical analysis on the data. In our

case, we don't need this kind of analysis. On the contrary, the ability to learn makes knowledge base extremely valuable to our prediction because prediction process will be time consuming especially when the input sequence is relatively long. Finally, as people understand how miRNAs regulate the gene expression more deeply, people's knowledge about the miRNAs' structure and function will also get updated. So, knowledge base fits our requirement more than database.

6.3.4 Creation of an effective Knowledge Base

Different knowledge representation languages and challenges of Knowledge Base

The key factors for knowledge based systems are knowledge acquisition, knowledge representation, and application of large bodies of knowledge to the problem domain. Basically, four bottlenecks exist in knowledge acquisition [122, 123, 126].

(1)Narrow bandwidth. The channels that exist to convert organizational knowledge from its source (either experts, documents, or transactions) are relatively narrow.

(2) Acquisition latency. The slow speed of acquisition frequently is accompanied by a delay between the time when knowledge (or the underlying data) is created and when the acquired knowledge becomes available to be shared.

(3) Knowledge inaccuracy. Experts make mistakes and so do data mining technologies (finding spurious relationships). Furthermore, maintenance can introduce inaccuracies or inconsistencies into previously correct knowledge bases.

(4) Maintenance trap. As the knowledge in the knowledge base grows, so does the requirement for maintenance. Furthermore, previous updates that were made with insufficient care and foresight (hacks) will accumulate and render future maintenance increasingly more difficult [65].

The major techniques used on knowledge representation include the Frame Formalism, Production Rule Systems, and Semantic Networks. A frame can be thought of a remembered framework which can be adapted to fit a given situation by changing

Figure 6.6: **A small example of using frame.**

the aspects of the frame as necessary [90].

Frames

Frames can be thought of as named lists of slots into which values can be placed. There are two types of frames: individual and generic frames. Individual frames represent single objects, whereas generic frames represent categories or classes of objects [118].

We could represent some knowledge about elephant in frames (Fig. 6.6) as follows:

A particular frame (such as elephant) has a number of attributes or slots such as color and size where these slots may be filled with particular values, such as grey. We have used a "*" to indicate those attributes that are only true of a typical member of the class, and not necessarily every member.

ProductionRules

One of the most popular approaches to knowledge representation is to use production rules. The production rules consist of a set of if-then rules, and a working memory. The working memory represents the facts that are currently believed to hold, while the if-then rules typically state that if certain conditions hold (e.g., cer-

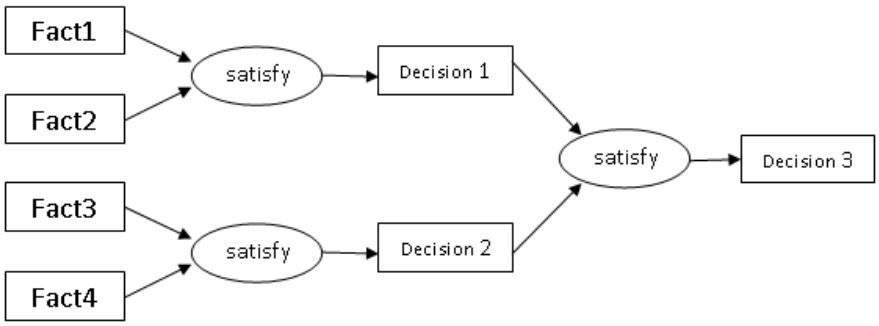


Figure 6.7: **One example of using production rule.**

tain facts are in the working memory), then some action should be taken (e.g., other facts should be added or deleted).

Some of the benefits of IF-THEN rules are that they are modular, each defining a relatively small and, at least in principle, independent piece of knowledge. New rules may be added and old ones deleted usually independently of other rules.

Fig. 6.7 shows an example of using production rule. If both Fact 1 and Fact 2 are satisfied we get decision 1. Similarly, if both Fact 3 and Fact 4 are satisfied we get decision 2. If decision 1 and decision 2 are satisfied, we get decision 3.

SemanticNetworks A semantic network is a directed graph consisting of vertices, which represent objects, individuals, or abstract classes; and edges, which represent semantic relations [118]. The most important relations between objects are *subclass* relations between classes and subclasses, and instance relations between particular objects and their parent class.

Construction of an effective Knowledge Base

Based on our problem requirements, we choose to use production rules to build the knowledge base.

We did not use regular if-then rules, instead, we used if-then rules with fuzzy strategy.

One of the rules is shown in Fig. 6.8: if all the three conditions are satisfied

```
RULE 1
if (sequence length is 65)  AND (fuzzy_rate is valid) (Note: fuzzy_rate is between 0.1 and 0.9)
                            AND (loop size is greater than 0)
  then
    action {
            compute the local length; (note: local_length = 9*(1+fuzzy_rate);)
            if (loop size <= local_length)
                allows the program to continue;
            else
                does not accept this loop size
    }
```

Figure 6.8: **Production rule example.**

the action is supposed to be performed, which calculate the acceptable loop size and judge whether the passed loop size is fine or not.

There are 42 production rules regarding the loop size. To go into them in detail, go to the Appendix part.

The other rules are concerning the mis-matched base and specific biological constraints from miRNA features.

6.4 Experiments and Results

Sets of human pre-miRNAs hairpins are collected to evaluate the prediction performance with the support of KB.

6.4.1 Human pre-miRNAs

The sequences of human pre-miRNAs are downloaded from the mirBase database (http://www.mirbase.org/cgi-bin/mirna_summary.pl?org=hsa), which contains 710 reported pre-miRNA entries from Homo Sapiens. Only the pre-miRNAs whose secondary structures do not contain multiple loops, the number of bases are greater than 15, their scores are less than -14.50, and the lengths are between 65 and 110, are considered.

	start		end	LP Len	Status
start: 11	11	-	16	6	O
	11	-	19	9	O
	11	-	25	15	-
	11	-	33	23	-
	11	-	39	29	-
start: 12	12	-	26	15	-
	12	-	27	16	-
	12	-	28	17	-
	12	-	29	18	-
	12	-	33	22	-
	12	-	39	28	-
	12	-	43	32	-
start: 13	13	-	17	5	O
	13	-	20	8	O
	13	-	21	9	O
	13	-	30	18	-
	13	-	36	24	-
	13	-	37	25	-
	13	-	40	28	-
start: 14	14	-	21	8	O
	14	-	22	9	O
	14	-	23	10	O
	14	-	24	11	O

	start		end	LP Len	Status
start: 15	15	-	18	4	O
	15	-	26	12	O
	15	-	27	13	O
	15	-	28	14	-
	15	-	29	15	-
	15	-	32	18	-
	15	-	33	19	-
	15	-	34	20	-
	15	-	35	21	-
	15	-	39	25	-
	15	-	43	29	-
	15	-	44	30	-
	15	-	45	31	-
	15	-	46	32	-
start: 16	16	-	21	6	O
	16	-	22	7	O
	16	-	23	8	O
	16	-	24	9	O
	16	-	37	22	-
	16	-	41	26	-

Figure 6.9: **Prediction of hsa-mir-155.** LP len indicates the loop size, and status indicates whether the prediction program executes under this case: O denotes: Yes, and - denotes: No

6.4.2 Improvement in terms of CPU usage

Since we used Knowledge Base to support the prediction, the computing time had been greatly reduced. Take prediction on hsa-mir-155 as an example, when we used prediction software without the KB, we got 171 structures based on different loops, which used 2.0 seconds. Instead, we got 56 structures, which used 0.65 seconds, when we used the same prediction software with the support of KB. Thus, we reduced the CPU time as one third as original. Let us see how it works in detail.

Fig. 6.9 shows the situation when we selected the loop that starts with 11, 12, 13, 14, 15, and 16. The sequence length is 65 so the rule1 is applied. Our fuzzy rate is 0.5 and threshold value is 14. All the computing whose LP Len is less than 14 will be blocked. So only 16 structures will be generated.

Fig. 6.10 shows the situation when we selected the loop that starts with 17, 18, 19, 20, 21, and 22. All the computing whose LP Len is less than 14 will be blocked. So 13 structures will be generated.

	start		end	LP Len	Status
start: 17	17	-	25	9	O
	17	-	31	15	-
	17	-	32	16	-
	17	-	33	17	-
	17	-	35	19	-
	17	-	38	22	-
	17	-	39	23	-
	17	-	42	26	-
	17	-	43	27	-
	17	-	45	29	-
	17	-	46	30	-
	17	-	48	32	-
start: 18	18	-	26	9	O
	18	-	27	10	O
	18	-	28	11	O
	18	-	29	12	O
	18	-	33	16	-
	18	-	39	22	-
	18	-	43	26	-
	18	-	46	29	-

	start		end	LP Len	Status
start: 19	19	-	22	4	O
	19	-	23	5	O
	19	-	24	6	O
	19	-	36	18	-
	19	-	37	19	-
	19	-	41	23	-
	19	-	49	31	-
start: 20	20	-	23	4	O
	20	-	25	6	O
	20	-	50	31	-
start: 21	21	-	25	5	O
	21	-	31	11	O
	21	-	38	18	-
	21	-	42	22	-
	21	-	48	28	-
	21	-	50	30	-
	21	-	52	32	-
start: 22	22	-	31	10	O
	22	-	38	17	-
	22	-	42	21	-
	22	-	48	27	-
	22	-	50	29	-
	22	-	52	31	-

Figure 6.10: **CONT Prediction of hsa-mir-155.** LP len indicates the loop size, and status indicates whether the prediction program executes under this case: O denotes: Yes, and - denotes: No

Fig. 6.11 shows the situation when we selected the loop that starts with 23, 24, 25, 26, 27, and 28. All the computing whose LP Len is less than 14 will be blocked. So 14 structures will be generated.

Fig. 6.12 shows the situation when we selected the loop that starts with 29, 30, 31, 32, 33, and 34. Again, all the computing whose LP Len is less than 14 will be blocked. So 13 structures will be generated.

6.4.3 Improvement in terms of accuracy

In our algorithm, we added some production rules in KB to improve the prediction accuracy. For example, we extract production rules from characteristics concerning mismatched base pairs and other production rules that are from microRNA features.Table 6.1 shows the performance comparison between prediction with KB and without KB. In terms of True positive rate, the prediction with support of KB achieved 3% improvement. In terms of True Negative rate, it achieved 4% improve-

	start		end	LP Len	Status
start: 23					
	23	-	31	9	O
	23	-	38	16	-
	23	-	42	20	-
	23	-	48	26	-
	23	-	50	28	-
	23	-	52	30	-
start: 24					
	24	-	28	5	O
	24	-	29	6	O
	24	-	32	9	O
	24	-	33	10	O
	24	-	34	11	O
	24	-	35	12	O
	24	-	39	16	-
	24	-	43	20	-
	24	-	44	21	-
	24	-	45	22	-
	24	-	46	23	-
	24	-	53	30	-
start: 25					
	25	-	30	6	O
	25	-	36	12	O
	25	-	40	16	-
	25	-	47	23	-
	25	-	49	25	-
	25	-	51	27	-
	25	-	54	30	-

	start		end	LP Len	Status
start: 26					
	26	-	30	5	O
	26	-	36	11	O
	26	-	40	15	-
	26	-	47	22	-
	26	-	49	24	-
	26	-	51	26	-
	26	-	54	29	-
	26	-	57	32	-
start: 27					
	27	-	36	10	O
	27	-	40	14	-
	27	-	47	21	-
	27	-	49	23	-
	27	-	51	25	-
	27	-	54	28	-
	27	-	57	31	-
start: 28					
	28	-	36	9	O
	28	-	40	13	O
	28	-	47	20	-
	28	-	49	22	-
	28	-	51	24	-
	28	-	54	27	-
	28	-	57	30	-

Figure 6.11: **CONT Prediction of hsa-mir-155.** LP len indicates the loop size, and status indicates whether the prediction program executes under this case: O denotes: Yes, and - denotes: No

ment.

Table 6.1: The performance comparison between prediction with KB and without KB.

Prediction Algorithms	TP rate	FP rate	TN rate	FN rate
Without KB	92%	8%	89%	11%
With KB	95%	5%	93%	7%

6.5 Discussion

From the experimental results, we can see that the prediction system with the support of KB obtained a lot of improvements in terms of accuracy and CPU time. However, we need to resolve the issue of conflict set. Fig. 6.13 shows a production system with conflict set. First, a pattern matcher looks at data and rules to make sure

	start		end	LP Len	Status
start: 29	29	-	37	9	O
	29	-	41	13	O
	29	-	55	27	-
start: 30	30	-	34	5	O
	30	-	35	6	O
	30	-	38	9	O
	30	-	39	10	O
	30	-	42	13	O
	30	-	43	14	-
	30	-	44	15	-
	30	-	45	16	-
	30	-	46	17	-
	30	-	48	19	-
	30	-	50	21	-
	30	-	52	23	-
	30	-	53	24	-
	30	-	58	29	-
	30	-	59	30	-
start: 31	31	-	40	10	O
	31	-	55	25	-

	start		end	LP Len	Status
start: 32	32	-	40	9	O
start: 33	33	-	36	4	O
	33	-	37	5	O
	33	-	40	8	O
	33	-	47	15	-
	33	-	49	17	-
	33	-	51	19	-
	33	-	54	22	-
	33	-	55	23	-
	33	-	57	25	-
	33	-	60	28	-
	33	-	61	29	-
	33	-	63	31	-
	33	-	64	32	-
start: 34	34	-	40	7	O
	34	-	55	22	-
	34	-	64	31	-
	34	-	65	32	-

Figure 6.12: **CONT Prediction of hsa-mir-155.** LP len indicates the loop size, and status indicates whether the prediction program executes under this case: O denotes: Yes, and - denotes: No

whether some rules' conditions have been satisfied. After matcher produces a list of rules (conflict set) whose conditions have been satisfied, the conflict resolver will determine (select) the best rule and fire (execute) it. In our original design, we use naive algorithm: try all the rules in sequence, stoping at the first match. In the future, we will use conflict resolution strategy. Strategies vary from the simple to complex. We want to use order approach, which assign weights or priorities to rules and sort the conflict set.

6.6 Conclusion

Identifying conserved characteristics in RNA can drastically improve the prediction [5, 10, 94]. After the characteristics are extracted from the secondary structures of human pre-microRNAs, rules are constructed from the features and further are used to create KB. We still used the original prediction algorithm, but this time we predict the secondary structure with the support from our KB. We conducted some experiments to evaluate the effectiveness and the efficiency of our new strategy. The experimental

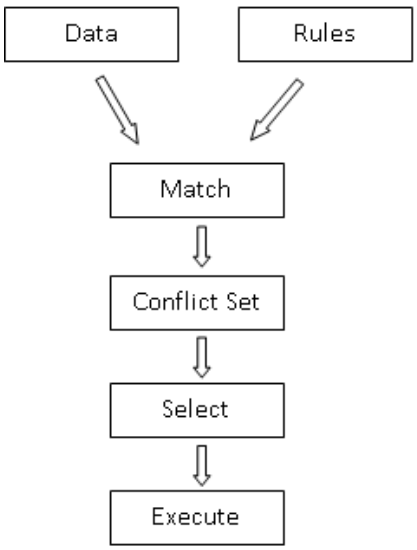


Figure 6.13: **An Production system with Conflict Set.**

results showed that with the support of KB the prediction algorithm can predict the secondary structures much faster and more accurately. More importantly, regular production rules are not adopted in the KB, instead fuzzy strategy is applied to them. The fuzzy rate is obtained from training data set. In the future, we will construct more stable and powerful KB that will incorporate more biological characteristics and features from the pre-miRNAs' structure to handle this domain problem.

7 A novel artificial poly-cistronic microRNA vector prediction and its application in silencing multiple genes in Arabidopsis

Artificial microRNA (miRNA)-directed gene silencing has advantages over traditional inverted-repeats gene silencing vector in terms of more gene silencing specificity and less off-target effects. Here we report the design of a novel poly-cistronic (poly-cis) miRNA vector that can mediate multiple gene silencing in plants. The poly-cis vector contains six modules that were modified from Arabidopsis miR168a (module 0) and the transcripts were able to be folded up into six independent stem-loop structures (module 0-5). Each module was modified to have unique sequences that allow the construction of the six modules simply by six consecutive polymerase chain reactions (PCRs) with the help of six unique restriction sites introduced to the poly-cis miRNA vector. A dedicated web-based poly-cis miRNA vecotor design interface was established to help the users to design their poly-cis miRNA-directed gene silencing construct to silence multiple genes of interest. Finally, the poly-cis miRNA vector was successfully applied to silence two Arabidopsis argonaute (AGO2 and AGO4) simultaneously. Thus, a new approach of using artificial miRNAs to silencing more than one gene at a time was made possible.

This chapter is organized as follows: Section 7.1 introduces the background and current major issues concerning this study. Section 7.2 introduces our algorithm to design a novel poly-cistronic (poly-cis) miRNA vector that can mediate multiple gene silencing at a time in plants. The experiments are carried out and the results are reported in Section 7.3. Conclusion of this chapter is in Section 7.4.

7.1 Introduction

MicroRNAs (miRNAs) are newly discovered endogenous small non-coding RNAs (21-25 nt) that regulate gene expression by targeting one of more mRNAs for translational repression or cleavage. To date, thousands of miRNAs have been found in animals

and plants. It has been shown that microRNAs play a very important role in regulation of gene expression [21, 70]. MicroRNAs are expressed at high levels in animal and plant cells during cell differentiation, apoptosis, growth, and development. Gene suppression is a powerful tool for functional genomics and silencing of specific gene products. RNA interference (RNAi) and microRNA pathways are efficient mechanisms of gene suppression, whose application in gene silencing is one of the current focuses in functional genomics and metabolic engineering [114]. With thousands of microRNA genes identified and genome sequences of diverse eukaryotes available for comparison, people begin to pay more attention on the origin and evolution of RNAi [107]. RNAi is a highly evolutionally conserved process of post transcriptional gene silencing by which double stranded RNA (dsRNA) causes efficient sequence-specific silence of gene expression through the cleavage and degradation of any mRNA that shares the same sequence of the dsRNA. It was first discovered in 1998 by Fire et al. [37]. MicroRNA pathway was discovered to have similar functions to the RNAi in down-regulating gene expression [7, 8]. The miRNA genes are located often in intergenic regions and sometimes in the intron of a coding gene [7, 8]. First, miRNA is transcribed into primary miRNA (pri-miRNA) by RNA polymerase II (pol II) [66]. Second, the pri-miRNA is processed into miRNA precursor (pre-miRNA) and then mature miRNA duplex by Dicer [67]. The short mature miRNAs interact with cellular proteins to form RISC termed miRNA associated RISC (miRISC) [52, 116]. miRISC finds its specific target mRNA and then cleaves and destroys the target mRNA in cells. Consequently, specific cellular mRNA is silenced by specific miRNA [76, 82, 116]. The discovery of RNAi and microRNA pathways has caused intensive studies on developing RNAi technologies for treating human diseases and for improving plant traits [54, 91, 115]. Currently available RNAi vectors [35] are designed to produce either short siRNAs, such as those produced by animal RNAi vectors, or long dsRNAs, such as those produced by plant RNAi vectors. Both animal and plant RNAi vectors have

shown great successes in suppressing specific gene expression. However, the RNA-dependent protein kinase (PKR) is activated by dsRNA [12, 27, 109]. In addition, in some RNA interference experiments, it has been shown that siRNAs might cause certain off-target effects through activation of PKR [99, 103]. Avoiding activation of the PKR pathway in cells, therefore, remains a major challenge to the development of RNAi technologies [115]. Another major problem associated with the current plant RNAi vectors is that the RNAi-mediated gene suppression progeny often suffers from the instability in the gene suppression suppressed [93].

In recent years, endogenous miRNAs have been designed to silence genes at high efficiency and in more gene specificity [116]. Fortunately, these modified miRNA vectors in plants and animals do not show adverse effects on their growth and development [2, 7, 8, 115]. It seems that the PKR pathway is not triggered due to the selective evolution of miRNAs. This new type of RNAi vectors based on the miRNA structures provide us with a more stable and powerful tool for repressing gene expression. Furthermore, most miRNA duplexes are highly asymmetric [72], which predestines one strand to enter the RISC to its maximum, while the other strand is destroyed [106] to reduce the off-target effects. Even though currently available RNAi and miRNA vectors have been widely used in both animals and plants, they have limitations. For example, most of these vectors can only be used to silence one gene at a time.

We propose to design a novel poly-cistronic (poly-cis) miRNA vector that can mediate multiple gene silencing at a time in plants. (1) design an artificial multi-module microRNA-based gene-silencing vector. (2) test the new poly-cis miRNA vectors for gene silence in Arabidopsis. The feasibility of this new approach was evaluated by showing that the poly-cis miRNA vector was successfully applied to silence two Arabidopsis argonaute (AGO2 and AGO4) simultaneously.

7.2 Methods and Algorithms

Developing a poly-cis miRNA vector is a novel idea that is currently not explored in plants. The idea is based on the observations that such poly-cis miRNA clusters naturally exist in both humans [45, 117] and plants [69]. The development of a poly-cis miRNA vector will not only simplify the gene silencing process, but also produce a high throughput tool for gene silencing in functional genomics. We present two parts in this section. Firstly, we describe how to design and implement the poly-cis vector by using computer techniques. Secondly, we show how to conduct the experiments and analyze the results. Overall, these essential steps are taken: (1) select a template. (2) design two DNA primers P1 and P2 that cover the stem-loop region from the template. (3) assembly the primers and promoter, terminator on this small vector. (4) construct plasmids. (5) bring the poly-cis miRNA vector containing multiple artificial miRNAs into the 5941 binary vector with XbaI site. (6) Northern blot analysis of miRNA expression and processing.

7.2.1 Construction of poly-cis miRNA vector using computer algorithms

Based on the miRNA array analysis established by Tang's lab, we choose pre-microRNA168 as the template in our work. We have many other candidate pre-microRNAs that can be used as the template. The following Fig. 7.1 shows the pre-microRNA 168 structure:

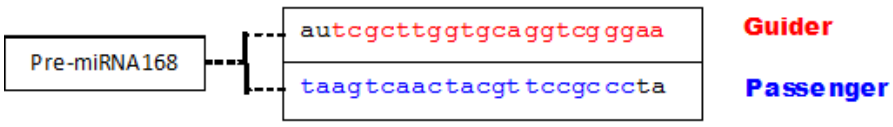


Figure 7.1: The basic structure of pre-miR168. The figure focuses on the guider strand and passenger strand. The rest part is ignored here.

Fig. 7.1 only shows the basic parts of pre-microRNA168. We may ignore the

other nucleotides in this structure for the time being. In the designing part, we need to implement some basic modules before we get two final primers P1 and P2. The first major algorithm is to predict the miR (miRNA transcript) sequence from gene coding sequence. The following pseudocode describes this algorithm.

Algorithm 1 : Predict the target miRNAs

```
1.  Select an appropriate backbone (template).
2.  Generate the pattern library: t0, t1, ..., tn;
3.  for i=0 to m-21 do (m denotes the length of gene coding sequence)
4.      form a candidate target string ci
5.      for j=0 to n do
6.          if (ci matches tj)
7.              write ci into output file
8.              break
9.          else
10.             continue
11.     endfor
12. endfor
```

Figure 7.2: Predict the target miRNAs

The input is the gene coding sequence and the output should be a set of artificial microRNAs which are of 21 nucleotides. In the very beginning, we need to select an appropriate backbone. Choosing a good backbone is heavily based on the requirement and the application. When we generate pattern library we should take the constraints which are from the template into account. For example, according to the guider of pre-miRN 168 (t c g c t t g g t g c a g g t c g g g a a) the 19th letter is g, so the 19th letter of our candidate artificial miRNA must be G or C. Sometimes we have to face such a problem: how to do if we get so many artificial microRNAs that we do not know which one is a good choice. Our solution is to let users run Basic Local Alignment Search Tool (BLAST) to go through the database and then to make a decision. Select the artificial microRNAs and run BLAST for the targeting specificity. Choose those artificial microRNAs that are only complementary to the target gene transcripts but

have no complementarity to other gene transcripts genome-wide around the 5' end of those artificial microRNAs. The second algorithm is to predict artificial miR* sequence from miR.

Algorithm 2: predict artificial miRNA*

1. Analyze the template and generate the translation table.
2. a[i]: element of artificial miRNA
3. b[i]: element of artificial miRNA*
4. for i=0 to 20 do
5. predict b[i] by looking up the a[i] in the translation table.
6. if could not determine call an algorithm based on probabilities.
7. endfor

Figure 7.3: Predict artificial miRNA*

The first step could be very difficult because it needs domain expertise. For example, suppose we select the first candidate as the miRNA: TGGGAGGTCAAG-GATTAGCAC. We get its miRNA*: GCTGTCCGGTTCTTCATCGTA. The last step is easy to understand and easy to implement as well. Algorithm 3 depicts this work.

Algorithm 3: Predict primers from the secondary structure of pre-microRNA

1. Reverse the artificial miRNA* to get P2.
2. Get the complementary string T1 for artificial miRNA string.
3. Reverse T1 to get P1.

Figure 7.4: Predict primers from the secondary structure of pre-microRNA

7.3 Experiments and Results

To validate our approach, we conducted some experiments to evaluate the efficacy of single-module miRNA vector and poly-cis miRNA vector. Firstly, we present the results generated by our web application. And then we analyze the results [110, 111] by applying single-module vector for silencing several gene and poly-cis miRNA vector for silencing more than one gene at a time.

7.3.1 Construction of poly-cis miRNA vector

In the very beginning, the user should input their gene coding sequences, as depicted in Fig. 7.5. The server will get the anti-sense of the sequence.

As shown in Fig. 7.6, the user may choose an appropriate backbone for their vector. Pre-microRNA 168 is selected in our prediction model (see Figure 12a) based on the miRNA array analysis established by Tang's lab. It is possible that server returns so many artificial miRNA candidates from a gene coding sequence that the user does not know which one is the best.

In order to select a good candidate that meets certain criteria, the user is supposed to run BLAST program (see Fig. 7.7) to achieve this goal. The purpose of running BLAST is to choose those artificial microRNAs that are only complementary to the target gene transcripts but have no complementarity to other gene transcripts genome-wide around the 5' end of those artificial microRNAs.

The results returned from BLAST are shown in Fig. 7.8. Selection of a good artificial miRNA largely depends on the user's needs and her/his domain knowledge. Take the result shown in Fig. 7.8 for example, we found that there are up to 11 matches between nucleotides 2 to nucleotide 11, and it is not a good candidate because it could trigger off-target effect.

After the user selected a candidate and converted it to an antisense the server predicts the secondary structure of the artificial pre-microRNA (see Fig. 7.7, Fig.

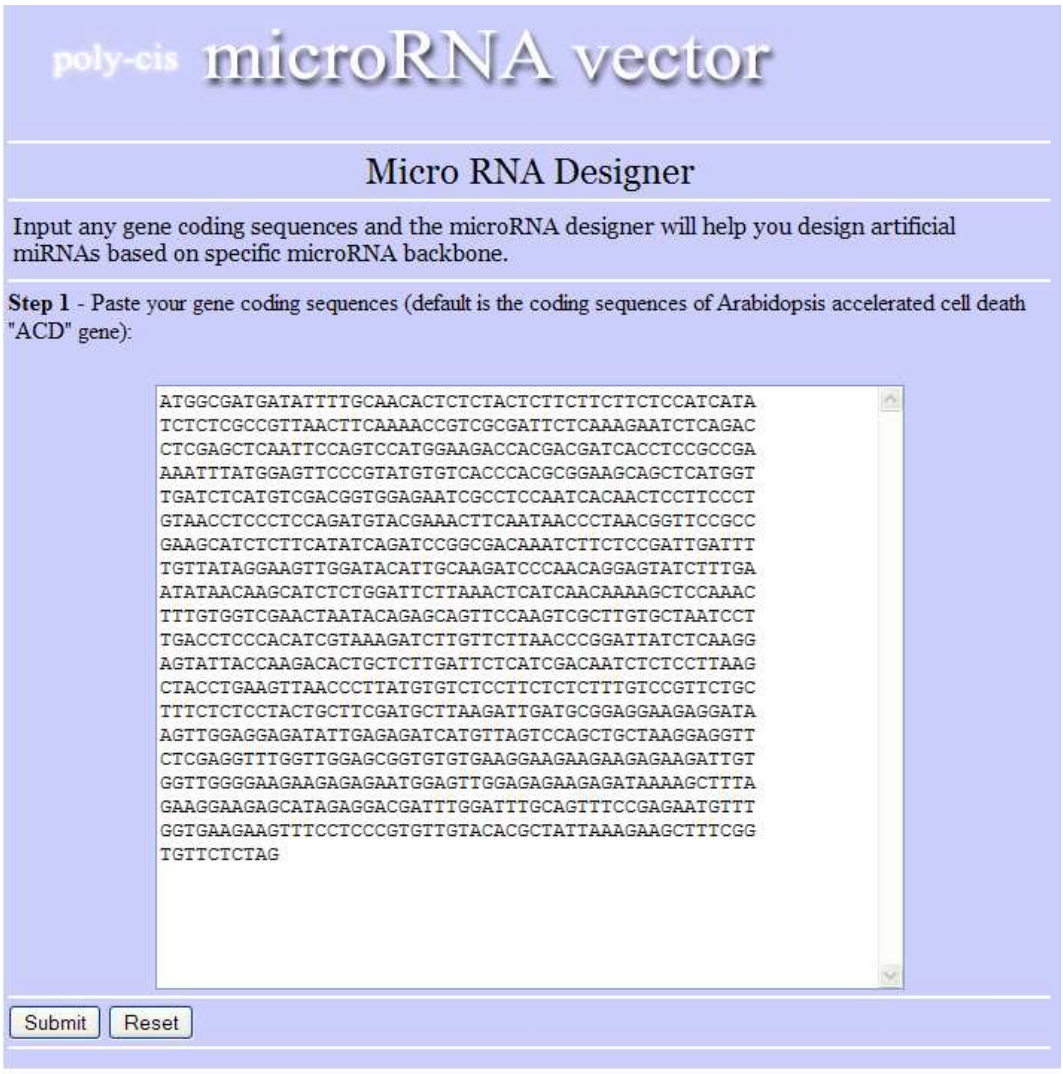


Figure 7.5: Enter gene coding sequences.

7.9, and Fig. 7.10).

The Fig. 7.10 shows the secondary structure of the artificial pre-microRNA. The red nucleotides denote the miRNA and the blue nucleotides denote the miRNA*. And they form the stem. We can see that this structure is a typical stem-loop structure. The Fig. 7.11 showed the final primers. The server is built on a Linux-based machine and programming languages are C and PERL. Our web site is available at http://www.cs.uky.edu/ $\sim$ dianweih/microRNA_vector/submit.htm.

Figure 7.6: Choose an appropriate backbone. Up to 9 backbones are available. The selection is based on the users application needs.

7.3.2 Expression of microRNAs from a monocistronic (miR168 backbone) microRNA vector

We conduct some experiments to evaluate the efficacy of specific gene expression by using single module microRNA vector. As shown in Fig. 7.12, the backbone of this vector is microRNA 168. And this vector is expected to produce the following three microRNAs: 1. Ago2: small RNA associated protein. Among the Argonaute family, the Ago2 is usually considered important because Ago2 is a critical component of RISC [75]. And it is one of major reasons that Ago2 had been mainly tested and

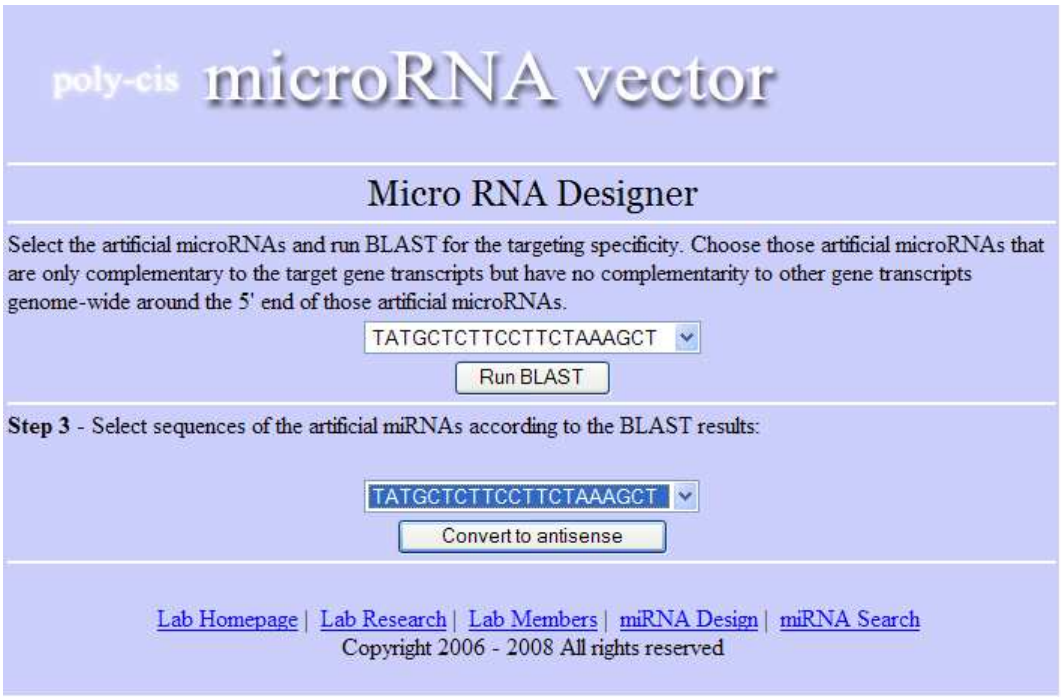


Figure 7.7: Run BLAST before you select one good candidate target microRNA.

analyzed in our experiments.

As shown in Fig. 7.13, the expression of Ago2 artificial miRNA*, ACD artificial miRNA and Terpene synthase artificial miRNA was sufficient to silence specific genes. For the AGO2 artificial miRNA, the expression was not very significant maybe because of the noise. Overall, The results indicated that this model works very well for silencing some specific genes of interests.

7.3.3 Expression of microRNAs from a poly-cis microRNA vector

While single-module microRNA vector appears to work well and it plays some role in silencing several genes in cell [24], single-module vector has a limitation on the number of genes silenced or mediated because it contains only one module. Therefore, we sought to overcome this limitation by using a poly-cis microRNA vector to mediate or silence multiple genes of interest at one time. As shown in Fig. 7.14, 6 different modules were used to construct a poly-cis vector. Each module was modified to

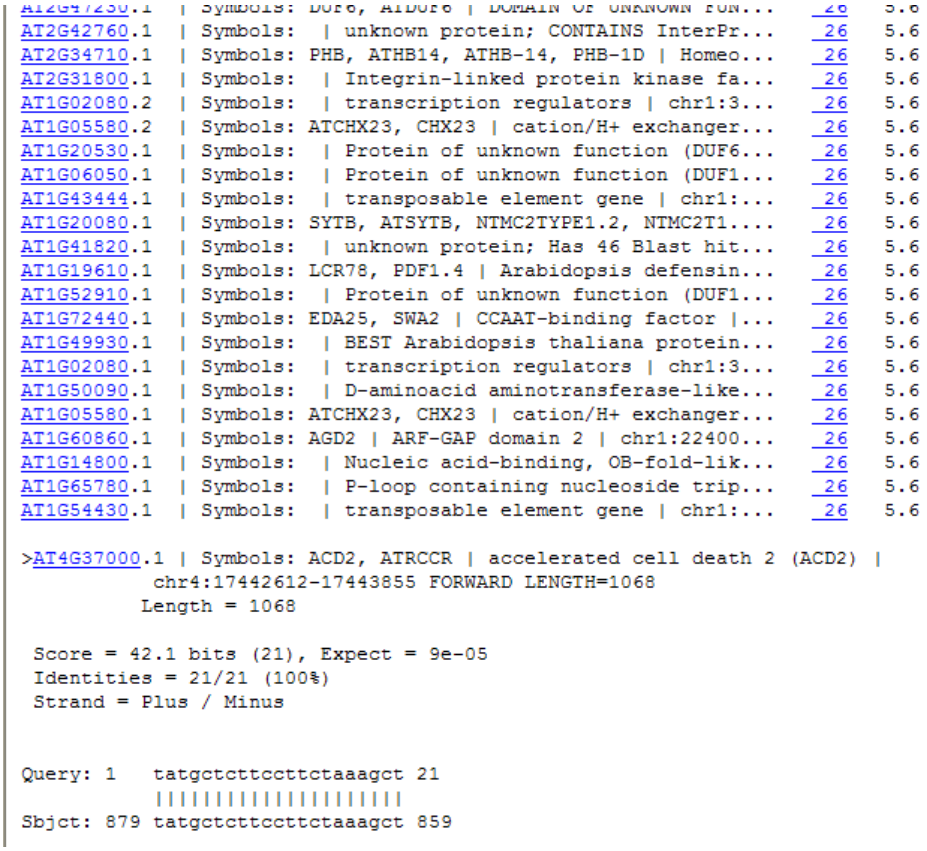


Figure 7.8: Check the result of running BLAST.

have unique sequences that allow the construction of the six modules simply by six consecutive polymerase chain reactions (PCRs) with the help of six unique restriction sites introduced to the poly-cis miRNA vector. Pre-microRNA-168 is used as the backbone on this prediction model.

A goal of this work is to create a plasmid capable of expressing multiple miRNAs targeting different regions while retaining efficacy against off-target effects. We used synthetic microRNAs in our vector to mimic the functions of natural endogenous miRNAs. Fig. 7.14b shows that Arabidopsis transgenic plants that are applied to this poly-cis vector are capable of producing specific miRNAs. When using AGO2-AGO6 artificial miRNA vector in ago2-6 transgenic lines, the expression of Ago2 is stronger than that of Ago6 (See Fig. 7.14b(1)). Likewise, as shown in Fig. 7.14b(2), when AGO2 artificial miRNA vector is used in ago2-7 transgenic lines, the expression

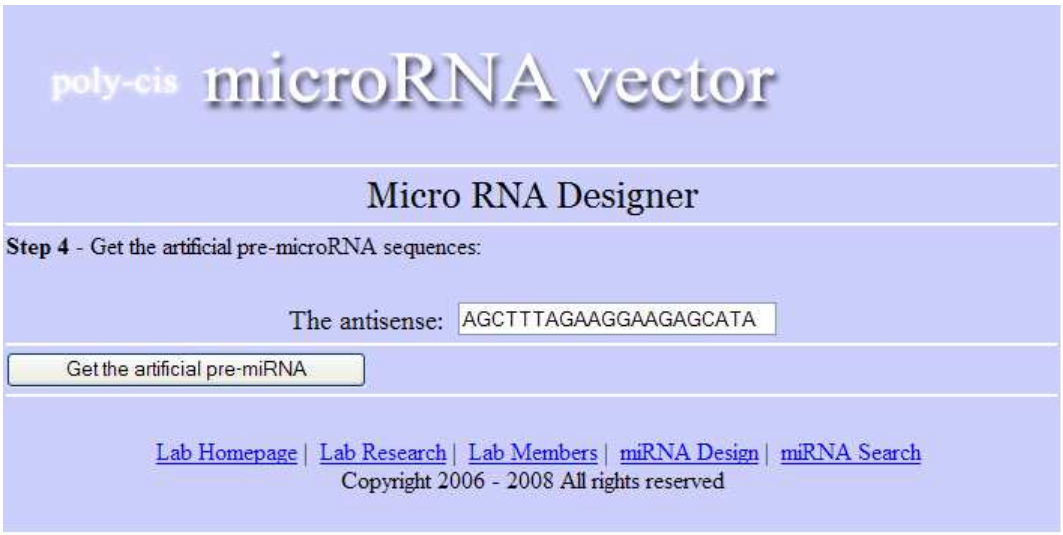


Figure 7.9: Select a candidate and convert it to antisense.

of Ago2 is very high. Fig. 7.14b(4) shows the different cases corresponding to different transgenic lines applied by an AGO2 artificial miRNA vector: the expression of AGO2-amiRNA is much higher in ago2 and ago2-6 transgenic lines than in others. In summary, AGO artificial miRNA vector in associated transgenic lines successfully produce sufficient microRNAs as expectation.

7.4 Conclusion

Based on the results, we demonstrate that it is possible to achieve efficient expression and processing of three different miRNAs (Ago2, Ago5, and Ago7) from a poly-cis microRNA vector. This study and the novel poly-cis miRNA gene vector will have greater significant applications in terms of functional genomics and plant improvement. And the new gene suppression vectors will be powerful in dissecting plant natural product pathways or metabolic engineering [114]. Rational engineering of complicated metabolic networks has largely depended on our understanding of target pathways and their associate genes, regulatory proteins and enzymes [101, 114]. The first step to metabolic engineering is to identify target pathways and their constituent genes through omics analysis. The second step is to select the target genes

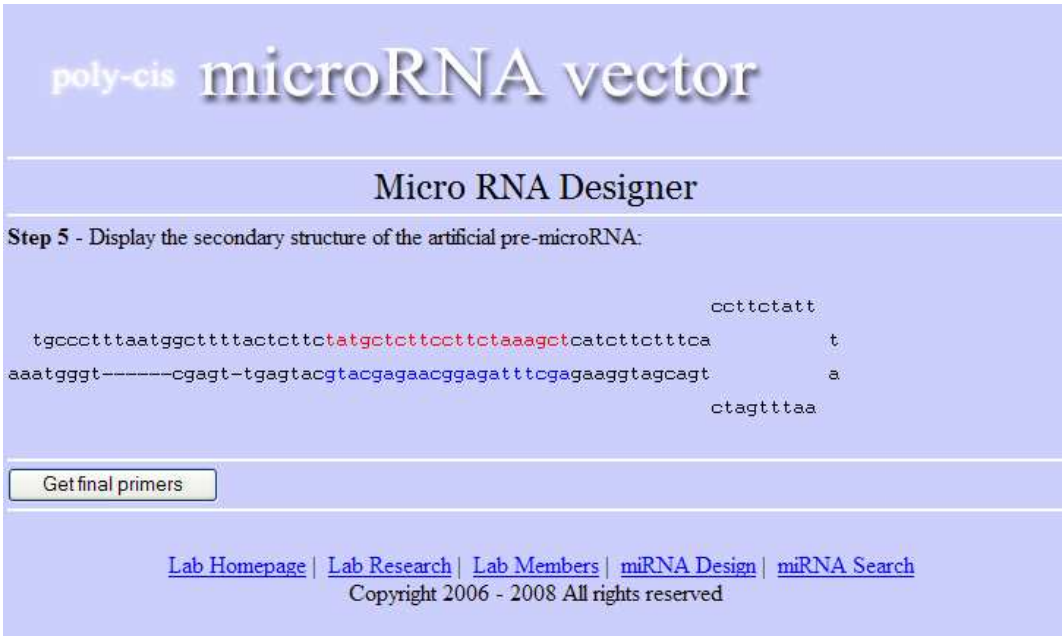


Figure 7.10: Display the secondary structure of artificial pre-microRNA.

in the pathways for engineering. The selected genes will be processed through being expressed or silenced individually or simultaneously. Poly-cis microRNA vector directed multiple gene silencing techniques will become a powerful and efficient tool in this step in silencing multiple unwanted genes simultaneously.

Figure 7.11: The primers: P1 and P2.

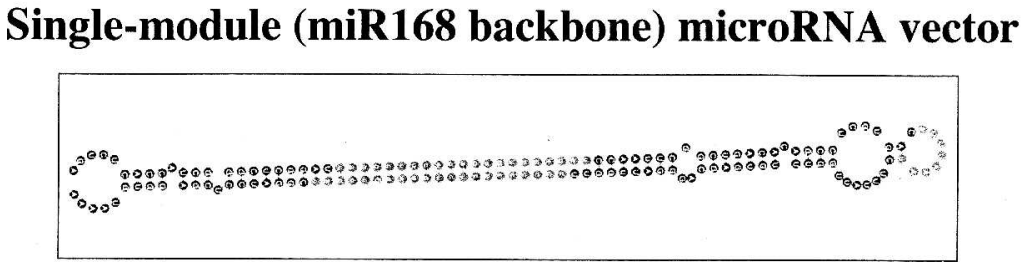


Figure 7.12: Prediction model for single-module (miR168 backbone) microRNA vector

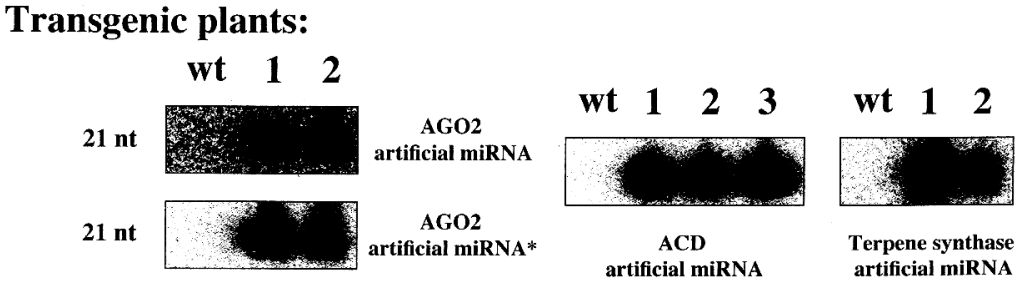


Figure 7.13: Gene expressions of three main microRNAs produced from single-module microRNA vector. wt denotes wild type.

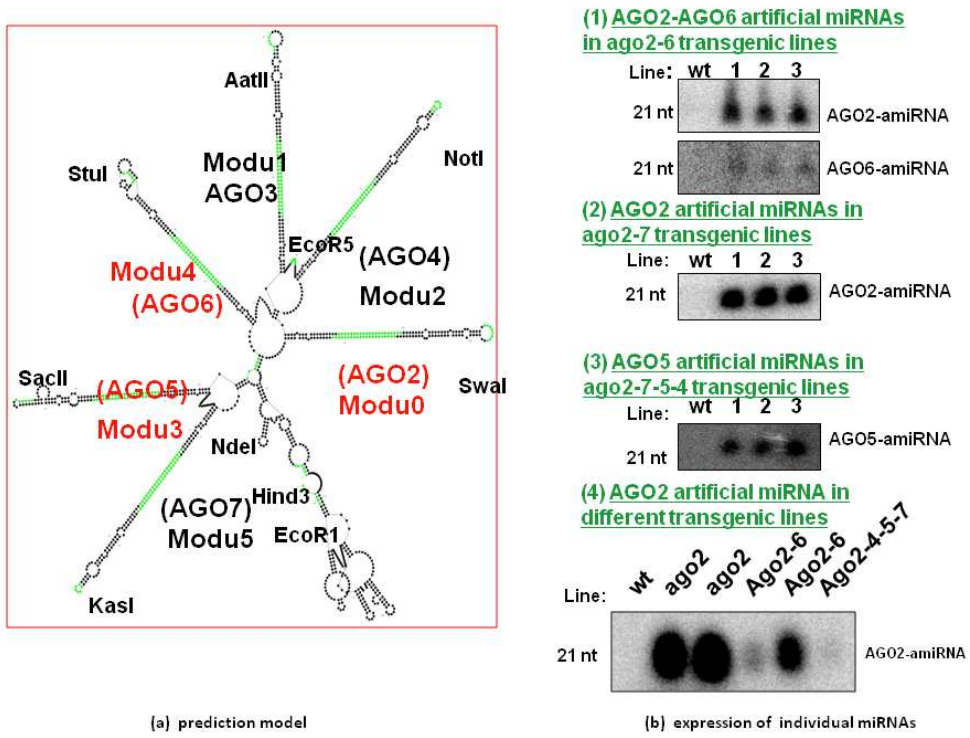


Figure 7.14: Prediction model. Pre-microRNA-168 is used as the backbone for the poly-cis miRNA vector. There are six modules on this model, which use different PCRs.

8.1 Conclusion

This dissertation introduces our research work on studying and designing computer model to predict the secondary structure of pre-microRNAs. We have conducted some research in the following directions:

- We have designed and implemented the pre-microRNA secondary structure prediction system based on energy-scoring strategy. This algorithm uses modified NCM structures and recursive method in predicting the optimal secondary structure of given sequence. The experimental results show that our prodiction algorithm achieves very encouraging performance.

- We have proposed a parallel processing technique to predict the secondary structure of endogenous polycistronic microRNAs. The benefit from using parallel computing is obvious. We use master-slave architecture and the trend of speedups of our parallel algorithm matches that of theoretical speedups.

- We have proposed a new effective method that can distinguish the real pre-miRNAs from pseudo pre-miRNAs and this approach is important for identifying novel and specific miRNAs. Our algorithm outperforms the triplet-SVM classifier, which uses sequence-structure as features in terms of accuracies. The other advantages that our method has include: our model uses fewer features (9 features) while triplet method uses 32 features, and our model decreases the dependency on the sequence data because it just uses the characteristics from the secondary structures of pre-MiRNAs.

- We have proposed an efficient and improved prediction algorithm that predicts the secondary structure with the support of knowledge base. The production

rules come from the characteristics of secondary structure and biological constraints from pre-microRNAs. The new algorithm improves the accuracy and reduces the computing time greatly.

- We have proposed a novel artificial poly-cistronic microRNA vector prediction and applied it to silence multiple genes in Arabidopsis.

- We designed and developed a web-based online tool for predicting the secondary structure of pre-microRNAs. Users can download the binary code and install our software on their machines as well. Users can run binary code on any Linux-based computer. The input is the sequence (primary structure) of pre-miRNAs and the output is predicted the secondary structure of sequence. Our state-of-the-art prediction software has intuitive and user-friendly interface, generates forecasting with just two clicks. In addition, it provides sufficient flexibility for researchers by generating five best structures based on the energy scores.

- We have implemented a web site that allows users to construct poly-cis miRNA vector on line from gene sequence. The web application is at www.cs.uky.edu/$\sim$ *dianweih/microRNA_vector/submit.htm*.

8.2 Future Work

We have done some fundamental research related to the secondary structure prediction of pre-microRNAs and obtained some encouraging results. More work needs to be fulfilled to make the prediction algorithm more mature, more accurate, and more powerful. Future work may follow the directions listed below:

- If domain knowledge could be incorporated into our model it would greatly improve the prediction. In the future, we can consider formulating a hybrid statistics/thermodynamic model, which could use the statistical frequencies as *a priori* for selecting competing thermodynamically favorable configurations.

- Apart from the above, we also plan to develop an algorithm that can obtain the most appropriate structure from a list of structures with the same score. Even though very few researches related to this aspect has been done, we do not think this issue is a trivial one, especially when the optimal solution is among such structures. In order to reach this goal, a machine learning model should be built. Some new challenges will be encountered when we work out this problem, such as how to select a better structure from two structures with the same energy score. The final judgement relies heavily on the experience of experts and the current related knowledge.

- In our experiment to predict the secondary structure of polycistronic microRNAs, our speedups is more flat than the theoretical values. We want to find the hidden factors that affect our algorithm performance negatively and propose new strategy to handle this problem. Also, we want to extend our test on more endogenous poly-cistronic miRNAs from plants and animals for their secondary structure prediction and validation.

- We also applied Knowledge Base to supporting the prediction. Due to the time limitation, we only chose human pre-miRNAs as data set. We want to choose all the mammals as data set so the Knowledge Base will be more powerful and we will take more production rules into consideration in the future.

- In order to validate and evaluate the predictive power of our system, we would like to consider a new identification algorithm, which focus on selecting the gene sequence, filtering the secondary structures, and validating the novel microRNAs. This ability to find new microRNAs using our prediction algorithm can give us more insight on designing the more reasonable and efficient algorithms.

Appendix: Production Rules

RULE 1

if (sequence length is 65) AND (fuzzy_rate is valid) *(Note: fuzzy_rate is between 0.1 and 0.9)*
AND (loop size is greater than 0)

then
action {
compute the local length; *(note: local_length = 9*(1+fuzzy_rate);)*
if (loop size <= local_length)
allows the program to continue;
else
does not accept this loop size
}

RULE 2

if (sequence length is 66) AND (fuzzy_rate is valid) *(Note: fuzzy_rate is between 0.1 and 0.9)*
AND (loop size is greater than 0)

then
action {
compute the local length; *(note: local_length = 10*(1+fuzzy_rate);)*
if (loop size <= local_length)
allows the program to continue;
else
does not accept this loop size
}

RULE 3

if (sequence length is 67) AND (fuzzy_rate is valid) *(Note: fuzzy_rate is between 0.1 and 0.9)*
AND (loop size is greater than 0)

then
action {
compute the local length; *(note: local_length = 11*(1+fuzzy_rate);)*
if (loop size <= local_length)
allows the program to continue;
else
does not accept this loop size
}

RULE 4

if (sequence length is 68) AND (fuzzy_rate is valid) *(Note: fuzzy_rate is between 0.1 and 0.9)*
AND (loop size is greater than 0)

then
action {
compute the local length; *(note: local_length = 12*(1+fuzzy_rate);)*
if (loop size <= local_length)
allows the program to continue;
else
does not accept this loop size
}

Cont. (Production Rules)

RULE 5

if (sequence length is 69) AND (fuzzy_rate is valid) *(Note: fuzzy_rate is between 0.1 and 0.9)*
AND (loop size is greater than 0)
then
action {
compute the local length; *(note: local_length = 13*(1+fuzzy_rate);)*
if (loop size <= local_length)
allows the program to continue;
else
does not accept this loop size
}

RULE 6

if (sequence length is 70) AND (fuzzy_rate is valid) *(Note: fuzzy_rate is between 0.1 and 0.9)*
AND (loop size is greater than 0)
then
action {
compute the local length; *(note: local_length = 11*(1+fuzzy_rate);)*
if (loop size <= local_length)
allows the program to continue;
else
does not accept this loop size
}

RULE 7

if (sequence length is 71) AND (fuzzy_rate is valid) *(Note: fuzzy_rate is between 0.1 and 0.9)*
AND (loop size is greater than 0)
then
action {
compute the local length; *(note: local_length = 8*(1+fuzzy_rate);)*
if (loop size <= local_length)
allows the program to continue;
else
does not accept this loop size
}

RULE 8

if (sequence length is 72) AND (fuzzy_rate is valid) *(Note: fuzzy_rate is between 0.1 and 0.9)*
AND (loop size is greater than 0)
then
action {
compute the local length; *(note: local_length = 8*(1+fuzzy_rate);)*
if (loop size <= local_length)
allows the program to continue;
else
does not accept this loop size
}

Cont. (Production Rules)

RULE 9

if (sequence length is 73) AND (fuzzy_rate is valid) *(Note: fuzzy_rate is between 0.1 and 0.9)*
AND (loop size is greater than 0)
then
action {
compute the local length; *(note: local_length = 13*(1+fuzzy_rate);)*
if (loop size <= local_length)
allows the program to continue;
else
does not accept this loop size
}

RULE 10

if (sequence length is 74) AND (fuzzy_rate is valid) *(Note: fuzzy_rate is between 0.1 and 0.9)*
AND (loop size is greater than 0)
then
action {
compute the local length; *(note: local_length = 10*(1+fuzzy_rate);)*
if (loop size <= local_length)
allows the program to continue;
else
does not accept this loop size
}

RULE 11

if (sequence length is 75) AND (fuzzy_rate is valid) *(Note: fuzzy_rate is between 0.1 and 0.9)*
AND (loop size is greater than 0)
then
action {
compute the local length; *(note: local_length = 9*(1+fuzzy_rate);)*
if (loop size <= local_length)
allows the program to continue;
else
does not accept this loop size
}

RULE 12

if (sequence length is 76) AND (fuzzy_rate is valid) *(Note: fuzzy_rate is between 0.1 and 0.9)*
AND (loop size is greater than 0)
then
action {
compute the local length; *(note: local_length = 8*(1+fuzzy_rate);)*
if (loop size <= local_length)
allows the program to continue;
else
does not accept this loop size
}

Cont. (Production Rules)

RULE 13

if (sequence length is 77) AND (fuzzy_rate is valid) *(Note: fuzzy_rate is between 0.1 and 0.9)*
AND (loop size is greater than 0)
then
action {
compute the local length; *(note: local_length = 13*(1+fuzzy_rate);)*
if (loop size <= local_length)
allows the program to continue;
else
does not accept this loop size
}

RULE 14

if (sequence length is 78) AND (fuzzy_rate is valid) *(Note: fuzzy_rate is between 0.1 and 0.9)*
AND (loop size is greater than 0)
then
action {
compute the local length; *(note: local_length = 11*(1+fuzzy_rate);)*
if (loop size <= local_length)
allows the program to continue;
else
does not accept this loop size
}

RULE 15

if (sequence length is 79) AND (fuzzy_rate is valid) *(Note: fuzzy_rate is between 0.1 and 0.9)*
AND (loop size is greater than 0)
then
action {
compute the local length; *(note: local_length = 13*(1+fuzzy_rate);)*
if (loop size <= local_length)
allows the program to continue;
else
does not accept this loop size
}

RULE 16

if (sequence length is 80) AND (fuzzy_rate is valid) *(Note: fuzzy_rate is between 0.1 and 0.9)*
AND (loop size is greater than 0)
then
action {
compute the local length; *(note: local_length = 8*(1+fuzzy_rate);)*
if (loop size <= local_length)
allows the program to continue;
else
does not accept this loop size
}

Cont. (Production Rules)

RULE 17

if (sequence length is 81) AND (fuzzy_rate is valid) *(Note: fuzzy_rate is between 0.1 and 0.9)*
AND (loop size is greater than 0)
then
action {
compute the local length; *(note: local_length = 12*(1+fuzzy_rate);)*
if (loop size <= local_length)
allows the program to continue;
else
does not accept this loop size
}

RULE 18

if (sequence length is 82) AND (fuzzy_rate is valid) *(Note: fuzzy_rate is between 0.1 and 0.9)*
AND (loop size is greater than 0)
then
action {
compute the local length; *(note: local_length = 9*(1+fuzzy_rate);)*
if (loop size <= local_length)
allows the program to continue;
else
does not accept this loop size
}

RULE 19

if (sequence length is 83) AND (fuzzy_rate is valid) *(Note: fuzzy_rate is between 0.1 and 0.9)*
AND (loop size is greater than 0)
then
action {
compute the local length; *(note: local_length = 14*(1+fuzzy_rate);)*
if (loop size <= local_length)
allows the program to continue;
else
does not accept this loop size
}

RULE 20

if (sequence length is 84) AND (fuzzy_rate is valid) *(Note: fuzzy_rate is between 0.1 and 0.9)*
AND (loop size is greater than 0)
then
action {
compute the local length; *(note: local_length = 9*(1+fuzzy_rate);)*
if (loop size <= local_length)
allows the program to continue;
else
does not accept this loop size
}

Cont. (Production Rules)

RULE 21

if (sequence length is 85) AND (fuzzy_rate is valid) *(Note: fuzzy_rate is between 0.1 and 0.9)*
AND (loop size is greater than 0)
then
action {
compute the local length; *(note: local_length = 11*(1+fuzzy_rate);)*
if (loop size <= local_length)
allows the program to continue;
else
does not accept this loop size
}

RULE 22

if (sequence length is 86) AND (fuzzy_rate is valid) *(Note: fuzzy_rate is between 0.1 and 0.9)*
AND (loop size is greater than 0)
then
action {
compute the local length; *(note: local_length = 10*(1+fuzzy_rate);)*
if (loop size <= local_length)
allows the program to continue;
else
does not accept this loop size
}

RULE 23

if (sequence length is 87) AND (fuzzy_rate is valid) *(Note: fuzzy_rate is between 0.1 and 0.9)*
AND (loop size is greater than 0)
then
action {
compute the local length; *(note: local_length = 13*(1+fuzzy_rate);)*
if (loop size <= local_length)
allows the program to continue;
else
does not accept this loop size
}

RULE 24

if (sequence length is 88) AND (fuzzy_rate is valid) *(Note: fuzzy_rate is between 0.1 and 0.9)*
AND (loop size is greater than 0)
then
action {
compute the local length; *(note: local_length = 11*(1+fuzzy_rate);)*
if (loop size <= local_length)
allows the program to continue;
else
does not accept this loop size
}

Cont. (Production Rules)

RULE 25

if (sequence length is 89) AND (fuzzy_rate is valid) *(Note: fuzzy_rate is between 0.1 and 0.9)*
AND (loop size is greater than 0)
then
action {
compute the local length; *(note: local_length = 7*(1+fuzzy_rate);)*
if (loop size <= local_length)
allows the program to continue;
else
does not accept this loop size
}

RULE 26

if (sequence length is 90) AND (fuzzy_rate is valid) *(Note: fuzzy_rate is between 0.1 and 0.9)*
AND (loop size is greater than 0)
then
action {
compute the local length; *(note: local_length = 9*(1+fuzzy_rate);)*
if (loop size <= local_length)
allows the program to continue;
else
does not accept this loop size
}

RULE 27

if (sequence length is 91) AND (fuzzy_rate is valid) *(Note: fuzzy_rate is between 0.1 and 0.9)*
AND (loop size is greater than 0)
then
action {
compute the local length; *(note: local_length = 13*(1+fuzzy_rate);)*
if (loop size <= local_length)
allows the program to continue;
else
does not accept this loop size
}

RULE 28

if (sequence length is 92) AND (fuzzy_rate is valid) *(Note: fuzzy_rate is between 0.1 and 0.9)*
AND (loop size is greater than 0)
then
action {
compute the local length; *(note: local_length = 7*(1+fuzzy_rate);)*
if (loop size <= local_length)
allows the program to continue;
else
does not accept this loop size
}

Cont. (Production Rules)

RULE 29

if (sequence length is 93) AND (fuzzy_rate is valid) *(Note: fuzzy_rate is between 0.1 and 0.9)*
AND (loop size is greater than 0)
then
action {
compute the local length; *(note: local_length = 12*(1+fuzzy_rate);)*
if (loop size <= local_length)
allows the program to continue;
else
does not accept this loop size
}

RULE 30

if (sequence length is 94) AND (fuzzy_rate is valid) *(Note: fuzzy_rate is between 0.1 and 0.9)*
AND (loop size is greater than 0)
then
action {
compute the local length; *(note: local_length = 9*(1+fuzzy_rate);)*
if (loop size <= local_length)
allows the program to continue;
else
does not accept this loop size
}

RULE 31

if (sequence length is 95) AND (fuzzy_rate is valid) *(Note: fuzzy_rate is between 0.1 and 0.9)*
AND (loop size is greater than 0)
then
action {
compute the local length; *(note: local_length = 13*(1+fuzzy_rate);)*
if (loop size <= local_length)
allows the program to continue;
else
does not accept this loop size
}

RULE 32

if (sequence length is 96) AND (fuzzy_rate is valid) *(Note: fuzzy_rate is between 0.1 and 0.9)*
AND (loop size is greater than 0)
then
action {
compute the local length; *(note: local_length = 12*(1+fuzzy_rate);)*
if (loop size <= local_length)
allows the program to continue;
else
does not accept this loop size
}

Cont. (Production Rules)

RULE 33

if (sequence length is 97) AND (fuzzy_rate is valid) *(Note: fuzzy_rate is between 0.1 and 0.9)*
AND (loop size is greater than 0)
then
action {
compute the local length; *(note: local_length = 10*(1+fuzzy_rate);)*
if (loop size <= local_length)
allows the program to continue;
else
does not accept this loop size
}

RULE 34

if (sequence length is 98) AND (fuzzy_rate is valid) *(Note: fuzzy_rate is between 0.1 and 0.9)*
AND (loop size is greater than 0)
then
action {
compute the local length; *(note: local_length = 6*(1+fuzzy_rate);)*
if (loop size <= local_length)
allows the program to continue;
else
does not accept this loop size
}

RULE 35

if (sequence length is 99) AND (fuzzy_rate is valid) *(Note: fuzzy_rate is between 0.1 and 0.9)*
AND (loop size is greater than 0)
then
action {
compute the local length; *(note: local_length = 14*(1+fuzzy_rate);)*
if (loop size <= local_length)
allows the program to continue;
else
does not accept this loop size
}

RULE 36

if (sequence length is 100) AND (fuzzy_rate is valid) *(Note: fuzzy_rate is between 0.1 and 0.9)*
AND (loop size is greater than 0)
then
action {
compute the local length; *(note: local_length = 13*(1+fuzzy_rate);)*
if (loop size <= local_length)
allows the program to continue;
else
does not accept this loop size
}

Cont. (Production Rules)

RULE 37

if (sequence length is 101) AND (fuzzy_rate is valid) *(Note: fuzzy_rate is between 0.1 and 0.9)*
AND (loop size is greater than 0)

then

action {

compute the local length; *(note: local_length = 6*(1+fuzzy_rate);)*

if (loop size <= local_length)

allows the program to continue;

else

does not accept this loop size

}

RULE 38

if (sequence length is 102) AND (fuzzy_rate is valid) *(Note: fuzzy_rate is between 0.1 and 0.9)*
AND (loop size is greater than 0)

then

action {

compute the local length; *(note: local_length = 10*(1+fuzzy_rate);)*

if (loop size <= local_length)

allows the program to continue;

else

does not accept this loop size

}

RULE 39

if (sequence length is 105) AND (fuzzy_rate is valid) *(Note: fuzzy_rate is between 0.1 and 0.9)*
AND (loop size is greater than 0)

then

action {

compute the local length; *(note: local_length = 17*(1+fuzzy_rate);)*

if (loop size <= local_length)

allows the program to continue;

else

does not accept this loop size

}

RULE 40

if (sequence length is 106) AND (fuzzy_rate is valid) *(Note: fuzzy_rate is between 0.1 and 0.9)*
AND (loop size is greater than 0)

then

action {

compute the local length; *(note: local_length = 11*(1+fuzzy_rate);)*

if (loop size <= local_length)

allows the program to continue;

else

does not accept this loop size

}

Cont. (Production Rules)

RULE 41

if (sequence length is 109) AND (fuzzy_rate is valid) *(Note: fuzzy_rate is between 0.1 and 0.9)*
AND (loop size is greater than 0)

then
action {
compute the local length; *(note: local_length = 6*(1+fuzzy_rate);)*
if (loop size <= local_length)
allows the program to continue;
else
does not accept this loop size
}

RULE 42

if (sequence length is 110) AND (fuzzy_rate is valid) *(Note: fuzzy_rate is between 0.1 and 0.9)*
AND (loop size is greater than 0)

then
action {
compute the local length; *(note: local_length = 10*(1+fuzzy_rate);)*
if (loop size <= local_length)
allows the program to continue;
else
does not accept this loop size
}

Bibliography

[1] Yael Altuvia, Pablo Landgraf, Gila Lithwick, Naama Elefant, Sbastien Pfeffer, Alexei Aravin, Michael J. Brownstein, Thomas Tuschl, Hanah Margalit. Clustering and conservation patterns of human microRNAs. *Nucleic Acids Research* , 33:2697-2706 (2005)

[2] Victor Ambros. MicroRNA pathways in flies and worms: growth, death, fat, stress, and timing. *Cell*, 113:673-676 (2003)

[3] Andrea Tanzer, Peter F. Stadler. Molecular evolution of a microRNA cluster. *Journal of molecular biology*, 339:327-335 (2004)

[4] Mohammad Anwar, Marcel Turcotte. Evaluation of RNA secondary structure motifs using regression analysis. *In proceedings of IEEE CCECE 2006*, Ottawa, Canada, page 1747-1752 (2006)

[5] Mohammad Anwar, Truong Nguyen, Marcel Turcotte. Identification of consensus RNA secondary structures using suffix arrays. *BMC Bioinformatics*, 7:244, (2006)

[6] Pierre Baldi, Søren Brunak. Neural Networks: Applications. *Bioinformatics: The Machine Learning Approach*, MIT Press, Boston, MA, page 113-155 (2001)

[7] David P. Bartel. MicroRNAs: genomics, biogenesis, mechanism, and function. *Cell*, 116:281-297 (2004)

[8] Bonnie Bartel, David P. Bartel. MicroRNAs: At the Root of Plant Development? *Plant Physiology*, 132:709-717(2003)

[9] Isaac Bentwich, Amir Avniel, Yael Karov, Ranit Aharonov, Shlomit Gilad, Omer Barad, Adi Barzilai, Paz Einat, Uri Einav, Eti Meiri, Eilon Sharon, Yael Spector, Zvi Bentwich. Identification of hundreds of conserved and nonconserved human microRNAs. *Nature Genetics*, 37:766-770 (2005)

[10] Stephan H Bernhart, Ivo L Hofacker, Sebastian Will, Andreas R. Gruber, Peter F Stadler. RNAalifold: improved consensus structure prediction for RNA alignments. *BMC Bioinformatics*, 9:474 (2008)

[11] Stephan H Bernhart, Hakim Tafer, Ulrike Mückstein, Christoph Flamm, Peter F Stadler, Ivo L Hofacker. Partition function and base pairing probabilities of RNA heterodimers. *Algorithms for Molecular Biology*, 1:3 (2006)

[12] Alan J Bridge, Stephanie Pebernard, Annick Ducraux, Anne-Laure Nicoulaz, Richard Iggo. Induction of an interferon response by RNAi vectors in mammalian cells. *Nature Genetics*, 34:263-264 (2003)

[13] C. Burges. *A tutorial on support vector machine for pattern recognition.* Kluwer Academic Publishers, 1998.

[14] C. Campbell. Kernel methods: A survey of current techniques. *Neurocomputing.*, 48:63–84, 2002.

[15] Michelle A. Carmell, Zhenyu Xuan, Michael Q. Zhang, Gregory J. Hannon. The Argonaute family: tentacles that reach into RNAi, developmental control, stem cell maintenance, and tumorigenesis. *GENES DEVELOPMENT*, 16 : 2733-2742 (2002)

[16] James C. Carrington, Victor Ambros. Role of microRNAs in plant and animal development. *Science*, 301:336-338 (2003)

[17] R. Caruana, D. Freitag. Greedy attribute selection. *In Proceedings of International Conference on Machine Learning*, Menlo Park, California, page 28-36 (1994)

[18] Chih-Chung Chang, Chih-Jen Lin. LibSVM: a library for support vector machines. 2001

[19] James Cheetham, Frank Dehne, Andrew Rau-Chaplin, Ulrike Stege, Peter J. Taillon. Solving large FPT problems on coarse grained parallel machines. *Journal of Computer and System Sciences*, 67:691-706 (2003)

[20] Larry Cheung, Paul W. Chung, Roger Stone, Wei Dai. A personal knowledge management tool that supports organizational knowledge management. *In Proceedings of the 3rd Asia-Pacific International Conference on Knowledge Management (KMAP)*, Hongkong, December, page 11-13 (2006)

[21] Lena J. Chin, Frank J. Slack. MicroRNA-Mediated Regulation of gene expression. RNA and the Regulation of Gene Expression: A Hidden Layer of Complexity, Edited by: Kevin V. Morris, Publisher: Caister Academic Press, 2008

[22] Christophe Reuzeau, Valerie Frankard, Yves Hatzfeld, Anabel Sanz, Wim Van Camp, Pierre Lejeune, Chris De Wilde, Katrien Lievens, Joris de Wolf, Ernst Vranken, Rindert Peerbolte, Willem Broekaert. Traitmill? a functional genomics platform for the phenotypic analysis of cereals. *Plant Genetic Resources*, 4:20-24 (2006)

[23] George Chuck, A Mark Cigan, Koy Saeteurn, Sarah Hake. The heterochronic maize mutant Corngrass1 results from overexpression of a tandem microRNA. *Nature Genetics*, 39:544-549 (2007)

[24] Raman M. Dasa, Nick J. Van Haterena, Gareth R. Howella, Elizabeth R. Farrellb, Fiona K. Bangsb, Victoria C. Porteousa, Elizabeth M. Manningc, Michael J. McGrewd, Kyoji Ohyamac, Melanie A. Saccoe, Pam A. Halleyb, Helen M. Sangd, Kate G. Storeyb, Marysia Placzekc, Cheryll Tickleb, Venugopal K. Naire, Stuart A. Wilsona. Data. In *Developmental Biology*, 294 : 554-563 (2006)

[25] Data. In *WordNet*, Retrieved September 28, 2011, from http://wordnetweb.princeton.edu/perl/webwn?s=data&sub=Search+WordNet &o2=&o0=1&o8=1&o1=1&o7=&o5=&o9=&o6=&o3=&o4=&h

[26] Data. In *TechTarget*, Retrieved September 21, 2011, from http://searchdatamanagement.techtarget.com/definition/data

[27] Susan Davis, AND Julia C. Watson. In vitro activation of the interferon-induced, double-stranded RNA-dependent protein kinase PKR by RNA from the 3' untranslated regions of human alpha-tropomyosin. *Proc. Natl. Acad. Sci.* ,93:508-513 (1993)

[28] Randall Davis, Howard Shrobe, Peter Szolowits. What is a Knowledge Representation?, *AI Magazine*, 14(1): 17-33 (1993)

[29] Database. In *Wikipedia*, Retrieved August 28, 2011, from http://en.wikipedia.org/wiki/Database

[30] Ahmet M. Denli, Bastiaan B. J. Tops, Ronald H. A. Plasterk, Ren F. Ketting, Gregory J. Hannon. Processing of primary microRNAs by the Microprocessor complex. *Nature*, 432: 231-235(2004)

[31] Chuong B. Do, Daniel A. Woods, Serafim Batzoglou. CONTRAfold: RNA secondary structure prediction without physics-based models. *Bioinformatics*, 22(14):e90-e98 (2006)

[32] Jean-Pierre Dumas, Jacques Ninio. Efficient algorithms for folding and comparing nucleic acid sequences. *Nucleic Acids Research*, 10:197-206 (1982)

[33] Sean R. Eddy, Richard Durbin. RNA sequence analysis using covariance models. *Nucleic Acids Research*, 22(11):2079-2088 (1994)

[34] U.M. Fayyad, K.B. Irani. The attribute selection problem in decision tree generation. *In Proceedings of the Ninth National Conference on Artificial Intelligence*, page 104-110 (1992)

[35] Gwen D. Fewell, Karin Schmitt. Vector-based RNAi approaches for stable, inducible and genome-wide screens. *Drug Discovery Today* , 11:975-982(2006)

[36] Thomas R. Fink, Donald M. Crothers. Free energy of imperfect nucleic acid helices. I. The bulge defect. *Journal of Molecular Biology*, 66:1-12 (1972)

[37] Andrew Fire, SiQun Xu, Mary K. Montgomery, Steven A. Kostas, Samuel E. Driver,Craig C. Mello. Potent and specific genetic interference by double-stranded RNA in Caenorhabditis elegans. *Nature*, 391(6669): 806-811 (1998)

[38] Xuezheng Fu, Hao Wang, Robert W. Harrison, William L. Harrison. A rule-based approach for RNA pseudoknot prediction. *International Journal of Data Mining and Boinformatics*, 2:78-93 (2008)

[39] Sam Griffiths-Jones, Harpreet Kaur Saini, Stijn van Dongen, Anton J. Enright. miRBase: tools for microRNA genomics. *Nucleic Acids Research*, 36: D154-D158 (2008)

[40] Sam Griffiths-Jones, Russell J. Grocock, Stijn van Dongen, Alex Bateman, Anton J. Enright. miRBase: microRNA sequences, targets and gene nomenclature. *Nucleic Acids Research*, 34: D140-D144 (2006)

[41] Sam Griffiths-Jones. The microRNA registry. *Nucleic Acids Research*, 32: D109-D111 (2004)

[42] Duncan R. Groebe, Olke C. Uhlenbeck. Thermal stability of RNA hairpins containing a four-membered loop and a bulge nucleotide. *Biochemistry*, 28:742-747 (1989)

[43] Sreelatha Guddeti, Dechun Zhang, Aili Li, Chuck H Leseberg, Hui Kang, Xiaoguang Li, Wenxue Zhai, Mitrick A Johns, Long Mao. Molecular evolution of the rice miR395 gene family. *Cell Research*, 15:631-638 (2005)

[44] Dianwei Han, Guiliang Tang, Jun Zhang. A novel method for microRNA secondary structure prediction using a bottom-up algorithm. *In Proceedings of the 47th Annual ACM Southeast Conference (ACMSE '09)*, Clemson, SC, USA. March 19-21,6 pages, (2009)

[45] Lin He, J. Michael Thomson, Michael T. Hemann, Eva Hernando-Monge, David Mu, Summer Goodson, Scott Powers, Carlos Cordon-Cardo, Scott W. Lowe, Gregory J. Hannon, Scott M. Hammond. A microRNA polycistron as a potential human oncogene. *Nature* , 435, 828-833 (2005)

[46] Lin He, Gregory J. Hannon. MicroRNAs: small RNAs with a big role in gene regulation. *Nature Reviews Genetics*, 5:522-531 (2004)

[47] Ivo L. Hofacker, Martin Fekete, Christoph Flamm, Martijn A. Huynen, Susanne Rauscher, Paul E. Stolorz, Peter F. Stadler. Automatic detection of conserved RNA structure elements in complete RNA virus genomes. *Nucleic Acids Research*, 26:3825-3836 (1998)

[48] Ivo L. Hofacker, Walter Fontana, Peter F. Stadler, Lukas Sebastian Bonhoeffer, Manfred Tacker, Peter Schuster. Fast folding and comparison of RNA secondary structures (the Vienna RNA package). *Monatshefte Fur Chemie*, 125:167-188 (1994)

[49] Lawrence B. Holder, ZdravkoMarkov, Ingrid Russell. Advances in Knowledge Acquisition and Representation. *International Journal on Artificial Intelligence Tools (IJAIT)*, 15:867-874 (2006)

[50] Tzung-Pei Hong, Yeong-Chyi Lee, Min-Thai Wu. Using the master-slave parallel architecture for genetic-fuzzy data mining. *Systems, Man and Cybernetics, 2005 IEEE International Conference on*, 4:3232-3237 (2005)

[51] C. W. Hsu and C. J. Lin. A comparison of methods for multi-class support vector machines. Technical report, Department of Computer Science and Information Engineering, National Taiwan University, Taipei, Taiwan, 2001

[52] György Hutvágner, Phillip D. Zamore. A microRNA in a Multiple-Turnover RNAi Enzyme Complex. *Science*, 297: 20562060 (2002)

[53] Donna Karolchik, Baertsch R, Diekhans M, Furey TS, Hinrichs A, Lu YT, Roskin KM, Schwartz M, Sugnet CW, Thomas DJ, Weber RJ, Haussler D, Kent WJ. The UCSC Genome Browser Database. *Nucleic Acids Research*, 31:51-54 (2003)

[54] Hiroaki Kawasaki, Kazunari Taira. Short hairpin type of dsRNAs that are controlled by tRNA(Val) promoter significantly induce RNAi-mediated gene silencing in the cytoplasm of human cells. *Nucleic Acids Research*, 31:700-707 (2003)

[55] Young-Kook Kim, Jieun Yu, Tae Su Han, Seong-Yeon Park, Bumjin Namkoong, Dong Hyuk Kim, Keun Hur, Moon-Won Yoo, Hyuk-Joon Lee, Han-Kwang Yang, V. Narry Kim. Functional links between clustered microRNAs: suppression of cell-cycle inhibitors by microRNA clusters in gastric cancer. *Nucleic Acids Research*, 37:1672-1681 (2009)

[56] V. Narry Kim, Jin-Wu Nam. Genomics of microRNA. *Trends in Genetics*, 22:165-173 (2006)

[57] V. Narry Kim. MicroRNA biogenesis: coordinated cropping and dicing. *Nature Reviews Molecular Cell Biology*, 6:376385(2005)

[58] David Kincaid, Ward Cheney. Numerical Analysis: Mathematics of Scientific Computing (Third Edition). *the Brooks/Cole Series in Advanced Mathematics*, edited by Paul J. Sally, page 273-287 (2002)

[59] Knowledge Base. In *TechTarget*, Retrieved September 21, 2011, from http://searchcrm.techtarget.com/definition/knowledge-base

[60] Knowledge Base. In *Wikipedia*, Retrieved August 28, 2011, from http://en.wikipedia.org/wiki/Knowledge_base

[61] T. Joachims. *Making large-scale SVM learning practical*, Advances in Kernel Methods Support Vector Learning, B. Schölkopf, C. Burges and A. Smola (ed.), MIT-Press, 1999.

[62] G.H. John, R. Kohavi, K. Pfleger. Irrelevant feature and the subset selection problem. *In Proceedings of Eleventh International Conference on Machine Learning*, New Brunswick, N.J., page 121-129 (1994)

[63] Matthew W Jones-Rhoades, David P Bartel. Computational Identification of Plant MicroRNAs and Their Targets, Including a Stress-Induced miRNA. *Molecular Cell*, 14(6):787-799 (2004)

[64] Eric C Lai, Pavel Tomancak, Robert W Williams, Gerald M Rubin. Computational identification of Drosophila microRNA genes. *Genome Biology*, 4(7):R42 (2003)

[65] Rikard Land. Software deterioration and maintainability: A model proposal. In *Proceedings of the Second Conference on Software Engineering Research and Practice in Sweden (SERPS)*, Karlskrona, Sweden. Research Report 2002:10 (2002)

[66] Yoontae Lee, Minju Kim, Jinju Han, Kyu-Hyun Yeom, Sanghyuk Lee, Sung Hee Baek,V Narry Kim. MicroRNA genes are transcribed by RNA polymerase II. *The EMBO Journal*, 23:4051-4060 (2004)

[67] Yoontae Lee, Kipyoung Jeon, Jun-Tae Lee, Sunyoung Kim, V. Narry Kim. MicroRNA maturation: stepwise processing and subcellular localization. *The EMBO Journal*, 21:4663-4670 (2002)

[68] Yoontae Lee, Chiyoung Ahn, Jinju Han, Hyounjeong Choi, Jaekwang Kim, Jeongbin Yim, Junho Lee, Patrick Provost, Olof Rdmark, Sunyoung Kim, V. Narry Kim. The nuclear RNase III Drosha initiates microRNA processing. *Nature*, 425:415-419 (2003)

[69] Aili Li, Long Mao. Evolution of plant microRNA gene families. *Cell Research*, 17: 212218 (2007)

[70] Zhongxing Liang,Hui Wu, Santosh Reddy, Aizhi Zhu, Sijia Wang, Dean Blevins, Younghyoun Yoon, Yawei Zhang, Hyunsuk Shim. newblock Blockade of invasion and metastasis of breast cancer cells via targeting CXCR4 with an artificial microRNA. *Biochemical and Biophysical Research Communications*, 363(3): 542-546(2007)

[71] Yongmin Li, Shaogang Gong, Heather Liddell. Support vector regression and classification based multiview face detection and recognition. In *Proc. of the IEEE International Conference on Automatic Face and Gesture Recognition (FGR'00).*, Grenoble, France, 2000.

[72] Shi-Lung Lin,Donald Chang, Shao-Yao Ying. Asymmetry of intronic pre-miRNA structures in functional RISC assembly. *Gene* , 356: 32-38(2005)

[73] Stinus Lindgreen, Paul P. Gardner, Anders Krogh. Mastr: multiple alignment and structure prediction of non-coding RNAs using simulated annealing. *Bioinformatics*, 23:3304-3311 (2007)

[74] Andreas Lingel, Bernd Simon, Elisa Izaurralde, Michael Sattler. Nucleic acid 3'-end recognition by the Argonaute2 PAZ domain. *Nature Structural & Molecular Biology*,11:576-577 (2004)

[75] Andreas Lingel, Bernd Simon, Elisa Izaurralde, Michael Sattler. Structure and nucleic-acid binding of the Drosophila Argonaute 2 PAZ domain. *Nature*,426: 465-469 (2003)

[76] Cesar Llave, Zhixin Xie, Kristin D. Kasschau, James C. Carrington. Cleavage of Scarecrow-like mRNA Targets Directed by a Class of Arabidopsis miRNA. *Science* , 297:2053-2056 (2002)

[77] Ying Liu. Active learning with support vector machine applied to gene expression data for cancer classification. *Journal of Chemical Information and Computer Sciences*, 44:1936-1941 (2004)

[78] Huiqing Liu, Jinyan Li, Limsoon Wong. A comparative study on feature selection and classification methods using gene expression profiles and proteomic patterns. *Genome Informatics*, 13:51-60 (2002)

[79] H. Liu, R. Setiono. A probabilistic approach to feature selection - a filter solution. *In Proceedings of International Conference on Machine Learning*, Bari, Italy, page 319-327 (1996)

[80] Carl E. Longfellow, Ryszard Kierzek, Douglas H. Turner. Thermodynamic and spectroscopic study of bulge loops in oligoribonucleotides. *Biochemistry*, 29:278-285 (1990)

[81] Jin-Biao Ma, Keqiong Ye, Dinshaw J. Patel. Structural basis for overhang-specific small interfering RNA recognition by the PAZ domain. *Nature*, 429:318322 (2004)

[82] Allison C Mallory, Brenda J Reinhart, Matthew W Jones-Rhoades, Guiliang Tang, Phillip D Zamore, M Kathryn Barton,David P Bartel. MicroRNA control of PHABULOSA in leaf development: importance of pairing to the microRNA 5' region. *The EMBO Journal*, 23:3356-3364 (2004)

[83] David H. Mathews, D. H. Turner. Prediction of RNA secondary structure by free energy minimization. *Current Opinion in Structural Biology*, 16:270-278 (2006)

[84] David H. Mathews, Michael Zuker. Predictive Methods Using RNA Sequences. *Bioinformatics: A Practical Guide to the Analysis of Genes and Proteins, 3rd ed.*, edited by Andreas D. Baxevanis and B.F. Francis Ouellette, John wiley & sons, page 144-167 (2005)

[85] David H. Mathews, D. H. Turner. Dynalign: An algorithm for finding the secondary structure common to two rna sequences. *Journal of Molecular Biology*, 317:191-203 (2002)

[86] David H. Mathews, Jeffrey Sabina, Michael Zuker. Expanded sequence dependence of thermodynamic parameters improves prediction of RNA secondary structure. *Journal of Molecular Biology*, 288:911-940 (1999)

[87] Brian W. Matthews. Comparison of the predicted and observed secondary structure of T4 phage lysozyme. *Biochimica Biophysica Acta*, 405:442-451 (1975)

[88] John S. McCaskill. The equilibrium partition function and base pair binding probabilities for RNA secondary structure. *Biopolymers*, 29:1105-1119 (1990)

[89] Francisco Merchan, Adnane Boualem, Martin Crespi, Florian Frugier. Plant polycistronic precursors containing non-homologous microRNAs target transcripts encoding functionally related proteins. *Genome Biology*, 10:R136 (2009)

[90] Marvin Minsky. A framework for representing knowledge. *In the Psychology of Computer Vision*, McGraw-Hill, page 211-277 (1975)

[91] Makoto Miyagishi, Hidetoshi Sumimoto3, Hiroyuki Miyoshi, Yutaka Kawakami, Kazunari Taira. Optimization of an siRNA-expression system with an improved hairpin and its significant suppressive effects in mammalian cells. *Journal of Gene Medicine* , 6:715-723 (2004)

[92] Minh N. Nguyen, Jagath C. Rajapakse. Prediction of protein secondary structure with two-stage multi-class SVMs. *International Journal of Data Mining and Boinformatics*, 1:248-269 (2007)

[93] Patrick J. Paddison, Amy A. Caudy, Gregory J. Hanno. Stable suppression of gene expression by RNAi in mammalian cells. *Proc Natl Acad Sci* , 99(3): 1443-1448 (2002)

[94] Marc Parisien, Francois Major. The MC-Fold and MC-Sym pipeline infers RNA structure from sequence data. *Nature*, 452:51-55 (2008)

[95] Partition Function. In *Wikipedia*, Retrieved Oct. 28, 2011, from http://en.wikipedia.org/wiki/Partition_function_(statistical_mechanics)

[96] J. Platt, N. Cristianini, and J. Shawe-Taylor. Large margin DAGS for multiclass classification. *Advances in Neural Information Processing Systems*, 12 ed. S.A. Solla, T.K. Leen and K.-R. Muller, MIT Press, 2000.

[97] Kim D. Pruitta, Donna R. Maglott. RefSeq and LocusLink: NCBI gene-centered resources. *Nucleic Acids Research*, 29(1):137-140 (2001)

[98] http://pseudoviewer.inha.ac.kr/

[99] Sujiet Puthenveetil, Landon Whitby, Jin Ren, Kevin Kelnar, Joseph F Krebs, Peter A Beal. Controlling activation of the RNA-dependent protein kinase by siRNAs using site-specific chemical modification. *Nucleic Acids Research*, 34:4900-4911 (2006)

[100] R. Rifkin and A. Klautau. In Defense of One-Vs-All Classification. *Journal of Machine Learning Research*, 5:101–141, 2004.

[101] Heiko Rischer, Matej Orešič, Tuulikki Seppänen-Laakso, Mikko Katajamaa, Freya Lammertyn, Wilson Ardiles-Diaz, Marc C. E. Van Montagu, Dirk Inzé, Kirsi-Marja Oksman-Caldentey, Alain Goossens. Gene-to-metabolite networks for terpenoid indole alkaloid biosynthesis in Catharanthus roseus cells. *Proc Natl Acad Sci* , 103(14): 5614-5619 (2006)

[102] Antony Rodriguez, Sam Griffiths-Jones, Jennifer L. Ashurst, Allan Bradley. Identification of Mammalian microRNA Host Genes and Transcription Units. *Genome Research*, 14:1902-1910 (2004)

[103] S. Michael Rothenberg, Jeffrey A. Engelman, Sheila Le, David J. Riese II, Daniel A. Haber, Jeffrey Settleman. Modeling oncogene addiction using RNA interference. *Proc Natl Acad Sci* , 105 :12480-12484 (2008)

[104] P. Saetrom, Ola Sn⊘ve, JR. Magnar Nedland, Thomas B. Grünfeld, Yun Lin, Michael B. Bass, Jude R. Canon. Conserved microRNA characteristics in mammals. *Oligonucleotides* , 16:115-144 (2006)

[105] Yasubumi Sakakibara, Michael Brown, Richard Hughey, I. Saira Mian, Kimmen Sjölander, Rebecca C. Underwood, David Haussler. Stochastic context-free grammars for tRNA modeling. *Nucleic Acids Research*, 22(23):5112-5120 (1994)

[106] Dianne S. Schwarz, György Hutvágner, Tingting Du, Zuoshang Xu, Neil Aronin, Phillip D. Zamore. Asymmetry in the Assembly of the RNAi Enzyme Complex. *Cell*, 115:199-208 (2003)

[107] Svetlana A. Shabalina, Eugene V. Koonin. Origins and evolution of eukaryotic RNA interference. *Trends Ecol Evol.* , 23:578-587 (2008)

[108] R.M. Skirvin, H. Abu-qaoud, S. Sriskandarajah, D.E. Harry. Genetics of micropropagated woody plants. *Micropropagation of Woody Plants*, Edited By M.R. Ahuja, Kluwer Academic publishers, Netherlands. page 121-152 (1993)

[109] Carol A. Sledz, Michelle Holko, Michael J. de Veer, Robert H. Silverman, Bryan R.G. Williams. Activation of the interferon system by short-interfering RNAs. *Nature Cell Biology*, 5:834-839 (2003)

[110] Lindsey L. Snyder, Jonathan M. Esser, Catherine J. Pachuk,Laura F. Steel. Vector design for liver specific expression of multiple interfering RNAs that target hepatitis B virus transcripts. *Antiviral Res*, 80:36-44 (2008)

[111] Jiyeon Son, Pradeep D. Uchil, Young Bong Kim, Premlata Shankar, Priti Kumar, Sang-Kyung Lee Effective suppression of HIV-1 by artificial bispecific miRNA targeting conserved sequences with tolerance for wobble base-pairing. *Biochemical and Biophysical Research Communications*, 374:214-218(2008)

[112] Ji-Joon Song, Stephanie K. Smith, Gregory J. Hannon,Leemor Joshua-Tor. Crystal Structure of Argonaute and Its Implications for RISC Slicer Activity. *Science* , 305:1434-1437 (2004)

[113] Gary D. Stormo An Overview of RNA Structure Prediction and Applications to RNA Gene Prediction and RNAi Design. *Current Protocols in Bioinformatics*, Volume: Chapter 12, Pages: Unit 12.1 (2006)

[114] Guiliang Tang, Gad Galili, Xun Zhuang. RNAi and microRNA: Breakthrough technologies for the improvement of plant nutritional value and metabolic engineering. *Metabolomics*, 3:357-369 (2007)

[115] Guiliang Tang, Gad Galili. Using RNAi to improve plant nutritional value: from mechanism to application. *Trends in Biotechnology*, 22: 463-469 (2004)

[116] Guiliang Tang, Brenda J. Reinhart, David P. Bartel,Phillip D. Zamore. A biochemical framework for RNA silencing in plants. *Genes Dev.*, 17:49-63 (2003)

[117] Andrea Tanzer, Chris T. Amemiya, Chang-Bae Kim, Peter F. Stadler. Evolution of MicroRNAs Located Within Hox Gene Clusters. *J Exp Zool B Mol Dev Evol.*, 304(1):75-85 (2005)

[118] Kerry Trentelman. Survey of Knowledge Representation and Reasoning Systems. in *Science And Technology* ,(2009)

[119] V.N. Vapnik. *The Nature of Statistical Learning Theory*. Springer, New York, 1995.

[120] V.N. Vapnik. Statistical Learning Theory. *John Wiley and Sons*, New York, (1998)

[121] V. Vural and J. G. Dy. A hierarchical method for multi-class support vector machines. In *Proc. of the Twenty-first International Conference on Machine Learning.*, Banff, Alberta, Canada, July 2004

[122] Christian Wagner. Breaking the Knowledge Acquisition Bottleneck Through Conversational Knowledge Management. *Information Resources Management Journal*, 19:70-83, (2006)

[123] Christian Wagner. End-users as expert system developers. *Journal of End User Computing*, 12:3-13, (2000)

[124] Xiaowo Wang, Jing Zhang, Fei Li, Jin Gu, Tao He, Xuegong Zhang, Yanda Li. MicroRNA identification based on sequence and structure alignment. *Bioinformatics*, 21:3610-3614 (2005)

[125] Stefan Washietl, Ivo L Hofacker, Melanie Lukasser, Alexander Hüttenhofer, Peter F Stadler. Mapping of conserved RNA secondary structures predicts thousands of functional noncoding RNAs in the human genome. *Nature Biotechnology*, 23:1383-1390 (2005)

[126] Donald Arthur Waterman. *A guide to expert systems*, Reading, PA: Addison-Wesley(1986)

[127] Michael S. Waterman, Thomas H. Byers. A dynamic programming algorithm to find all solutions in the neighborhood of the optimum. *Mathematical Biosciences*, 77:179-188 (1985)

[128] Tianbing Xia, David H. Mathews, Douglas H. Turner. Thermodynamics of RNA secondary structure formation. *In Prebiotic Chemistry, Molecular Fossils, Nucleotides, and RNA*, edited by D. G. So"ll, S. Nishimura, and P. B. Moore, Elsevier, page 21-48 (1999)

[129] Tianbing Xia, J.A. McDowell, Douglas H. Turner. Thermodynamics of nonsymmetric tandem mismatches adjacent to G·C base pairs in RNA. *Biochemistry*, 36:12486-12487 (1997)

[130] Chenghai Xue, Fei Li, Tao He, Guo-Ping Liu, Yanda Li, Xuegong Zhang. Classification of real and pseudo microRNA precursors using local structure-sequence features and support vector machine. *BMC Bioinformatics*, 6:310 (online) (2005)

[131] Yan Zeng, Rui Yi, Bryan R Cullen. Recognition and cleavage of primary microRNA precursors by the nuclear processing enzyme Drosha. *EMBO Journal*, 24:138-148 (2005)

[132] Yan Zeng, Bryan R Cullen. Structural requirements for pre-microRNA binding and nuclear export by Exportin 5. *Nucleic Acids Research*, 32(16):4776-4785(2004)

[133] Nouhad J. Rizk. Parallel and Evolutionary Approaches to Computational Biology. *Parallel Computing for Bioinformatics and Computational Biology: Models, Enabling Technologies, and Case Studies*, edited by Albert Y. Zomaya page 1-816 (2006)

[134] Michael Zuker. Mfold web server for nucleic acid folding and hybridization prediction. *Nucleic Acids Research*, 31(13):3406-3415 (2003)

[135] Michael Zuker, Patrick Stiegler. Optimal computer folding of large RNA sequences using thermodynamics and auxiliary information. *Nucleic Acids Research*, 9:133-148 (1981)

Vita

Personal Data:

Name: Dianwei Han

Date of Birth: 01/09/1973

Place of Birth: Yingkou, China

Educational Background:

- Master of Science in Computer Science, Lamar University, USA, 2003.
- Master of Science in Computer Science, Beijing Institute of Technology, China, 1995.
- Bachelor of Engineering in Computer Engineering, North China University of Technology, China, 1990.

Professional Experience:

- Teaching Assistant, 09/2007 - 05/2011. Department of Computer Science, University of Kentucky, Kentucky.
- Research Assistant, 01/2005 - 05/2007. Department of Computer Science, University of Kentucky, Kentucky.
- Web Master, 09/1999 - 12/2004. College of Engineering, Lamar University, Texas
- Software Engineer, 06/1995 - 07/1998. Hitachi Corporation, Beijing, China

Awards:

- Student Travel Support from Graduate School Fellowship of the University of Kentucky, 2006-2010.
- The third place for the poster section at the 2nd Annual Midwest Symposium on Computational Biology Bioinformatics, Oct. 4th, 2008, UIUC (the University of Illinois)
- Award of Merit for a paper presentation at the 20th Annual EKU Symposium in the Mathematical, Statistical and Computer Sciences, 2006.

Refereed Publications:

- Journal Publications
  - Dianwei Han, Guiliang Tang, and Jun Zhang, MicroRNAfold: pre-microRNA secondary structure prediction based on Modified NCM model with thermodynamics-based scoring strategy, *International Journal of Data Mining and Bioinformatics* , (2011), in press.
  - Dianwei Han, Guiliang Tang, and Jun Zhang, A Parallel Strategy for Predicting the Secondary Structure of Polycistronic MicroRNAs, *International Journal of Bioinformatics Research and Application*, (2011), in Press.
  - Shuting Xu, Jun Zhang, Dianwei Han, and Jie Wang. Singular value decomposition based data distortion strategy for privacy protection. *Knowledge and Information Systems*, Volume 10 , Issue 3 (October 2006) Pages: 383 - 397
- Refereed Conference Proceeding Articles
  - Dianwei Han, Guiliang Tang, and Jun Zhang, A parallel algorithm for predicting the secondary structure of polycistronic microRNAs, In *proceedings*

of Ninth IEEE International Conference on Machine Learning and Applications (ICMLA 2010), Washington, D.C., December 2010, pp. 509- 514.

– Dianwei Han, Guiliang Tang, and Jun Zhang. A novel method for microRNA secondary structure prediction using a bottom-up algorithm, In *Proceedings of the 47th Annual ACM Southeast Conference (ACMSE '09)*, Clemson, SC, USA. March 19-21, 2009, 6 pages

– Dianwei Han, Shuting Xu, and Jun Zhang. The relationship between the features of sparse matrix and the matrix solving status. In *Proceedings of the ACM Southeast Conference (ACMSE '08)*, Auburn AL, March 2008, pp. 501- 506.

– Dianwei Han, Shuting Xu, and Jun Zhang. An Online condition number query system. In *Proceedings of the ACM Southeast Conference (ACMSE '08)*, Auburn AL, March 2008, pp. 264-267.

– Dianwei Han and Jun Zhang. A Comparison of Two Algorithms for predicting the Condition Number. In *Proceedings of the Sixth International Conference on Machine Learning and Applications (ICMLA'07)*, pp. 223-228, Cincinnati, OH, 2007

– Jie wang, Jun Zhang, Lian liu, Dianwei Han. Simultaneous Pattern and Data Hiding in Unsupervised Learning. In *proceedings of ICDM Workshops 2007*, pp. 729-734, 2007. USA.

– Shuting Xu, Ju Zhang, Dianwei Han, Jie wang. Data Distortion for Privacy Protection in a Terrorism Analysis System. In *proceedings of IEEE International Conference on Intelligence and Security Informatics (ISI 2005)*, pp. 459-464, Atlanta, GA. 2005

– Dianwei Han. Problem decomposition and task distribution based on Distributed Expert System. In *proceedings of Fourth Chinese Conference on*

AI, Beijing, 1996

Teaching Experience:

- Instructor, Fall 2010. Department of Computer Science, University of Kentucky.
- Instructor, Spring 2010. Department of Computer Science, University of Kentucky.
- Teaching Assistant, 09/2007 - 12/2009. Department of Computer Science, University of Kentucky.
- Instructor, Fall 2004. Department of Electrical Engineering, Lamar University.