2013

Computational Methods For Comparative Non-coding Rna Analysis: From Structural Motif Identification To Genome-wide Functional Classification

Cuncong Zhong
University of Central Florida

COMPUTATIONAL METHODS FOR COMPARATIVE NON-CODING RNA ANALYSIS: FROM STRUCTURAL MOTIF IDENTIFICATION TO GENOME-WIDE FUNCTIONAL CLASSIFICATION

by

Cuncong Zhong
B.S. Huazhong University of Science and Technology, 2007
B.S. Huazhong University of Science and Technology, 2007
M.S. University of Central Florida, 2009

A dissertation submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in Computer Science
in the Department of Electrical Engineering and Computer Science
in the College of Engineering and Computer Science
at the University of Central Florida
Orlando, Florida

Summer Term
2013

Major Professor: Shaojie Zhang

ABSTRACT

Recent advances in biological research point out that many ribonucleic acids (RNAs) are transcribed from the genome to perform a variety of cellular functions, rather than merely acting as information carriers for protein synthesis. These RNAs are usually referred to as the non-coding RNAs (ncRNAs). The versatile regulation mechanisms and functionalities of the ncRNAs contribute to the amazing complexity of the biological system.

The ncRNAs perform their biological functions by folding into specific structures. In this case, the comparative study of the ncRNA structures is key to the inference of their molecular and cellular functions. We are especially interested in two computational problems for the comparative analysis of ncRNA structures: the alignment of ncRNA structures and their classification. Specifically, we aim to develop algorithms to align and cluster RNA structural motifs (recurrent RNA 3D fragments), as well as RNA secondary structures. Thorough understanding of RNA structural motifs will help us to disassemble the huge RNA 3D structures into functional modules, which can significantly facilitate the analysis of the detailed molecular functions. On the other hand, efficient alignment and clustering of the RNA secondary structures will provide insights for the understanding of the ncRNA expression and interaction in a genomic scale.

In this dissertation, we will present a suite of computational algorithms and software packages to solve the RNA structural motif alignment and clustering problem, as well as the RNA

secondary structure alignment and clustering problem. The summary of the contributions of this dissertation is as follows.

(1) We developed `RNAMotifScan` for comparing and searching RNA structural motifs. Recent studies have shown that RNA structural motifs play an essential role in RNA folding and interaction with other molecules. Computational identification and analysis of RNA structural motifs remain to be challenging tasks. Existing motif identification methods based on 3D structure may not properly compare motifs with high structural variations. We present a novel RNA structural alignment method for RNA structural motif identification, `RNAMotifScan`, which takes into consideration the isosteric (both canonical and non-canonical) base-pairs and multi-pairings in RNA structural motifs. The utility and accuracy of `RNAMotifScan` are demonstrated by searching for Kink-turn, C-loop, Sarcin-ricin, Reverse Kink-turn and E-loop motifs against a 23s rRNA (PDBid: 1S72), which is well characterized for the occurrences of these motifs.

(2) We improved upon `RNAMotifScan` by incorporating base-stacking information and devising a new branch-and-bound algorithm called `RNAMotifScanX`. Model-based search of RNA structural motif has been focused on finding instances with similar 3D geometry and base-pairing patterns. Although these methods have successfully identified many of the true motif instances, each of them has its own limitations and their accuracy and sensitivity can be further improved. We introduce a novel approach to model the RNA structural motifs, which incorporates both base-pairing and base-stacking information. We also develop a new algorithm to search for known motif instances with the consideration of both base-pairing and base-stacking information. Benchmarking of `RNAMotifScanX` on searching known RNA structural motifs including kink-turn, C-loop, sarcin-ricin, reverse kink-turn, and E-loop

clearly show improved performances compared to its predecessor `RNAMotifScan` and other state-of-the-art RNA structural motif search tools.

(3) We develop an RNA structural motif clustering and *de novo* identification pipeline called `RNAMSC`. RNA structural motifs are the building blocks of the complex RNA architecture. Identification of non-coding RNA structural motifs is a critical step towards understanding of their structures and functionalities. We present a clustering approach for *de novo* RNA structural motif identification. We applied our approach on a data set containing 5S, 16S and 23S rRNAs and rediscovered many known motifs including GNRA tetraloop, kink-turn, C-loop, sarcin-ricin, reverse kink-turn, hook-turn, E-loop and tandem-sheared motifs, with higher accuracy than the currently state-of-the-art clustering method. More importantly, several novel structural motif families have been revealed by our novel clustering analysis.

(4) We propose an improved RNA structural clustering pipeline that takes into account the length-dependent distribution of the structural similarity measure. We also devise a more efficient and robust CLique finding CLustering algorithm (`CLCL`), to replace the traditional hierarchical clustering approach. Benchmark of the proposed pipeline on Rfam data clearly demonstrates over 10% performance gain, when compared to a traditional hierarchical clustering pipeline. We applied this new computational pipeline to cluster the post-transcriptional control elements in fly 3'-UTR. The ncRNA elements in the 3' untranslated regions (3'-UTRs) are known to participate in the genes' post-transcriptional regulation, such as their stability, translation efficiency, and subcellular localization. Inferring co-expression patterns of the genes by clustering their 3'-UTR ncRNA elements will provide invaluable knowledge for further studies of their functionalities and interactions under specific physiological processes.

(5) We develop an ultra-efficient RNA secondary structure alignment algorithm `ERA` by using a sparse dynamic programming technique. Current advances of the next-generation sequencing technology have revealed a large number of un-annotated RNA transcripts. Comparative study of the RNA structurome is an important approach to assess the biological functionalities of these RNA transcripts. Due to the large sizes and abundance of the RNA transcripts, an efficient and accurate RNA structure-structure alignment algorithm is in urgent need to facilitate the comparative study. By using the sparse dynamic programming technique, we devised a new alignment algorithm that is as efficient as the tree-based alignment algorithms, and as accurate as the general edit-distance alignment algorithms. We implemented the new algorithm into a program called `ERA` (Efficient RNA Alignment). Benchmark results indicate that `ERA` can significantly speedup RNA structure-structure alignments compared to other state-of-the-art RNA alignment tools, while maintaining high alignment accuracy.

These novel algorithms have led to the discovery of many novel RNA structural motif instances, which have significantly deepened our understanding to the RNA molecular functions. The genome-wide clustering of ncRNA elements in fly 3'-UTR has predicted a cluster of genes that are responsible for the spermatogenesis process. More importantly, these genes are very likely to be co-regulated by their common 3'-UTR elements. We anticipate that these algorithms and the corresponding software tools will significantly promote the comparative ncRNA research in the future.

To my parents and my wife

ACKNOWLEDGMENTS

First, I would like to thank Dr. Shaojie Zhang for his persistency and encouragement during my Ph.D. studies. I would also like to thank him for his insightful direction of my research topics, wise suggestion for my research strategies, and meticulous supervision on my works. I am very grateful to Dr. Shaojie Zhang who makes my academic achievements possible.

Especially, I want to thank Dr. Kien Hua for his advising during my first year of Ph.D. studies. I also want to thank Dr. Kien Hua, Dr. Haiyan Hu, and Dr. Xiaoman Li, for serving in my thesis committee and for their inspiring suggestions and comments.

Finally, I also want to heartily thank my parents Erhao Zhong and Aiqin Zheng, and my wife Yifan Zhu for their love, support, and patience.

TABLE OF CONTENTS

LIST OF FIGURES xv

LIST OF TABLES xix

CHAPTER 1: INTRODUCTION 1

1.1 Non-coding RNAs and Their Structures 1

1.2 RNA Structural Motifs and Isosteric Base Pairs 4

1.3 Identification and Classification of RNA Structural Motifs 6

1.4 Generalized Non-coding RNA Classification for the Genome 9

1.5 Speeding Up the Genome-wide RNA Classification 11

1.6 Overview of the Dissertation 12

CHAPTER 2: SEARCHING RNA STRUCTURAL MOTIFS USING NON-CANONICAL BASEPAIRS 15

2.1 Novel Modeling Method of RNA Structural Motif 16

2.2 Materials and Methods 19

2.2.1 Base-pairing Relations in RNA Structural Motifs 21

2.2.2 Aligning two RNA Structural Motifs 21

2.2.3 *P*-value Computation 26

2.2.4 Data Processing 26

2.3 Results 27

2.3.1 Kink-turn 28

2.3.2 C-loop 30

2.3.3 Sarcin-ricin 31

2.3.4 Reverse Kink-turn 31

2.3.5 E-loop 33

2.3.6 3D Resolution Affects Identification Accuracy 35

2.3.7 Scanning PDB 35

2.4 Discussion 37

CHAPTER 3: SEARCHING RNA STRUCTURAL MOTIFS BY ADDING BASE-STACKING INFORMATION 40

3.1 Intuition of Incorporating Base-stacking Information 42

3.2 Materials and Methods 46

3.2.1 Alignment of RNA Structural Motif Graphs 47

3.2.2 Reformulation of the RNA Structural Motif Alignment Problem into a Clique Finding Problem 48

3.2.3 Identification of the Optimal Alignment Clique 51

3.2.4 Comparison of Base-stacking Interactions 52

3.2.5 *P*-value Computation 53

3.3 Results 60

3.3.1 Prioritizing the Rankings of True RNA Structural Motif Instances . . 61

3.3.2 Universal *P*-value Cutoff Towards Automatic Identification of RNA Structural Motif Instances 67

3.3.3 New Insights into the Kink-turn Motif Family 70

3.4 Discussion 74

CHAPTER 4: *DE NOVO* CLUSTERING OF RNA STRUCTURAL MOTIFS . . . 76

4.1 RNA Structural Motif Identification without Explicit Query 76

4.2 Materials and Methods 79

4.2.1 Data Preparation 79

4.2.2 Aligning Structural Components using `RNAMotifScan` 80

4.2.3 Generating Random Structural Motif Instances 81

4.2.4 Extracting Significant Clusters 82

4.3 Results 83

4.3.1 Clustering of Known Motifs and Their New Instances 85

4.3.2 Novel RNA Structural Motif Families 98

4.4 Discussion 103

CHAPTER 5: GENOME-WIDE STRUCTURAL CLUSTERING OF RNA SECONDARY STRUCTURES 104

5.1 Limitation of Clustering Analysis of Post-transcriptional Control Elements . 105

5.2 Methods 108

5.2.1 Generating Random RNA Structural Alignment Scores 108

5.2.2 Optimal Parameters Fitting 110

5.2.3 Extracting ncRNA Clusters 111

5.2.4 Rfam Data Set 115

5.2.5 *D. Melanogaster* 3'-UTR Candidate ncRNA Elements 115

5.3 Results 116

5.3.1 Benchmarking using Rfam Database 116

5.3.2 Finding ncRNA Elements in *D. melanogaster* 3'-UTR 119

5.4 Conclusions 128

CHAPTER 6: EFFICIENT ALIGNMENT OF RNA SECONDARY STRUCTURES 130

6.1 General Edit-Distance RNA Secondary Structure Alignment with Sparse Dynamic Programming 132

6.2 Methods 135

6.2.1 Preliminaries and Definitions 136

6.2.2 The Original $O(n^4 + n^2l^2)$ Algorithm 139

6.2.3 Triangular Inequality and Optimal Pair Matchings 140

6.2.4 Detection of Optimal Pair Matchings 144

6.2.5 A New Algorithm with Cubic Time Complexity 145

6.2.6 Online Pruning of Optimal Pair Matchings 149

6.2.7 Pseudo-code 153

6.3 Results 154

6.3.1 Running `LocARNA` 155

6.3.2 Time Complexity 156

6.3.3 Alignment Quality 157

6.3.4 Running Time Speedup 160

6.4 Conclusions and Discussion 162

CHAPTER 7: CONCLUSIONS AND DISCUSSION 164

7.1 RNA Structural Motif Identification 164

7.2 Genome-wide Non-coding RNA Classification 168

LIST OF REFERENCES 170

LIST OF FIGURES

2.1 Three different representations of the kink-turn motif 17

2.2 An artificial RNA structural motif containing all base-pairing relations including multi-pairing 22

2.3 Base-pairing patterns of the query motif structures in 2D diagrams 27

2.4 The superimposition of the new E-loop motif found by RNAMoitfScan, a segment of regular A-form helix, and a well characterized E-loop motif . . . 32

2.5 The 2D diagrams and 3D structures of newly identified motifs with sequence or base-pair variations 34

2.6 The Superimposition between the newly identified motifs and the queries at the regions where nucleotide insertion(s) are observed 38

3.1 An example showing that base-stacking information can distinguish structure-conserved mutations from other mutations 41

3.2 Summary of the `RNAMotifScanX` algorithm shown by aligning two artificial motif instances 45

3.3 Six relation categories that can be formed between two base interactions . . 47

3.4 The revised consensus base-interaction patterns used by `RNAMotifScanX` to search for related motif instances 52

3.5 A real example showing that the base-stacking information and the optimal alignment of crossing base pairs help to improve C-loop motif identification accuracy 62

3.6 A real example showing that the base-stacking information is capable of improving sarcin-ricin motif identification accuracy 63

3.7 A tandem kink-turn motif instance found at 1S72, chain '0', 2818-2856/2901-2930/2667-2671 69

3.8 The new kink-turn-like motif instance found at 1S72, chain '0', 1068-1075/1081-1088/1045-1046 71

4.1 The base-pairing patterns and superimposition of two GNRA tetraloop motif instances clustered in CH3 86

4.2 The base-pairing patterns and structures of the two kink-turn motif instances clustered in CL7 87

4.3 The base-pairing patterns and structures of the two kink-turn motif instances clustered in CL6 88

4.4 The base-pairing patterns, structures and superimposition of the three base pairs formed near the bulged 'G' of four sarcin-ricin motif instances clustered in CL13 91

4.5 The tandem reverse kink-turn motif instance found in 1S72, chain '0', 1515-1540/1645-1670 94

4.6 The base-pairing patterns and superimposition of the base-triple interactions of the two known hook-turn instances identified in cluster CL17 95

4.7 The base-pairing patterns and structures of two tandem-sheared instances identified in cluster CL23 97

4.8 Potential novel motif family that resembles the rope sling 98

4.9 Potential novel motif family that increases the twists at the helical region . . 100

4.10 A novel type of hexaloop motif subfamily detected by our clustering method 101

5.1 Four distributions that have been used to model the RNA structure alignment scores 109

5.2 An overview of the `CLCL` algorithm 114

5.3 Comparison of the clustering performance between `CLCL` and hierarchical clustering 117

5.4 Functional inferences of the genes clustered in C19 123

5.5 The consensus secondary structure and multiple alignments of the 3'-UTR RNA elements of all six genes that have been clustered in C37 127

6.1 Comparison between the tree-based alignment approach and the SAF-style alignment approach in handling mis-predicted base pairs 131

6.2 Illustration of the triangular inequality property 141

6.3 Pseudo-code for the implementation of the speedup techniques 154

6.4 Time complexity and OPM reduction of `ERA` 159

6.5 Alignment quality comparison of `ERA`, `LocARNA` and `RNAforester` 159

6.6 Computational efficiency comparison between `ERA`, `LocARNA` and `RNAforester` on aligning randomly selected RNA structures from Rfam 161

LIST OF TABLES

2.1 Top hits obtained by searching the five motifs against 1S72 using `RNAMotifScan` 29

2.2 The performance of `RNAMotifScan` with different resolutions of RNA structures 36

2.3 Summary of the `RNAMotifScan` search results against the entire PDB comparing with SCOR 36

3.1 Comparison between `RNAMotifScan` and `RNAMotifScanX` in identifying Kink-turn motifs from ribosomal RNA 1S72 55

3.2 Comparison between `RNAMotifScan` and `RNAMotifScanX` in identifying C-loop motifs from ribosomal RNA 1S72 56

3.3 Comparison between `RNAMotifScan` and `RNAMotifScanX` in identifying Sarcin-ricin motifs from ribosomal RNA 1S72 57

3.4 Comparison between `RNAMotifScan` and `RNAMotifScanX` in identifying Reverse kink-turn motifs from ribosomal RNA 1S72 58

3.5 Comparison between `RNAMotifScan` and `RNAMotifScanX` in identifying E-loop motifs from ribosomal RNA 1S72 59

3.6 The optimal performance of RNAMotifScan and RNAMotifScanX with a universal P-value cutoff 64

4.1 Comparison between two base-pairing pattern based clustering methods: RNAMSC (RNAMotifScan based Clustering) and LENCS 84

5.1 Detailed clustering results on *Rfam_LowID* data set 116

5.2 The expression profile of genes clustered in C19 and the consensus structure and multiple alignments of their conserved 3'-UTR RNA elements 124

5.3 Expression profile of the gene cluster C37 125

6.1 Comparison on running time of ERA, LocARNA, and RNAforester 158

CHAPTER 1: INTRODUCTION

1.1 Non-coding RNAs and Their Structures

The central dogma dictates that the genetic information of a living organism is encoded in deoxyribonucleic acid (DNA), transcribed to ribonucleic acid (RNA), and then translated into protein. The RNAs transcribed from the protein-coding genes are called messenger RNAs (mRNAs), and were once considered as the major form of RNAs in the biological system. Another important type of RNAs that have long been recognized is the transfer RNA (tRNA), which is used to carry amino acids into the ribosome for protein synthesis. As the tRNAs do not code for proteins, they are classified as non-coding RNAs (ncRNAs) to make the distinction from protein-coding RNAs. Together, the mRNAs and tRNAs are enough to complete the central dogma hypothesis, and explain the essential operation of the biological system.

However, as more complex biological systems are being studied, it is difficult to explain two major observations when we recognize the genome as primarily a collection of protein-coding genes. First, the genes that have been studied seem not enough to build up such biological systems with their amazingly high complexity. Second, the function of a large fraction (>95% in human, while much less in bacteria) of the genome is unclear, as it appears to be transcribed but not code for any proteins. Combining these two questions, it

is natural to come up with a hypothesis that the 'junk' regions of the genome are in fact very important to the very complexity of the biological system.

Indeed, recent research advances have discovered many ncRNAs with a variety of functionalities [45, 121]. Most importantly, the discoveries of these ncRNAs open a new direction for us to understand the regulation of the biological systems. For example, microRNA [14] is able to recognize its target mRNA through sequence complementarity, and direct the degradation of the mRNA. Second, the riboswitch [132] elements can alter their structures while under different physiological conditions (mostly by binding to small metabolites), and thus control the transcription or translation of their downstream genes. With the discovery of more ncRNAs, their different functionalities other than regulation are revealed (for example, catalysis (ribozyme) [40], signaling (SRP) [51], and modification (snoRNA) [88] *etc.*). While the ncRNAs are taken into consideration, a much larger fraction of the genome appears to be annotated (it is estimated that the fraction of genome that codes for ncRNAs is at least as large as that which codes for proteins [18]). The versatile functions of the ncRNAs also contribute to the deeper understanding of the biological systems.

The ultimate goal for ncRNA research is to annotate their locations in the genome, elucidate their individual functionalities, and model their interactions. In this dissertation, we approach this ultimate goal through comparative studies of ncRNA structures, as their structures are hypothesized to determine their specific functions. The structure of an ncRNA can be represented using three different levels. First, we would like to know the order of the nucleotides, with which they form the ncRNA chain. We call it the *sequence* or the *primary structure* of the ncRNA. Similar to DNA, the RNA nucleotide residues can also form strong hydrogen bonding interaction with each other through their bases, with a rule that A (adenine) binds to U (uracil) and C (cytosine) binds to G (guanine). The strong hydrogen

bonding interactions (*canonical base pairs*) determine the scaffold of the ncRNA, and we call the interaction pattern for these types of interaction the *secondary structure* of the ncRNA. Besides these two types of strong base interactions, other types of base interactions (*non-canonical base pairs*), or even interactions between the sugar rings or between bases and sugar rings, may also occur in an ncRNA. In this case, a full collection of atomic coordinates in the three-dimensional (3D) space is desired to completely represent all interactions, and we call it the *tertiary structure* of the ncRNA.

These three types of ncRNA structures facilitate the understanding of their functions on different levels. The primary structure tells the general size of the ncRNA and indicates its genomic location. The secondary structure further depicts a high-level backbone configuration of the ncRNA, and allows us to roughly infer their cellular functions. The tertiary structure contains the most complete information, and can be used to study the detailed molecular functions of the RNA and determine its specific operational mechanisms. Unfortunately, the difficulty in obtaining and analyzing these three types of structures also increases with the information contained in them. For example, to obtain the primary structure, we can directly sequence the corresponding ncRNA (e.g. the whole transcriptome can be probed using the RNA-seq technology). To obtain the secondary structure, we may apply various chemical probing methods followed by electrophoresis. However, to obtain the tertiary structure, we need to use the much more expensive experimental techniques such as X-ray diffraction or NMR (Nuclear Magnetic Resonance). Computationally, the comparison of ncRNA primary structures takes $O(l^2)$ time [98] (where l is the average length of the ncRNA sequences), while the comparison of ncRNA secondary structures takes $O(l^4)$ time [68]. The problem of comparing the ncRNA tertiary structures, however, appears to be computationally intractable.

We are interested in the comparative analysis of ncRNA secondary and tertiary structures. This is because there is little information that can be identified from the primary sequence to distinguish structured RNA from random genomic sequences. It is also observed that random sequences can fold into thermodynamically stable structures under the current RNA structure prediction rules [46]. In this case, the secondary and tertiary structure will provide us enough information to estimate the structural conservation and infer the molecular functions of the ncRNAs. Nevertheless, it is also highly desirable that we can utilize the knowledge learned from studying ncRNA secondary and tertiary structures to improve the ncRNA structure prediction rules, and identify ncRNA genes from genomic sequences with a higher accuracy.

1.2 RNA Structural Motifs and Isosteric Base Pairs

Despite the importance in analyzing RNA tertiary structures, computational estimation of their structural similarity is still an open problem. First of all, as shown by Jiang *et al.* [68], the comparison of RNA tertiary structures (or even RNA secondary structure with crossing base pairs) is unlikely to be solvable within polynomial time. Second, the ncRNA structures have different structural flexibility in different domains, i.e. the functional domains require high structural conservation to maintain their proper functions. However, such information is usually unavailable unless the ncRNA structure is well annotated. The different structural flexibility limits the application of the generalized (without assigning weight to specific regions of interest) RNA tertiary structure comparison method. As a result, manual inspection is still the most popular and reliable approach in analyzing ncRNA tertiary structures.

However, it is very difficult for a human to analyze large RNA structures, such as the large ribosomal RNAs subunits. It is highly desirable that we can study the large RNA structures in a modulated manner. In other words, it would greatly improve ncRNA tertiary structure analysis if we can decompose the ncRNA structure into recurrent fragments, while these fragments have relatively rigid 3D structures and predictable functionalities. These structural fragments are referred to as the *RNA structural motifs* [62, 78, 95]. RNA structural motifs are usually small in size (5 - 20 bps) and found in the junction regions between regular A-form helices [37]. Their functionalities are still not fully understood; however, current knowledge indicates that they are either ultra thermodynamically stable, or are required for the inter or intra molecular interactions (between itself and other DNA, RNA, proteins, or small metabolite molecules) that are associated with the ncRNA.

Even we have now reduced our problem of analyzing the entire RNA tertiary structure into the study of the much smaller RNA structural motifs, there exist potential drawbacks in direct comparisons of the RNA tertiary structures based on their 3D coordinates. Specifically, we must consider potential structural variations in the RNA structural motifs which are functionally conserved. Therefore, comparing a full set of atomic coordinates of the RNA structural motifs is not theoretically sound. Existing computational methods for RNA tertiary structure comparison tackle this problem by representing the RNA 3D structure with means of abstractions. For example, representing a nucleotide using its key atoms [59, 113] or representing the overall structure using its backbone trajectory [44], are used by a variety of 3D geometry-based RNA structural motif analysis methods to consider potential structural variations. Apparently, this research progress has greatly relieved the computational burden and led to numerous significant discoveries about ncRNA tertiary structures. However, it is still questionable whether the abstraction methods that have been used here are the best

way to model ncRNA structures. And the identification performances of these methods, when compared to the manual analysis results, need further improvements.

An alternative approach for representing an RNA structural motif is to use its base-pairing pattern that includes non-canonical base pairs [79, 80]. The non-canonical base pairs are base-pairing interactions other than the canonical Watson-Crick base pairs. The non-canonical base pairs are summarized into different *isosteric groups* based on their C1'-C1' distance, where the base-pair substitutions within the same isosteric group are considered to be structurally conserved [80]. The summarization of isosteric base pairs opens a new direction for analyzing RNA structural motifs. First, it provides a natural way for us to model the RNA structural motifs using their base-pairing patterns (including both canonical and non-canonical base pairs), which reduces computational complexity significantly and takes into account potential structural variations. Second, the isosteric group serves as a theoretical foundation for us to derive *ad hoc* scoring functions, which can help to determine whether the base-pair substitutions are structurally conserved.

1.3 Identification and Classification of RNA Structural Motifs

A direct application of comparative analysis of the RNA structural motifs is the prediction of its occurrences in a given RNA structure of interest. Such information, when coupled with the molecular function of each structural motif, can provide invaluable insight for us to understand the RNA structures of interest. We refer to this problem as the *RNA structural motif identification* problem. We are also interested in the structural classification of the RNA structural motifs, especially for the purpose of *de novo* discovery of new RNA structural

motif families. We refer to this problem as the *RNA structural motif classification* problem. What lies in the center of these two problems is the comparison (or *alignment*) of RNA structural motifs. For the RNA structural motif identification problem, all candidate regions are compared with the query model (usually the consensus structure), and the high-score hits are reported as potential instances. For the RNA structural motif classification problem, all-against-all alignments are performed on the candidate motif instances, and subsequently a clustering algorithm is applied to define groups of closely related motif instances. In both cases, alignments of the RNA structural motifs (specifically, their base-pairing patterns) become the central problem for RNA 3D structure analysis.

The base-pairing pattern of a given RNA structural motif can be summarized into a graph, where the vertices in the graph correspond to the nucleotides in the RNA structural motif, and the edges indicate the base-pair interactions. The edges in the graph can be labeled to indicate which isosteric group the base-pair interaction belongs to. Because the graph isomorphism is a computationally hard problem, the naive comparison of RNA structural motif graphs is computationally demanding. However, due to their limited sizes, it would be feasible to devise a branch-and-bound algorithm to optimally align RNA structural motif graphs. These observations leave us with two options: either reduce the RNA structural motif base-pairing patterns into planer graphs such that they are solvable using polynomial time solutions, or develop a branch-and-bond algorithm to compute more accurate alignments with a higher computational overhead. The choice between these two strategies should be made according to the purpose of the study.

When we only focus on the base-pair patterns (both canonical and non-canonical), it is observed that the majority of the base pairs are nested, and only a small number of them cross with each other. In this case, if we temporarily remove the crossing base pairs in the

motif instances, the remaining base pairs become fully nested, and then the motif instances can be compared in polynomial time. Because the majority of the base pairs will remain, it is expected that the corresponding alignment will be generated with satisfying quality. After producing the alignments with nested base pairs, the crossing base pairs can be added back to potentially recover alignment errors. To compare two RNA structures without crossing, the algorithm framework of `RNAscf` [10] is borrowed (modified by incorporating an *ad hoc* scoring function that describes base-pair isostericity), which can run in $O(l^4)$ time. The strategy is implemented into an RNA structural motif search tool named `RNAMotifScan`. `RNAMotifScan` can align RNA structural motif instances with high computational efficiency, and therefore is suitable for large-scale analysis such as database search or clustering analysis. The details of this work will be discussed in Chapter 2.

However, although `RNAMotifScan` has shown significant improvement over the other 3D geometry-based methods in RNA structural motif identification accuracy, its performance can still be further improved. First, `RNAMotifScan` only considers base-pairing information, while another important type of base interaction, the base-stacking interaction, is completely ignored. Second, the heuristic assumption made on crossing base pairs does not always correspond to the optimal scenario, and will sometimes lead to incorrect alignments. Therefore, we incorporate the base-stacking information and consider both base-stacking and base-pairing interactions when modeling the RNA structural motif. An optimal solution for aligning motif instances with crossing interactions is necessary, because there will be many more crossing interactions in the motif instances when base-stacking information is considered. Note that an exponential solution is feasible as long as the algorithm is elegantly developed to efficiently prune the search space. This algorithm has been implemented into a program called `RNAMotifScanX`. `RNAMotifScanX` can align RNA structural motif instances with high

accuracy, and therefore is suitable for detailed study of a few RNA structures. The details related to this work will be discussed in Chapter 3.

Finally, empowered by these RNA structural motif comparison tools, we are able to cluster the RNA structural motifs in ribosomal RNAs (including 5S, 16S, and 23S). Our major goal for this study is to define a structural classification for the structural motif instances, and at the same time identify novel motif families. Because the clustering analysis requires all-against-all comparison of the motif instances, it is desirable to use a more efficient version of the RNA structural motif comparison tool, i.e. `RNAMotifScan`. Correspondingly, we have developed a clustering pipeline for RNA structural motif clustering called `RNAMSC` (`RNAMotifScan`-based Clustering). By applying such clustering pipeline to candidate motif instances from the ribosomal RNAs, we have been able to identify many new occurrence of the known motif families, and more importantly, two completely novel motif families. The details for the design of the clustering pipeline and the biological discoveries will be discussed in Chapter 4.

1.4 Generalized Non-coding RNA Classification for the Genome

After developing the clustering pipeline for RNA structural motif instances, it is also desirable to apply it to genome-wide ncRNA classification. To make this step forward, we have to solve three central problems that are specific to the genome-wide ncRNA classification. First, the identification of ncRNAs from the genome is itself a difficult problem, as the information contained in the sequence alone is insufficient for accurate prediction of ncRNA genes [46]. Second, the comparison between ncRNA structures, although can be finished in a polynomial

time $O(l^4)$, is still of relatively high complexity when performing all-against-all alignments on candidate ncRNA elements from the entire genome. Third, as the majority of the ncRNAs from the genome are not yet annotated, there is little inference we can make, even if we had discovered any interesting structure clusters. To solve these problems, we decided to focus on ncRNA elements from the 3'-UTR of the fly (*Drosophila melanogaster*) genome.

Focusing on clustering the post-transcriptional control ncRNA elements from the 3'-UTR of the fly genome solves these three problems. Biologically, it is more likely that the ncRNA elements will reside in the untranslated regions (UTR) instead of the protein coding regions [67]. Computationally, such an observation indicates that the candidates discovered in the UTR are more likely to be real ncRNA elements. Thus, this strategy indirectly improves the *de novo* ncRNA gene prediction accuracy, and, to a certain degree, solves the first problem. It is also clear that restricting our study focus to subregions of the genome will reduce the number of candidate ncRNAs, and relieve the computational burden for the all-against-all structural alignment step. In this case, this strategy solves the second problem. At last, as we have associated these ncRNA elements with their upstream protein coding regions, this strategy solves the third problem by allowing functional inference of potential ncRNA clusters by referring to their upstream genes' functions.

Finally, a key technical challenge for the ncRNA clustering problem is how to estimate the cutoff to define the individual clusters. It is well known that the raw RNA structure alignment scores are length biased, i.e. larger RNA structures tend to result in higher alignment scores. In this case, the raw alignment score cannot be used as a direct measure for the structure similarity, and it must be normalized before being used. To accomplish this task, we devised a simulation-based statistical framework to estimate the p-values for the structure alignment scores. We randomly generate a large number of RNA sequences by

preserving the original di-nucleotide frequency of a given RNA template, and then predict their secondary structures using `RNAfold` [66]. These random structures are subject to alignment with the template RNA structure, and the corresponding alignment scores are taken as the background scores distribution for the template RNA structure. Based on this score normalization strategy, we further design a more accurate and robust CLique CLustering (`CLCL`) algorithm (compared to the hierarchical clustering algorithm), which predicts ncRNA clusters from the normalized p-values for alignment scores. The entire clustering pipeline is applied to the candidate ncRNA elements predicted (using `RNAz` [137]) from the fly 3'-UTR. The details for the design of the clustering pipeline and corresponding biological discoveries will be discussed in Chapter 5.

1.5 Speeding Up the Genome-wide RNA Classification

One of the potential problem of the previously discussed `CLCL` pipeline is that the time to to align candidate structures is too slow for analyzing long ncRNAs or large data sets. As more long ncRNAs are being discovered [140], a faster alignment tool for their alignments is in urgent demand. In addition, RNA structure chemical probing experiments have been coupled with the next-generation sequencing (NGS) technology to predict accurate RNA secondary structures in a high-throughput manner [72, 85, 134]. Specifically, our major objective is to devise a novel RNA secondary structure alignment algorithm, which can run more efficiently and at the same time produce high-quality alignments (not heuristics).

We adopt the idea of sparse dynamic programming technique to solve this problem. The sparse dynamic programming is a technique that aims to prune the search space of the

algorithm by exploring the triangular inequality of the scoring functions. Once a scenario is determined to be suboptimal, it will be marked for deletion and such a scenario will not be considered in the future. The sparse dynamic programming technique has been applied in many problems that are related with RNA structure analysis, including RNA structure folding, co-folding, and RNA-RNA interaction. In this work, our goal is to incorporate the sparse dynamic programming technique into RNA structure alignment algorithm. By careful redesign of the algorithm and the incorporation of a new online pruning technique, the resulting new algorithm `ERA` is capable of speeding up the RNA structure alignment by approximately 5 - 100 fold, with an average speedup of 10 fold. Meanwhile, benchmark results show that the alignment quality of `ERA` is as good as the one with a guaranteed optimal solution. The details of the algorithm design and benchmark experiments will be discussed in Chapter 6.

1.6 Overview of the Dissertation

In summary, we have developed a suite of computational methods for the central problems of RNA structure analysis and functional inference: the alignments and classification of the RNA structures. We have developed computational methods to analyze both 3D RNA structural motifs and general RNA secondary structures. We have also developed a clustering pipeline that integrates our alignment tool for *de novo* RNA structural motif discovery and genome-wide RNA secondary structure survey. Specifically, Chapters 2 - 6 of this dissertation will be dedicated to the discussion of these computational methods. The brief overviews of these chapters are listed as follows.

In Chapter 2, we will describe `RNAMotifScan`, an alignment algorithm to compare 3D RNA structural motifs using the comparison of their base-pairing patterns which include non-canonical base pairs and their isostericty. Chapter 2, in part, is a reprint of the article, 'RNAMotifScan: Automatic Identification of RNA Structural Motifs using Secondary Structural Alignment', co-authored with Haixu Tang and Shaojie Zhang in *Nucleic Acids Research*, 38 (18), pp 1−11.

In Chapter 3, we will describe `RNAMotifScanX`, an improved version of `RNAMotifScan` due to its consideration of the base-stacking information and its new branch-and-bound algorithm that is able to handle crossing base interactions. Chapter 3, in part, is a reprint of the manuscript, 'RNA structural motif identification through incorporating base-stacking information', co-authored with Shaojie Zhang.

In Chapter 4, we will describe `RNAMSC`, a clustering pipeline for analyzing RNA structural motifs based on the motif alignment tool `RNAMotifScan`. We will also present its application to clustering RNA structural motifs from ribosomal RNAs. Chapter 4, in part, is a reprint of the article, 'Clustering RNA structural motifs in ribosomal RNAs using secondary structural alignment', co-authored with Shaojie Zhang in *Nucleic Acids Research*, 40 (3), pp 1307−1317.

In Chapter 5, we will describe `CLCL`, a clustering pipeline for genome-wide classification of RNA secondary structures. We will also present its application to the 3'-UTR of the *D. melanogaster* genome. Chapter 5, in part, is a reprint of the article, "Discovering non-coding RNA elements in Drosophila 3' un-translated regions", co-authored with Justen Andrews and Shaojie Zhang in *IEEE International Conference on Computational Advances in Bio and Medical Sciences*, 2012, Feb 23−25, Las Vegas, Nevada, USA, 2012, and is also a reprint

of the manuscript, "Discovering non-coding RNA elements in Drosophila 3' un-translated regions", accepted by *BMC Genomics*.

In Chapter 6, we will describe `ERA`, an efficient RNA secondary structure alignment algorithm developed using a sparse dynamic programming technique. Chapter 6, in part, is a reprint of the manuscript, 'Efficient alignment of RNA secondary structures using sparse dynamic programming', co-authored with Shaojie Zhang.

All of the computational tools (except the computational pipelines `RNAMSC` and `CLCL`, which are presented descriptively) will be made publicly available at the supporting website upon the publication of the corresponding manuscripts (`http://genome.ucf.edu`). We anticipate that the computational methods developed by us will significantly promote and improve RNA structural analysis and functional inference in the future.

CHAPTER 2: SEARCHING RNA STRUCTURAL MOTIFS USING NON-CANONICAL BASEPAIRS

Non-coding RNAs (ncRNAs) play important functional roles in the biological system, and recent discoveries point to many of their novel cellular functions [45, 121]. The cellular functions of the ncRNAs are determined by their specific structures. Unlike DNAs, which usually exhibit regular double helical structures due to the interactions with the complementary strands, RNAs are single strand molecules and can fold into irregular three dimensional (3D) structures. Among the complex structures, there exist many conserved and recurrent segments whose arrangement, abundance and interaction largely determine the folding behaviors and functionalities of the RNA structures. These segments, viewed as the 'building blocks' of RNA architecture, are usually referred to as RNA structural motifs [62, 78, 95]. The identification and analysis of these RNA structural motifs will significantly deepen our understanding of ncRNA structures and help us to elucidate the structure-function relationship.

In this chapter, we set our focus on developing algorithms to align two RNA structural motif instances. By using the non-canonical base pairs and their associated isostericty, we develop a new RNA structural motif alignment and search tool named `RNAMotifScan`. The new tool is benchmarked against other three state-of-the-art RNA structural motif identification tools. The benchmark results clearly show the improvements of `RNAMotifScan` in terms of both accuracy and sensitivity.

2.1 Novel Modeling Method of RNA Structural Motif

The common approach for RNA structural motif identification is to represent the RNA structural motifs by different 3D properties (i.e., torsion angles or atomic distances) of the key nucleotides and then apply heuristics to searching for the topological occurrences of the motif in the 3D RNA structures (similar to the methods for 3D protein structure comparison [3]). Computer program, such as `PRIMOS` [44] and `COMPADRES` [136], represents and searches certain backbone conformations using pseudotorsion angles. On the other hand, `NASSAM` encodes the 3D motif by using a graph to store pairwise atomic distances between key nucleotides [59]. To reduce the information contained in pairwise atomic distances, `ARTS` builds approximated anchors based on a set of seed points before detailed matching [42]. Recent progress uses shape histograms, which are also computed from pairwise atomic distances, to summarize the structural motifs [6]. This method has identified the occurrences of many structural motifs in ribosomal RNAs [112]. Instead of considering solely torsion angles or atomic distances, `FR3D`, which searches for recurrent motifs considering a combination of geometric, symbolic, and sequence information, achieves the most satisfying performance [113]. Although the existing methods have successfully identified many occurrences of several known RNA structural motifs, most of them require the accurate 3D coordinates of the query motif, and thus are limited to structural motifs with rigid 3D topologies. However, it is known that many motifs exhibit certain structural variation and thus cannot be well characterized by their 3D topologies [83]. Therefore, the more conserved base-pairing pattern should be considered when searching for RNA structural motifs [87, 101].

It was observed that many non-canonical base-pairs in RNA structural motifs are *isosteric* and these base-pairs can interchange with each other without affecting the overall RNA

structure [80]. Generally, a base-pair should have three properties: (a) the two nucleotides interacting through hydrogen bonds; (b) nucleotide *edges* participating in the interaction; and (c) the relative orientation of the glycosidic bonds which is either '*cis*' or '*trans*'. Each nucleotide has three edges that can interact with another nucleotide to form a base-pair, namely the Watson-Crick edge (denoted as 'WC' edge), Hoogsteen edge (denoted as 'H' edge) and Sugar edge (denoted as 'SE' edge). Given the three properties, it is sufficient to classify all base-pairs into one of the isosteric groups [80]. Modeling RNA structural motifs through non-canonical base-pairs is theoretically sound and can largely reduce the complexity of 3D RNA motifs. First, the definition of isostericity serves as the foundation of relating tertiary structure with non-canonical base-pairs. Second, some motifs are defined by their characterized non-canonical base-pairing patterns, instead of their 3D structures. Finally, modeling RNA structural motifs by their base-pairing pattern is easier to understand comparing to by their atomic coordinates.

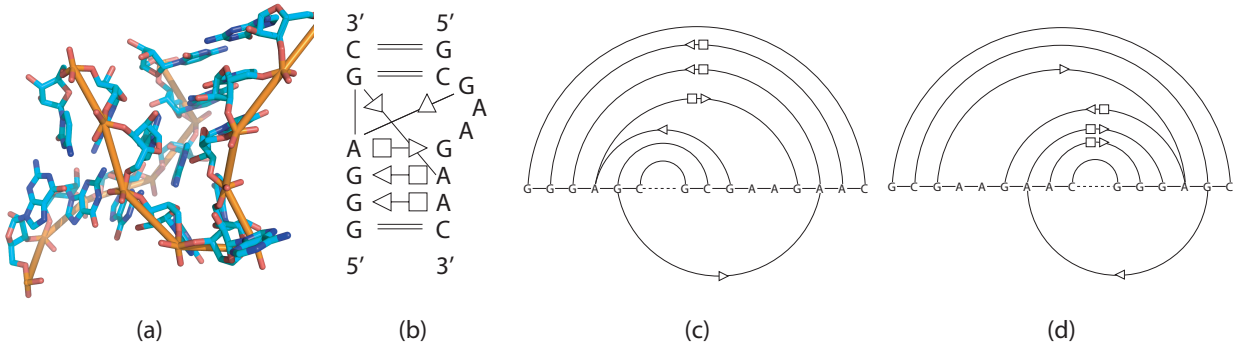

Figure 2.1: Three different representations of the kink-turn motif. (a) 3D structure. (b) 2D diagram for base-pairing patterns (notation is the same as proposed in [82]). (c) and (d) Arc representations built by concatenating the two strands of the motif with two different orders. For (c) and (d), the arcs rest above on the horizontal line represents the base-pairs that are optimally aligned in the first step, while the arcs below are processed in the second step. The motif is from a 23S rRNA in *H. marismortui* (1S72, chain '0', location 77-82/92-100).

Djelloul and Denise [37] modeled the RNA structural motifs through graph representation of these non-canonical base-pairs. They extracted structural segments containing non-canonical base-pairs from the annotated RNA 3D structure. By constructing clusters through the measurement of pairwise maximum isomorphic base-pairing cores, they characterized the recurrent base-pairing patterns among these structural segments. This method has led to the rediscovery of many structural motifs, which shows the potential power of utilization of non-canonical base-pairs in modeling RNA structural motifs. However, this method is not optimized for structural motif identification, for the isomorphic condition is not suitable to identify the motifs that exhibit variations in non-canonical base-pairs.

Therefore, well developed algorithms for comparing the non-canonical base-pair patterns between two RNA tertiary structural segments are in urgent demand. However, most existing methods model and compare RNA structures only through canonical base-pairs. In a typical approach, free energy values are assigned to the canonical base-pairs, and secondary structure with minimum free energy are computed to model the structure [64, 128, 139, 154, 155]. Comparative genomics approaches aim at the identification of *consensus* canonical base-pairs from a set of synthetic genomic sequences of multiple species that are previously aligned [102, 137] or even unaligned [10, 21, 34, 55]. The RNA homolog search approaches attempt to find genome sequences that match a query RNA in sequence and a model secondary structure annotated with canonical base-pairs [76, 149, 150]. RNA canonical base-pairs are also modeled into tree structures, and the edit distance between two tree structures is then computed [63, 68]. Recently, variants of Sankoff's algorithm [111] are also used to compare the canonical base-pairs between two RNA structures [129, 142].

These computational methods can be extended to comparing RNA structures with non-canonical base-pairs. We need to address the following issues raised by the inclusion of

non-canonical base-pairs. Most importantly, the similarity between two non-canonical base-pairs should be measured. The reason is that canonical base-pairs can interchange with each other while maintaining the tertiary structure, but such possibility is not guaranteed for non-canonical base-pairs as defined in the isosteric matrices. In addition, canonical base-pairs are usually nested stacked in forming the A-form helical regions, while RNA structural motifs usually include many multi-pairings (interactions involves more than two nucleotide residues, i.e base-triples) and pseudoknots (crossing base-pairs), see Figure 2.3. Therefore, non-canonical base-pairs, multi-pairing and crossing base-pairs must be handled in order to properly compare the structural motifs.

2.2 Materials and Methods

The query RNA structural motif base-pairing patterns are adopted from related publications (see Data processing Section). We concatenate two strands of the query RNA motif into one sequence for the alignment (see Figure 2.1 (c) and (d), there are two ways to concatenate the query and both are searched against the target). For the target RNA segments, we first use annotation software (see Data processing Section) to translate the RNA 3D coordinates into base-pair patterns that contain sufficient information for isosteric group classification (i.e. pairing nucleotides, interacting edges, and relative glycosidic bond orientations). We then cut the annotated target RNA structure into many local (interactions within two strands, long range interactions are ignored) RNA structural segments. Similarly, we concatenate two strands of the target RNA structural segments into one sequence. To identify RNA motif instances, we use a dynamic programming procedure to compute the

similarity between the query RNA motif and all structural segments in the target RNA and report the significant hits.

The recursive functions of the alignment procedure need to address three major issues. First, the isostericity of the base-pairs should be incorporated into the scoring functions such that only base-pairs belong to the same isosteric group [80] can be matched to each other. Second, there are many multi-pairings occurring in the RNA structural motif and the target RNA, which is introduced by one nucleotide simultaneously paired with two or more other nucleotides. This can be observed since each nucleotide has three edges, thus the nucleotide is able to participate in at most three base-pairs. We discuss the multi-pairing issue in the next section for the alignment procedure. Finally, both the query RNA motif and the target RNA segments may contain crossing base-pairs.

We divide the alignment into two steps. We first align non-crossing base-pairs in the query. (Crossing base-pairs in query are removed temporarily and processed in the second step, while the crossing base-pairs in target structure are retained.) We then try to reinsert the removed crossing base-pairs based on the resulting alignment. Note that we select the minimum number of base-pairs to be matched in the second step so that most of the base-pairs can be aligned optimally in the first step. Because the structural motifs are likely to be well represented by its major part of nested base-pairs, which are matched optimally, it should work in most practical cases. Also, users can select the base-pairs to form the query motif for the first step searching.

2.2.1 Base-pairing Relations in RNA Structural Motifs

Multi-pairings are not only frequently occurred, but also important in forming the RNA structural motifs. Here we formally define the classifications and relations of base-parings including multi-pairings. We denote the indices of the left and right nucleotides of a base-pair P as P_l, P_r. Generally, two base-pairs P^A and $P^{A'}$ may have one of the following relations: (1) P^A and $P^{A'}$ are interleaving; (2) $P^{A'}$ is enclosed with P^A (denoted by $P^{A'} <_I P^A$); (3) $P^{A'}$ is juxtapose to P^A and before P^A (denoted by $P^{A'} <_p P^A$). Specifically, RNA structural motifs may contain multi-pairings. To handle these situations, we need to redefine the above definition. We extend the enclosing relation ($<_I$) to three subgroups (see Figure 2.2 (c)): $P^{A'} <_{I_1} P^A$ ($P_l^A < P_l^{A'} < P_r^{A'} < P_r^A$), $P^{A'} <_{I_2} P^A$ ($P_l^A = P_l^{A'} < P_r^{A'} < P_r^A$) and $P^{A'} <_{I_3} P^A$ ($P_l^A < P_l^{A'} < P_r^{A'} = P_r^A$). We also extend the juxtaposing relation ($<_p$) to two subgroups (see Figure 2.2 (d)): $P^{A'} <_{p_1} P^A$ ($P_l^{A'} < P_r^{A'} < P_l^A < P_r^A$) and $P^{A'} <_{p_2} P^A$ ($P_l^{A'} < P_r^{A'} = P_l^A < P_r^A$).

2.2.2 Aligning two RNA Structural Motifs

We can use a dynamic programming algorithm to compute an optimal alignment between two RNA structural segments [10]. There are three major contributions in this algorithm. First, the dynamic programming algorithm is guided by the partial order base-pairs. Second, we consider non-canonical base-pairs and their isostericity. Finally, we also allow non-crossing multi-pairings for the query and target structure.

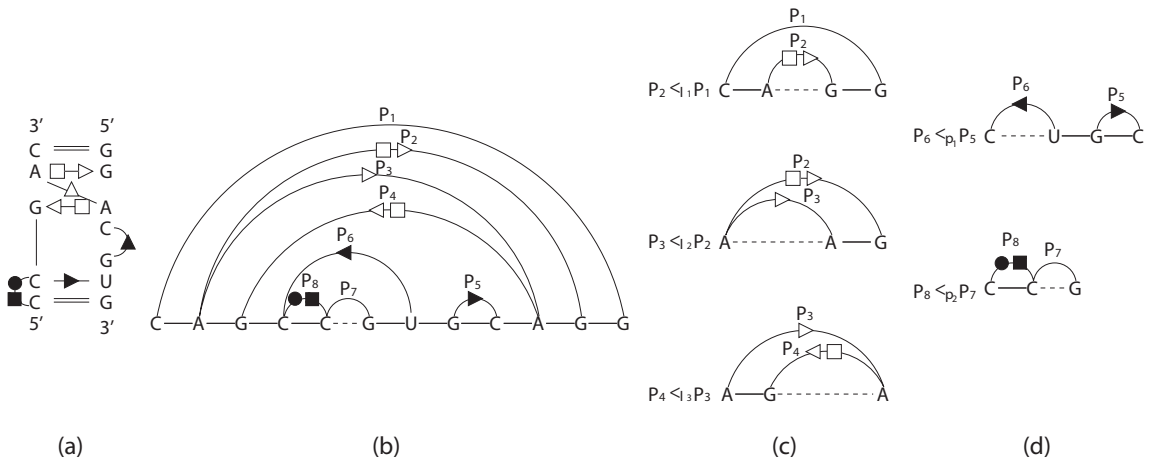

Figure 2.2: An artificial RNA structural motif containing all base-pairing relations including multi-pairing. (a) The base-pairing pattern of the motif. (b) The arc representation of the motif. (c) Base-pairs relation subgroups in the motif belong to enclosing relation. (d) Base-pairs relation subgroups in the motif belong to the juxtaposing relation.

Given an RNA structural motif A and a target RNA structural segment B with concatenated strands and m and n base-pairs respectively. Dummy base-pairs were added between nucleotides $A[0]$ and $A[|A|+1]$ and between nucleotides $B[0]$ and $B[|B|+1]$. Let $\mathcal{P}^A = P_1^A, P_2^A, ..., P_m^A$ and $\mathcal{P}^B = P_1^B, P_2^B, ..., P_n^B$ denote the two sets of base-pairs, ordered according to increasing values of the right-most base. Define the following terms:

1. Seq(P^A): The two nucleotides that form the base-pair P^A, given by $A[P_l^A]$ and $A[P_r^A]$.

2. Loop(P^A): The sub-sequence covered by the two nucleotides of the base-pair P^A excluding the two nucleotides themselves. In other words, the sequence $A[P_l^A+1 \ldots P_r^A-1]$.

3. Loop($P^A, P^{A'}$): The term is defined if and only if $P^{A'}$ is completely juxtaposing to the left of P^A, as the loop region corresponding to $A[P_r^{A'}+1 \ldots P_l^A-1]$.

The score of the optimal alignment between two RNA sequences consists of three parts: the score of matching base-pairs, the score of matching paired bases, and the score of matching unpaired subsequences (including gaps). These scores are assigned with different weights (w_1, w_2 and w_3, respectively) to distinguish the importance of them in building an RNA motif. Define the following terms:

1. $\mathcal{I}(P^A, P^B)$: The matching score between two base-pairs P^A and P^B. The score is evaluated by the isostericity between two P^A and P^B. Base-pairs within the same isostericity group are considered to have similar structural contribution to the motifs, and their matching is given higher bonus score. Non-isosteric matching is also allowed, but with less bonus score.

2. $\mathcal{S}(A[i...j], B[k...l])$: The matching score between two subsequences $A[i...j]$ and $B[k...l]$. The score is evaluated through the optimal global alignment between the two subsequences.

3. $Gap(k)$: The gap penalty of inserting/deleting a sequence of length k.

4. $M[P^A, P^B]$: The score of the optimal alignment of the regions enclosed by base-pairs P^A and P^B, given that P^A and P^B are aligned to each other. Entry $M[P^A_m, P^B_n]$ records the score of the optimal alignment between two structures A and B.

All the weights and scores defined above are fixed for all searches conducted in this work.

We can compute $M[P^A, P^B]$ for all pairs in $\mathcal{P}^A \times \mathcal{P}^B$, which would take $O(m^2n^2)$ time, where m and n are the number of base-pairs in A and B, respectively. While many RNA structural alignment algorithms have biquadratic time complexity in terms of sequence length, our

algorithm is relatively efficient since the number of base-pairs in an RNA structure is much smaller than its length in sequence. In computing $M[P^A, P^B]$, we have two choices for matching the subsequences inside P^A and P^B, as they could either form consensus hairpin loops (the terminal case) or there are base-pairs to be matched inside (nested base-pairs, internal loop, or multi-loop). Therefore,

$$M[P^A, P^B] = M_s[P^A, P^B] + \max \begin{cases} M_h[P^A, P^B], \\ M_l[P^A, P^B]. \end{cases} \tag{2.1}$$

Here, $M_s[P^A, P^B]$ is the score of matching base-pairs P^A and P^B based on both structure isostericity and sequence conservation, and thus can be computed by:

$$M_s[P^A, P^B] = w_1 \mathcal{I} \begin{pmatrix} P^A, \\ P^B \end{pmatrix} + w_2 \mathcal{S} \begin{pmatrix} \text{Seq}(P^A), \\ \text{Seq}(P^B) \end{pmatrix}. \tag{2.2}$$

$M_h[P^A, P^B]$ is the score of matching the loop regions of P^A and P^B, assuming that no consensus base-pair is included by P^A and P^B. (For example, these regions form matched hairpin loops.) It can be computed by:

$$M_h[P^A, P^B] = w_3 \mathcal{S} \begin{pmatrix} \text{Loop}(P^A), \\ \text{Loop}(P^B) \end{pmatrix}. \tag{2.3}$$

For the nested base-pairs, internal loop, or multi-loop case, we need to define some additional terms. A sequence of base-pairs $P_1, P_2, \ldots, P_k$ form a *chain* if $P_1 <_p P_2 <_p \ldots <_p P_k$. $M_l[P^A, P^B]$ represents the matching score between P^A, and P^B, given that there is a pair of chains included by P^A and P^B which form the loop. Let $P_1^A, P_2^A, \ldots$ ($P_1^B, P_2^B, \ldots$, respectively) denote base-pairs enclosed by P^A (P^B, respectively), and ordered according to

increasing values of the last coordinate. For two base-pairs $P^{A'}$, P^A that $P^{A'} <_I P^A$, Loop(P^A) is separated into three major regions: left region, Loop($P^{A'}$) and right region. We denote the left region as LoopL($P^A, P^{A'}$) ($A[P_l^A + 1 \dots P_l^{A'} - 1]$) and the right region as LoopR($P^A, P^{A'}$) ($A[P_r^{A'} + 1 \dots P_r^A - 1]$). Then, we will have

$$M_l[P^A, P^B] = \max_{i,j} \left\{ M_c[P_i^A, P_j^B] + w_3 \mathcal{S} \left(\begin{array}{l} \text{LoopR}(P_i^A, P^A), \\ \text{LoopR}(P_j^B, P^B) \end{array} \right) \right\}. \tag{2.4}$$

To enforce the matched base-pairs have the same multi-pairing pattern, we must ensure that P_i^A and P^A, P_j^B and P^B are in the same enclosing subgroup ($<_{I_1}, <_{I_2}$, or $<_{I_3}$, see Figure 2). Here, $M_c[P_i^A, P_j^B]$ is defined as the score of two chains of the optimal matching configurations that end at P_i^A, and P_j^B, and begin at some $P_{i'}^A <_p P_i^A$, and $P_{j'}^B <_p P_j^B$. Denote $P_{i_1}^A \in F(P_{i_2}^A)$ if $P_{i_1}^A <_p P_{i_2}^A$ and there is no base-pair P_j^A such that $P_{i_1}^A <_p P_j^A <_p P_{i_2}^A$. Then,

$$M_c[P_i^A, P_j^B] = \max_{\substack{P_x^A \in F(P_i^A) \\ P_y^B \in F(P_j^B)}} \left\{ \begin{array}{l} w_3 \mathcal{S} \left(\begin{array}{l} \text{LoopL}(P_i^A, P^A), \\ \text{LoopL}(P_j^B, P^B) \end{array} \right), \\ M_c[P_x^A, P_y^B] + M[P_i^A, P_j^B] + w_3 \mathcal{S} \left(\begin{array}{l} \text{Loop}(P_x^A, P_i^A), \\ \text{Loop}(P_y^B, P_j^B) \end{array} \right), \\ M_c[P_i^A, P_y^B] + w_3 Gap(|\text{Loop}(P_y^B, P_j^B)| + |\text{Loop}(P_j^B)|), \\ M_c[P_x^A, P_j^B] + w_3 Gap(|\text{Loop}(P_x^A, P_i^A)| + |\text{Loop}(P_i^A)|). \end{array} \right. \tag{2.5}$$

The *Gap* means the corresponding sequences are matched to nothing (i.e., they are deleted). Similarly, to enforce the matched base-pairs have the same multi-pairing constraint, we must ensure that P_x^A and P^A, P_y^B and P^B are in the same enclosing subgroup, and P_x^A and P_i^A, P_y^B and P_j^B are in the same juxtaposing subgroup.

2.2.3 P-value Computation

To compute the p-value for the probability that an RNA motif hits a random substructure in the database, we used the non-parametric Chebyshev's inequality. In future research, we will optimize these parameters by fitting the distribution of the overall alignment scores between pairs of RNA structures into a Gumbel-like distribution to get more accurate p-value. To obtain the mean and variance, the query is aligned against the background segments, which are generated by randomly picking base-pairs from real RNA structures while maintaining the similar GC content, as well as frequencies of the interacting edges and glycosidic bonds orientations. We applied this approach on Kink-turn motif, and observed Gumbel's distribution of the alignment scores (see supplementary website). Since each motif has its own base-pairing patterns and degree of tolerance against base-pair variations, we suggest different p-value cutoffs for different motifs based on tested results (see Table 2.3 for the cutoffs). Additionally, False Positive Rates (FPR) are computed through simulation and available on the supplementary website.

2.2.4 Data Processing

Base-pair interactions of all RNA 3D structures from PDB [17] (released on August 2008) were first annotated by using `MC-Annotate` [53]. `RNAVIEW` [147] generates similar results based on our experiments, and `RNAMotifScan` provides interfaces for both annotation tools. After annotation, 1445 RNA structures were generated from PDB (including incomplete RNA chains in the raw PDB file). Five RNA structural motifs were used as queries to

test our method: the Kink-turn, C-loop, Sarcin-ricin, Reverse Kink-turn and E-loop motifs. Because they are well characterized, documented and important for many RNA folding behaviors or functionalities. The query base-pairing patterns for these motifs come from the following references: Kink-turn [84], C-loop [83], Sarcin-ricin [79], Reverse Kink-turn [78] and E-loop [83]. The two dimensional (2D) diagrams for query base-pairing patterns of these motifs are shown in Figure 2.3. `RNAMotifScan` was implemented in ANSI C. All experiments were carried out on an Intel Xeon 2.66GHz workstation. The tertiary structure figures were generated using `PyMol` (http://www.pymol.org).

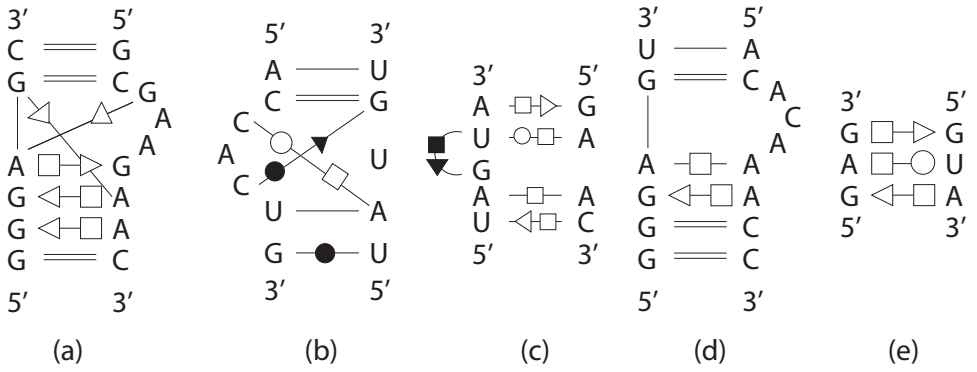

Figure 2.3: Base-pairing patterns of the query motif structures in 2D diagrams. (a) Kink-turn motif. (b) C-loop motif. (c) Sarcin-ricin motif (d) Reverse Kink-turn motif. (e) E-loop motif query structures.

2.3 Results

To assess the performance of `RNAMotifScan`, we searched the five RNA motifs against a 23S rRNA structure from *H. marismortui* (1S72, resolution 2.40 Å). We compared our results with three latest methods: `FR3D` [113], a *de novo* clustering method developed by Djelloul and Denise [37], and the shape histogram method developed by Apostolico *et.al* [6]. Since the clustering method mainly aims at the *de novo* motif discovery, the method may miss

some true instances. We also used RNAMotifScan to search the five motifs against the entire PDB for new motif occurrences.

2.3.1 Kink-turn

The Kink-turn motif is an asymmetric internal loop serving as an important site for protein recognition and RNA tertiary interactions [75, 135]. The 'kink' can be observed in the longer strand of the loop, which is stabilized by the two cross-strand stacking adenine residues. It brings together the two minor groove edges, and, consequently, produces a sharp turn of the two supporting helices [83, 84].

RNAMotifScan has identified 6 local motifs (motifs involve 2 or less strands) following by 1 composite motif (motifs involve 3 or more strands) from 1S72 (see Table 4.1). FR3D finds all these 7 motifs but introducing several 'related motifs' using the same query (see Table 5 of FR3D results [113]). FR3D also retrieves 2 more composite motifs. (The reason is that FR3D produces target segment structure based on spacial frame instead of sequence order.) The current version of RNAMotifScan does not focus on identifying composite motifs, but this feature can be included in the future (see Discussion Section). The shape histogram method finds all the 6 local motifs, but missing all the composite motifs. The *de novo* clustering method successfully rediscovers the motif, however, it misses 4 out of the 6 local motifs and all composite motifs. The results suggest that RNAMotifScan has higher sensitivity than shape histogram method and *de novo* clustering method in identifying Kink-turn motifs.

Table 2.1: Top hits obtained by searching the five motifs against 1S72 using `RNAMotifScan`

Ranking	Chain	Location	Score	p-value	`FR3D`	*de novo* Clustering	Shape Histogram
KT							
1	0	**77-82/92-100**	70.2	0.009	*	*	*
2	0	**1211-1217/1146-1156**	62.1	0.014	*		*
3	0	**936-941/1025-1034**	55.8	0.022	*	*	*
4	0	**1338-1343/1311-1319**	54.7	0.024	*		*
5	0	**1586-1593/1601-1609**	45.4	0.062	(*)		*
6	0	**244-250/259-267**	44.4	0.072	(*)		*
7	0	**2903-2906/2845-2855**	43.8	0.078	(*)		
CL							
1	0	**1436-1440/1424-1430**	40.9	0.033	-	*	-
2	0	**2760-2764/2716-2722**	39.1	0.041	-	*	-
3	0	1939-1945/1892-1898	38.4	0.044	-		-
4	0	**1004-1009/957-964**	34.4	0.081	-		-
SR							
1	0	**211-215/225-228**	42.8	0.007	*	*	-
2	0	**1368-1372/2053-2056**	42.8	0.007	*	*	-
3	0	**2690-2694/2701-2704**	42.8	0.007	*	*	-
4	9	**76-80/102-105**	42.0	0.007	*		-
5	0	**461-466/475-478**	37.5	0.010	*	*	-
6	0	**380-383/406-408**	34.4	0.013		*	-
7	0	**951-955/1012-1016**	33.4	0.015			-
8	0	**173-177/159-162**	29.8	0.022	*	*	-
9	0	2090-2094/2651-2654	26.2	0.037			-
10	0	1775-1779/1765-1768	25.5	0.042			-
11	0	1542-1545/1640-1643	21.0	0.117			-
12	0	**585-590/568-572**	20.8	0.126	*		-
13	0	**355-360/292-296**	20.8	0.126	*		-
RK							
1	0	**1661-1666/1520-1530**	48.6	0.114	-	*	-
2	0	**1530-1536/1649-1661**	46.8	0.145	-	*	-
EL							
1	0	**706-708/720-722**	21.2	0.052	-	*	
2	0	**1543-1545/1640-1642**	20.6	0.061	-	*	
3	0	**174-177/159-161**	18.7	0.098	-		*
4	0	**663-666/680-683**	18.6	0.100	-		
5	0	**586-590/568-571**	18.0	0.120	-		*
6	0	**356-360/292-295**	18.0	0.120	-		*
7	0	**2691-2694/2701-2703**	17.8	0.130	-		*
8	0	**1369-1372/2053-2055**	17.8	0.130	-		*
9	0	**463-466/475-477**	17.8	0.130	-		*
10	0	**380-383/406-408**	17.8	0.130	-		*

KT: kink-turn, CL: C-loop, SR: sarcin-ricin, RK: reverse kink-turn, EL: E-loop. Symbol notations: '*': identified; '(*)': identified after other related instances; '-': not studied. **Bold** typeface: *bona fide* motifs. Underlined: *de novo* found by `RNAMotifScan`.

2.3.2 C-loop

The C-loop motif is an RNA-protein binding site, and characterized by the unique multi-pairings formed by its two cytosine residues [83]. The two interleaving non-canonical base-pairs from the two multi-pairings bring together the interacting nucleotides, leaving the unpaired adenine residue at the minor groove and fully accessible [130].

`RNAMotifScan` has identified 3 C-loop motifs in 1S72 (see Table 4.1). The *de novo* clustering method can also classify the first 2 C-loop motifs. (`FR3D` and shape histogram methods were not used to search C-loop motifs. Because it is difficult for these 3D structure based methods to identify motifs that are small and usually exhibit high structural variations, such as C-loops.) The first 2 C-loop motifs exhibit high conservation comparing to the query motif (isomorphic as defined in the *de novo* clustering method), such that they can be easily detected by *de novo* clustering method. The 4th C-loop motif (supported by Lescoute *et al.* [84]) has one nucleotide inserted between the two multi-paired cytosine residues. Therefore, it cannot be found by the *de novo* clustering method but still can be detected by `RNAMotifScan` in which insertions (deletions) are taken into account. The results suggest that `RNAMotifScan` has higher sensitivity than the *de novo* clustering method. At the same time, we expect that our specificity can also be raised by carefully distinguishing the effects of different variations (see 'Discussion' Section).

2.3.3 Sarcin-ricin

The Sarcin-ricin motif in the ribosomal RNAs is involved in the interaction with elongation factors [125]. This interaction can be inhibited while the motif is bounded and modified by ribotoxins such as α-sarcin (ribonuclease) and ricin (RNA N-glycosidase) [118]. The base-pair pattern is highly conserved in 23S-28S rRNA from large ribosomal subunit, producing an 'S' shape bend in most of the Sarcin-ricin motifs.

`RNAMotifScan` has identified 9 known Sarcin-ricin motifs, whereas 8 were identified by `FR3D` and 6 were classified by *de novo* clustering method. `RNAMotifScan` identified 1 new Sarcin-ricin motif which was also observed by St-Onge *et al.* [119]. Three other motifs found by `RNAMotifScan` rank at low places in the results, showing a satisfactory specificity for our method (see Table 4.1). Even though these instances show higher structural variation from the query structure, we suggest that they should be further inspected as they show interesting conservations in base-pairing pattern comparing to the known Sarcin-ricin motifs.

2.3.4 Reverse Kink-turn

The Reverse Kink-turn is also an asymmetric internal loop that produces sharp bend as the Kink-turn motif, however, towards the opposite direction [78]. Another difference is that the longer strand of the Kink-turn motif makes a tight bend, while in the Reverse Kink-turn motif, the tight bend is observed in the shorter strand as the longer strand gradually turns to the major/deep groove [122].

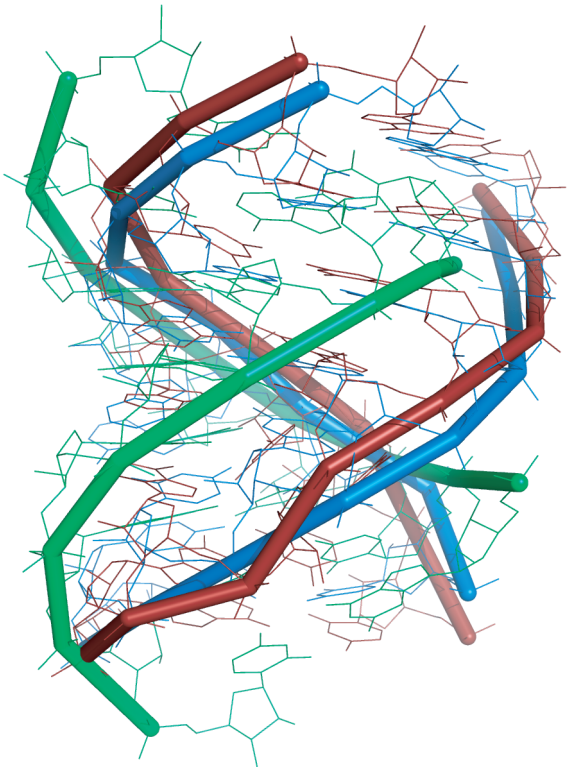

Figure 2.4: The superimposition of the new E-loop motif found by RNAMoitfScan (red), a segment of regular A-form helix (green), and a well characterized E-loop motif (blue). The new E-loop is found at 1S72, chain '0', 662-669/677-684, the A-form helix is found at 1S72, chain '0', 13-20/523-530, and the well-characterized E-loop motif is found at 1S72, chain '0', 1639-1646/1539-1546. The RMSD resulting from superimposing the new motif (red) and the model (blue) is 2.496Å; while the RMSD for superimposing the regular A-form helix (green) and the model (blue) is 4.807Å.

The *de novo* clustering method suggests 6 Reverse Kink-turn occurrences. (`FR3D` and shape histogram method were not used to search Reverse Kink-turn motifs either.) We noticed that 3 of these 6 motifs given by clustering are false positives (2397-2399/2389-2391, 2307-2310/2298-2300 and 1132-1134/1228-1230), as they either come from the irregular pairing regions near hairpin loop regions instead of being the junction regions between two helical regions, or do not produce significant sharp turns. `RNAMotifScan` has identified 2 of the 3 true Reverse Kink-turn motifs (see Table 4.1). The 1 motif missed is due to its higher structural variation. Even though `RNAMotifScan` may miss several occurrences, it has much higher specificity and thus more reliable is practical applications.

2.3.5 E-loop

The E-loop was originally defined as the symmetric internal loop region in the 5S rRNA that separates its helical regions IV and V [28, 81]. The motif can be decomposed into two isosteric submotifs, which are positioned with relative 180° rotation [79, 81]. The submotif is usually referred to as 'bacterial E-loop', and its base-pair pattern was summarized as a *trans* H/SE base-pair, a *trans* WC/H or *trans* SE/H base-pair, and a *cis* bifurcated or *trans* SE/H base-pair by Leontis *et al.* [79]. Since the isostericity related with bifurcated base-pair is not defined, we consider only the *trans* SE/H as the third base-pair in the query.

There are 2 E-loop motifs classified by *de novo* clustering method and 8 identified by shape histogram method. The two sets of results show no overlap and the union of them gives totally 10 E-loop motifs. `RNAMotifScan` has successfully identified 9 of them (see Table 4.1), and 1 new E-loop occurrence. This new E-loop occurrence, as well as a segment of regular

A-form helix, are superimposed with a well characterized E-loop motif (see Figure 2.4). The superimposition of the new E-loop instance results much smaller RMSD than the superimposition of the A-form helix, indicating that this E-loop occurrence cannot be expected to find randomly. `RNAMotifScan` has missed 1 E-loop motif that has both high sequence and base-pair variations. Note that E-loop motifs can tolerate higher variations comparing to other motifs. (They were clustered into 3 families using the *de novo* clustering method [37].) Therefore, the results generated by searching only one of its variants could be limited. However, `RNAMotifScan` outperforms both methods when given only one query, and the E-loop identification can be further optimized by including other variants of E-loop motifs as query.

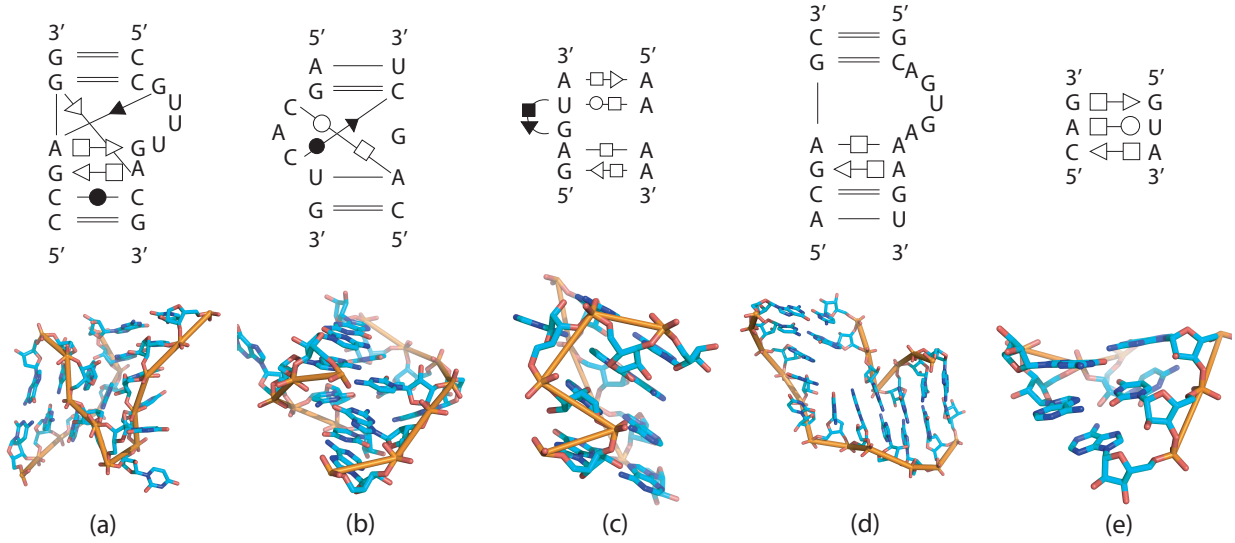

Figure 2.5: The 2D diagrams and 3D structures of newly identified motifs with sequence or base-pair variations. (a) Kink-turn motif from 23S rRNA in *H. marismortui* (PDBid: 1QVF, chain '0', location 936-941/1025-1034). (b) C-loop motif from 5.8S/28S rRNA in *S. cerevisiae* (PDBid: 1S1I, chain 3, location 1436-1440/1424-1430). (c) Sarcin-ricin motif from 16S rRNA in *E. coli* (PDBid: 1VS7, chain A, location 888-892/906-909). (d) Reverse Kink-turn motif from 23S rRNA in *H. marismortui* (PDBid: 1QVF, chain '0', location 1661-1666/1520-1530). (e) E-loop motif from 23S rRNA in *S. oleracea* (PDBid: 3BBO, chain A, location 1392-1394/1379-1381).

2.3.6 3D Resolution Affects Identification Accuracy

We observe that the identification results of `RNAMotifScan` is dependent on the quality of the annotation program, which turns out to be dependent on the resolution of the 3D RNA structure. To demonstrate this, we selected three PDB entries with different resolutions for the same 16S rRNA structure from *T. thermophilus* (PDBid: 2VQE, 1J5E, and 1I95), and used `RNAMotifScan` to identify the five motifs in them. Only hits with p-value less than the defined cutoffs (see Table 2.3) are counted. Since the RNA structure from 2VQE contains three RNA chains while the other two structures contain only one RNA chain, we only consider their common RNA chain (chain A in the comparison). The results are shown in Table 2.2. In the table we can find that MC-Annotate tends to annotate fewer base-pairs in the low resolution RNA structures. Among those missed base-pairs, most of them are non-canonical base-pairs, which are critical for the structural motif identification. Even if the numbers of annotated base-pairs are comparable for two structures with different resolutions, their qualities differ. For example, 2VQE and 1J5E have almost the same number of annotated base-pairs, but one kink-turn that can be identified in 2VQE is missed in 1J5E.

2.3.7 Scanning PDB

Finally, we searched the entire PDB for the five query motifs. The running time for scanning PDB is 64m35s for Kink-turn, 74m29s for C-loop, 51m49s for sarcin-ricin, 77m59s for Reverse Kink-turn and 72m55s for E-loop motif. The results are summarized in Table 2.3. The motifs identified by `RNAMotifScan` are several times more than the current known instances (p-value

cutoffs are shown in Table 2.3, the estimated FPR is less than 0.01). Still, we expect the numbers are underestimated since our cutoffs are set to be rather stringent. Although the large difference between the identified motifs and the currently known ones may due to the fast growing of RNA structures deposited in PDB, we still find new RNA motif occurrences in non-ribosomal RNAs, such as riboswitches, ribozymes, and protein-mRNA complexes. The complete results can be found at the supplementary website.

Table 2.2: The performance of `RNAMotifScan` with different resolutions of RNA structures

PDB ID	Resolution	Length	#bp	#Can.	#Non-can.	#KT	#CL	#SR	#RK	#EL
2VQE	2.50Å	1522	766	433	333	3	0	2	0	6
1J5E	3.05Å	1522	761	434	327	2	0	2	0	6
1I95	4.50Å	1514	699	422	277	1	0	0	0	3

The columns in the tables represent PDB codes of the RNA structures, the resolution, the length, the number of base-pairs (bp) annotated by MC-Annotate, the number of annotated canonical base-pairs (Can.), the number of annotated non-canonical base-pairs (Non-can.), the number of Kink-turn (KT), C-loop (CL), Sarcin-ricin (SR), Reverse Kink-turn (RK) and E-loop (EL) being identified. All structures are *T. thermophilus* 16S rRNA structures. The *p*-value cutoffs are the same as those shown in Table 2.3.

Table 2.3: Summary of the `RNAMotifScan` search results against the entire PDB comparing with SCOR

Motif	*p*-value cutoff	PDB	NR PDB	SCOR
Kink-turn	0.07	553	39	195
C-loop	0.04	167	18	-
Sarcin-ricin	0.02	633	46	107
Reverse Kink-turn	0.14	56	3	-
E-loop	0.13	1356	148	37

C-loop and Reverse Kink-turn are not included in SCOR. Motifs characterized in SCOR were from the entire PDB released by Oct. 24, 2004. The non-redundant set (NR PDB) is constructed by removing entries with sequence identities greater than 90%.

To demonstrate the advantages of `RNAMotifScan`, we compared five query motifs (Figure 2.3) with five different newly identified motifs (Figure 2.5). For C-loop motif, we observed that the sequence identity is 66% between the C-loop query (Figure 2.3 (b)) and the new identified C-loop motif (Figure 2.5 (b)), which sequence-based search methods may miss. The

Sarcin-ricin motif (Figure 2.3 (c)) and the E-loop motif (Figure 2.3 (e)) consist of all non-canonical base-pairs, such that they cannot be searched by methods that are restricted to canonical base-pairs. The newly identified Sarcin-ricin motif and E-loop motifs also have three isosteric base-pair changes (Figure 2.5 (c) and (e)). The newly identified Kink-turn motif (Figure 2.5 (a)) shows two base-pair variations (*trans* SE-H to *cis* SE-SE, and *trans* SE-H to *cis* WC-WC), which would be missed by the strict base-pair graph isomorphism search. More importantly, we found that the newly identified Kink-turn (Figure 2.5 (a)) and Reverse Kink-turn motifs (Figure 2.5 (d)) show structural variations comparing to the query motifs. One nucleotide is inserted at the 'kink' region of the newly identified Kink-turn motif, resulting an 'U' shape 'kink' rather than the 'V' shape 'kink' in the query (see Figure 2.6 (a)). For the newly identified Reverse Kink-turn motif, the structural variation is observed at the longer strand of its junction between two helices. Two nucleotides are inserted at this region, relaxing the turn significantly (Figure 2.5 (d)). At the same time, a sharp bend is created at this region (see Figure 2.6 (b)), in order to accommodate the insertions and maintain the proper structure of the motif.

2.4 Discussion

The base-pairs from the RNA 3D structures are extracted and classified by various annotation tools. The annotations of base-pairs are produced based on the geometric constraints among atoms involving the hydrogen bond interactions. In another word, the accurate coordinates of atoms are critical for the classification of base-pairs. Therefore, the quality of annotation results, and consequently the accuracy of `RNAMotifScan`, depends largely on the resolution of the RNA 3D structure (see Table 2.2). We anticipate that with the advances of RNA

structure determination techniques, more and more high quality data can be produced and the RNA motif identification can be more reliable.

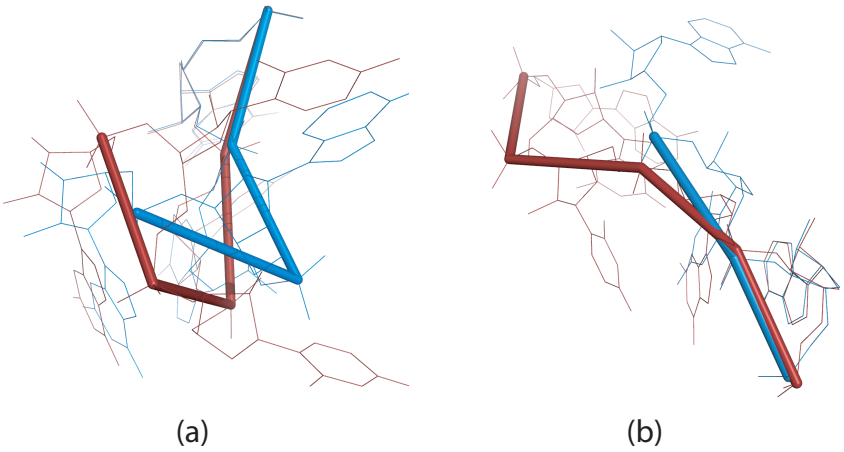

Figure 2.6: The Superimposition between the newly identified motifs (red) and the queries (blue) at the regions where nucleotide insertion(s) are observed. (a) The 'kink' regions in Kink-turn motifs (red structure: 1QVF, chain '0', 1027-1031; blue structure: 1S72, chain '0', 94-97). (b) The longer strands at the junctions between helices in Reverse Kink-turn motif (red structure: 1QVF, chain '0', 1522-1526; blue structure: 1ZZN, chain B, 198-200).

It is mentioned that `FR3D` is capable of discovering composite motifs while `RNAMotifScan` mainly focuses on local motifs. However, `RNAMotifScan` can be easily extended to include RNA composite motifs. If the motif consists of n strands, there are in total $n!$ combinations of orders that these strands can be concatenated. Theoretically, it is possible to include any number of strands with the compensation of running time. In practice, there is only a small number of strands in RNA structural motifs. Therefore, it is feasible to enumerate all possible strand concatenations. We plan to include this feature in the future versions of `RNAMotifScan`.

Currently, `RNAMotifScan` uses a scoring function that does not distinguish substitutions between different isosteric groups. Recently, Stombaugh *et al.* studied the frequencies of non-canonical base-pair substitution among different isosteric groups and proposed a more

sophisticated scoring function [120]. We plan to incorporate such scoring function into our method. Moreover, the scoring function should also be position dependent (similar as the Position Specific Scoring Matrix). For example, the determination of C-loop motif relies on the two multi-paired cytosine residues. We should assign heavy penalty to the mutations on these nucleotides. Similarly, for E-loop motifs, we should give heavy weight to the conserved *trans* H/SE base-pair according to the E-loop motif definition. With the incorporation of more sophisticated base-pair substitution scoring function and position dependent weights, we anticipate that `RNAMotifScan` will become much more accurate in identifying RNA structural motifs.

CHAPTER 3: SEARCHING RNA STRUCTURAL MOTIFS BY ADDING BASE-STACKING INFORMATION

In Chapter 2, we have described `RNAMotifScan`, a new RNA structural motif alignment tool based on the non-canonical base-pairing patterns of the motifs. Although this work is of great success compared to the current state-of-the-art RNA structural motif identification tools, improvements can still be expected to further increase its accuracy. In this chapter, we will discuss `RNAMotifScanX`, an enhanced version of `RNAMotifScan` by incorporating base-stacking information, which is extremely important for the folding of RNA structures. Benchmark experiments between `RNAMotifScanX` and `RNAMotifScan` clearly show the improvement of `RNAMotifScanX` in both accuracy and sensitivity.

One should note that, however, `RNAMotifScanX` cannot fully replace `RNAMotifScan`. This is because `RNAMotifScanX` is developed using a graph alignment algorithm, which requires exponential time to run. In this case, when computational efficiency is of high importance (such as searching the entire PDB or clustering a large number of motif instances), `RNAMotifScan` should be used. On the other hand, when the detailed analysis of one or several RNA structures is the major purpose, we recommend `RNAMotifScanX` for its improved accuracy and sensitivity.

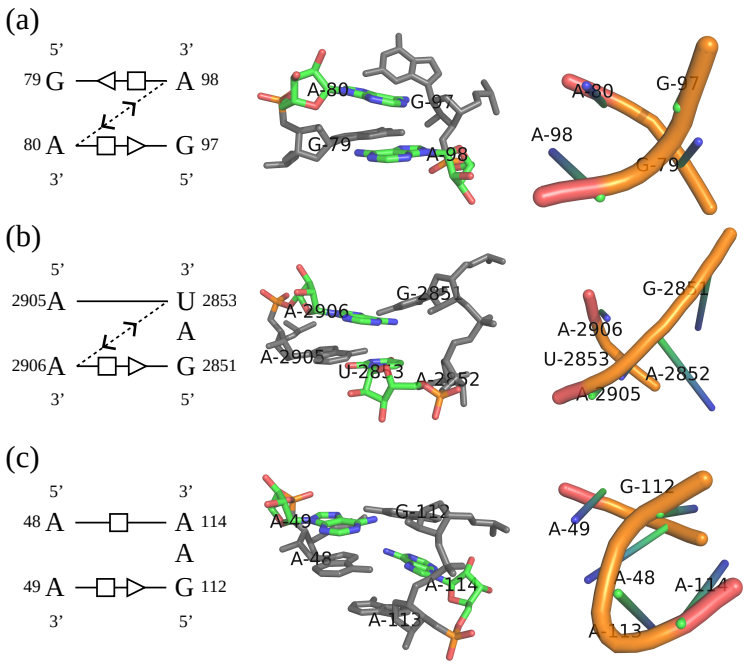

Figure 3.1: An example showing that base-stacking information can distinguish structure-conserved mutations from other mutations. Structure components from 1S72, chain '0', (a) 79-80/97-98 (a real kink-turn motif instance), (b) 2905-2906/2851-2853 (a real motif instance with non-isosteric base-pair variation), and (c) 48-49/112-114 (a unrelated motif instance). Left panels: base-interaction patterns of the structure components. Nomenclature for base pairs follows Leontis *et al.* [82] and for base-stacking interactions follows Major *et al.* [89]. Middle panels: the 3D structure of the structure components. The two stacking nucleotides (or the corresponding nucleotides in (c)) are colored. Right panels: the trajectories of the right-hand side strands of the structure components. The backbone trajectories of (a) and (b) are similar, as they share conserved base-stacking interactions.

3.1 Intuition of Incorporating Base-stacking Information

Different RNA structural motif search tools model RNA structural motif from different perspectives, and tools developed based on 3D geometry and base-pairing pattern have their own advantages and limitations. The 3D geometry-based methods are highly specific, as the conserved 3D geometry is a direct and strong indication of true motif occurrence. On the other hand, the base-pairing pattern-based methods are highly sensitive, because of its more flexible modeling of RNA structural motif that takes into account of the possible structural variations, or the plasticity [5, 32], of the RNA structural motifs. Tools that consider both information, e.g. `FR3D`, usually prioritize one type of these information over the other, for that it is difficult to simultaneously optimize both of them using a computational approach.

In this case, to improve the RNA structural motif identification accuracy, we may either relax the 3D geometry information or incorporate additional constraint to the base-pairing information. The first strategy has been tried out by a number of existing methods, such as representing a nucleotide with its key atoms [59] or its geometric center [105, 113], or characterizing the backbone trajectory using the dihedral angles formed between different key atoms of the nucleotides [42, 44, 49]. Although improvements have been made by using these means of abstraction, further improvements can still be expected. In this case, we turn to the second strategy and pose the following question: instead of generalizing the 3D geometric information, can we find a way to enrich the base-pairing information, so as to maintain the original high sensitivity and at the same time increase the specificity?

We propose to incorporate the base-stacking [22] information into the original base-pairing modeling of RNA structural motif for an improved specificity. Contemporary RNA secondary

structure prediction tools such as `Mfold` [153, 155] and `RNAfold` [66] rely on the experimentally determined base-stacking energy parameters [20, 133] to predict the thermodynamically stable RNA structure(s). Meanwhile, base-stacking interactions are also important when describing the formation and characteristics of RNA structural motifs [78, 79]. Previously, Leontis *et al.* have categorized the non-canonical base pairs into isostericity groups based on the C1'-C1' distance of the pairing bases [80, 120]. Similarly, the base-stacking interactions can also be categorized based on the directions of the normal vectors to the base planes that are being stacked. Major and Thibault defined four categories of base-stacking interactions, namely *upward* (>>), *downward* (<<), *inward* (><), and *outward* (<>) [89, 101]. Using the classification of both non-canonical base pairs and base-stacking interactions, we can estimate the conservation of both base-pairing and base-stacking patterns between RNA structural motif instances.

We now show a real example to demonstrate the importance of base-stacking information in RNA structural motif identification. The base-stacking information can provide additional evidence for structure conservation, while the information regarding sequence and base-pairing pattern is vague and inadequate. In Figure 3.1 we show a tandem-sheared non-canonical base-pair core found in the kink-turn motif (Figure 3.1a), as well as two structure components found in 1S72 with exactly one nucleotide insertion and one base-pair variation in each (Figure 3.1b and c). The first structure component (Figure 3.1b) contains a base-pair mutation that changes the original sheared base pair (*trans* S/H G79-A98 in Figure 3.1a) into a canonical A2905-U2853 base pair. The corresponding base-pair mutation found in the second structural component (Figure 3.1c) alters the sheared pair into a *trans* H/H A48-A114 pair. Because both structural components contain the same degree of sequence and base-pair variation, it is difficult to distinguish which one is structurally conserved comparing

to the true kink-turn motif instance. In fact, `RNAMotifScan` (discussed in Chapter 2), which considers the sequence and base-pairing information, favors the second structure component. This is because the base-pair mutation found in the second structural component is also a non-canonical base pair, and at the same time it adopts the *trans* orientation (while the orientation of the canonical base pair A2905-U853 in Figure 3.1b is *cis*).

When base-stacking information is considered, we found a conserved base-stacking interaction from the true kink-turn motif instance (Figure 3.1a) in the first structural component (Figure 3.1b), but not in the second (Figure 3.1c). The conservation of the base-stacking interaction can be seen from the middle panels in Figure 3.1. In this sense, the first structural component is more similar to the true kink-turn instance than the second structure component. Indeed, in Figure 3.1, right panels, we show that the backbone trajectory of the first structure component is highly similar to the kink-turn instance, while the second structural component exhibits a large degree of structural variation. In order to preserve the base-stacking interaction, the right-hand side strands (in the left panels) of both the true kink-turn instance and the first structural component adopt a slight clockwise bend (see the right panels), so as to bring together the nucleotides that form the base stacking. In contrast, without the pressure of forming such base-stacking interaction, the right-hand side strand of the second structure component adopts a sever anti-clockwise bend, which is completely different from the true kink-turn motif instance. In fact, the first structural component is taken from another real kink-turn motif instance with non-isosteric base-pair mutation, while the second structure component is an unrelated motif instance. This example clearly shows that the base-stacking information can be used to improve the RNA structure motif identification accuracy.

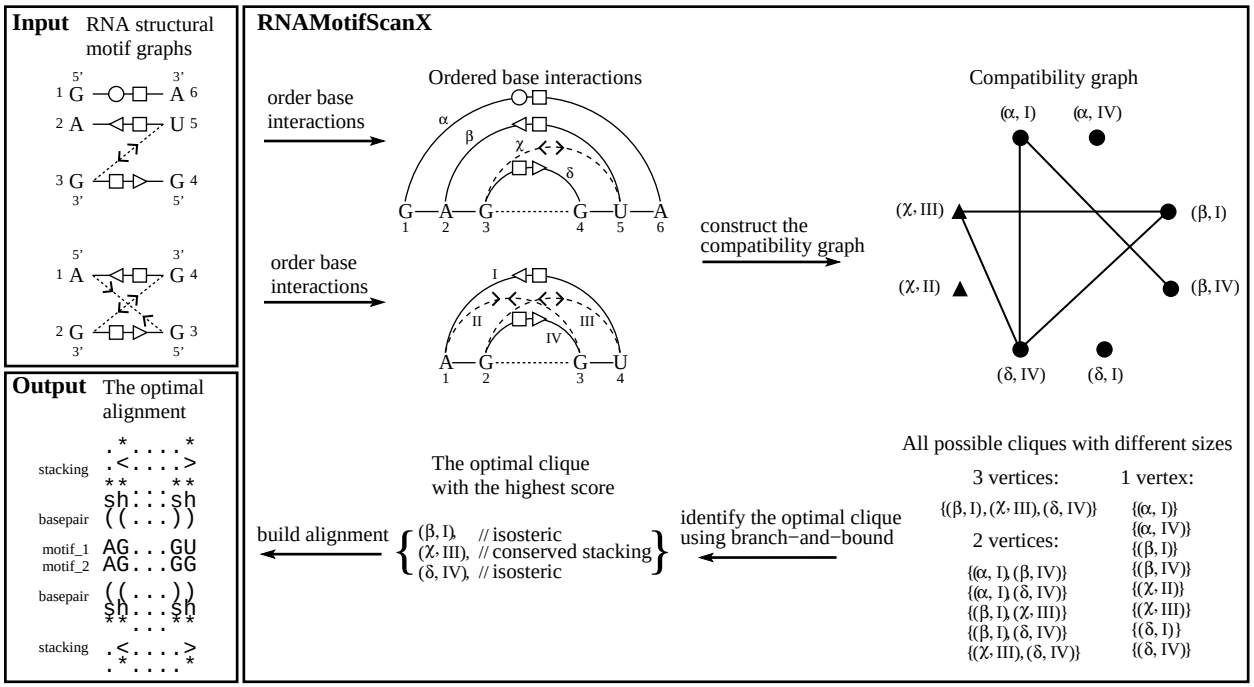

Figure 3.2: Summary of the `RNAMotifScanX` algorithm shown by aligning two artificial motif instances. The inputs of the algorithm are two RNA structural motif graphs of the motif instances to be aligned. The RNA structural motif graph are generated based on the base interaction patterns generated by using `MC-Annotate` [53] or `RNAVIEW` [147]. **Step 1**: The base interactions in the RNA structural motif graphs are then sorted based on the increasing order of their starting nucleotide (ties are broken by the decreasing order of their ending nucleotide). The base interactions in the first motif instance are sorted and labeled in order as $\alpha, \beta, \chi, \delta$, and those for the second motif instance are labeled in order as I, II, III, IV. **Step 2**: The compatibility graph is generated to account for all base-interaction matchings and their compatibility. A base-interaction matching is represented using a vertex in the compatibility graph. The base-pair matchings are indicated using rounds and the base-stacking matchings are indicated using triangles. Note that a base-pairing interaction cannot match with a base-stacking interaction. **Step 3**: A branch-and-bound version of the Bron and Kerbosch algorithm [23] is used to traverse all cliques in the compatibility graph and finds the optimal clique that corresponds to the highest alignment score. **Step 4**: The optimal clique is identified, and the corresponding alignment is generated based on such optimal clique.

The above example shows the importance of base-stacking information in RNA structural motif modeling and identification. Based on the intuition, we developed `RNAMotifScanX`, which considers all sequence similarity, base-pair isostericity, and base-stacking conservation when identifying RNA structural motifs. Benchmark results on `RNAMotifScanX` against its predecessor `RNAMotifScan` (where base-stacking information is not considered) on five well-characterized motifs (kink-turn, C-loop, sarcin-ricin, reverse kink-turn, and E-loop) show significantly improved accuracy. We have also found a novel kink-turn-like motif instance whose non-canonical helix turns to the opposite direction of its canonical helix. In this case, we have shown the utility of base-stacking interaction in modeling RNA structural motifs, and suggest that it should be considered by future motif search tools. We also anticipate `RNAMotifScanX` will significantly benefit related RNA structural motif research.

3.2 Materials and Methods

In this section, we will present the core algorithm of `RNAMotifScanX` and related technical details. We will first introduce the RNA structural motif graph alignment problem and discuss how to solve it by reformulating it into a clique finding problem. Then, a branch-and-bound solution will be presented to find the optimal alignment between the RNA structural motif instances. The main algorithm behind `RNAMotifScanX` is summarized in Figure 3.2. We will then present how base-stacking information is used by `RNAMotifScanX`, and finally a new approach to compute P-value with higher efficiency and more realistic universal cutoff for automatic motif identification.

3.2.1 Alignment of RNA Structural Motif Graphs

RNA structural motifs are naturally modeled as graphs, where the nucleotides are represented by the vertices and the base interactions are represented by the edges. We refer to these graphs as the *RNA structural motif graphs*. For example, Djellou *et al.* used graph isomorphism algorithm to evaluate the conservation between two RNA structural motif graphs and implemented the algorithm in their motif clustering method `LENCS` [37]. However, the isomorphism algorithm makes the evaluation of nucleotide insertion/deletion and base-pair substitution in different isosteric [80] groups rather difficult. In addition, the `LENCS` method does not consider complete base-stacking information. In this work, our objective is to devise a graph alignment algorithm that can more accurately compare two RNA structural motif instances by resolving these issues.

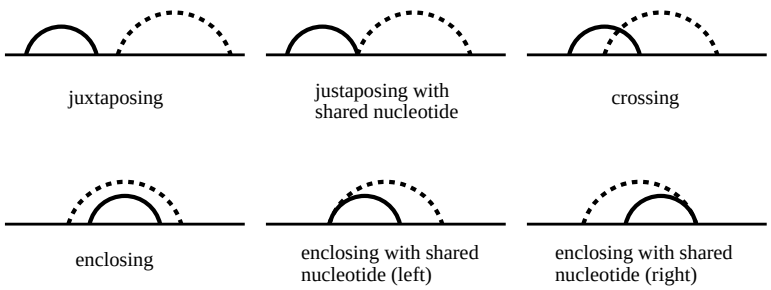

Figure 3.3: Six relation categories that can be formed between two base interactions. The horizontal lines indicate the RNA sequences and the arcs represent the corresponding base interactions. The base interactions indicated with the solid arcs are ordered before the ones indicated with the broken arcs.

Here we briefly define an alignment between two RNA structural motif instances as a list of one-to-one correspondence and well-ordered matchings between the two sets of nucleotides in the instances. By one-to-one correspondence we mean that each nucleotide in a motif instance can be matched to at most one nucleotide in the other motif instance. And by well-ordering we mean that the matchings of nucleotides cannot cross with each other (i.e.,

if the nucleotide i is matched to i' and the nucleotide j is matched to j', then $i' < j'$ if and only if $i < j$). The alignment maximizes an object function, which combines the base-pair similarity, base-stacking similarity (both adjacent and non-adjacent), and sequence similarity between the two RNA structural motif instances. The object function can be computed as follows:

$$\begin{aligned} &S(M^A, M^B) = \\ &w_1 * \textstyle\sum_i S^{BP}(p_i^A, p_i^B) + w_2 * \sum_i S^{NAT}(t_i^A, t_i^B) + \\ &w_3 * \textstyle\sum_i S^{AT}(\bar{t}_i^A, \bar{t}_i^B) + w_4 * \sum_i S^{SEQ}(L_i^A, L_i^B). \end{aligned} \tag{3.1}$$

Here, S^{BP} is the base-pair similarity between base pairs p^A and p^B (including the sequence similarity for the nucleotides that form the base pairs), S^{NAT} is the similarity between non-adjacent base-stacking interactions t^A and t^B, and S^{AT} is the similarity between adjacent base-stacking interactions $\bar{t}^A$ and $\bar{t}^B$ (we will discuss the handling of base-stacking interactions later), finally S^{SEQ} is the sequence similarity (computed by using the Needleman-Wunsch algorithm [98]) between the loop regions L^A and L^B. A loop region is defined as a set of continuous nucleotides that none of them participates in a base-pairing or a non-adjacent base-stacking interaction. The associated weights w_1 to w_4 are used to model the different impacts made by these structural features in defining the RNA structural motif of interest.

3.2.2 Reformulation of the RNA Structural Motif Alignment Problem into a Clique Finding Problem

The naive solution to the graph alignment problem is to enumerate all possible matchings of the vertices in the RNA structural motif graphs, which will likely lead to an inefficient implementation. To improve the computational efficiency, we observe that the base inter-

actions in the RNA structural motif graphs can be ordered, and they only form a limited number of pairwise relations in a given motif graph. These observations are key to the branch-and-bound algorithm that will be discussed in the next section. We order the base interactions according to the increasing order (from 5' to 3') of their first nucleotides, and break ties based on the decreasing order of their second nucleotides (see Figure 3.2). Given such ordering, we can categorize the relation between all pairwise base interactions into six groups (see Figure 3.3). In Figure 3.3, the base interactions indicated by the solid arcs are ordered before the interactions indicated by the broken arcs. To ensure a valid alignment, we claim that (1) the relation group of any aligned base interactions in the first motif instance must be the same as those in the second motif instance, and (2) the relative ordering of any aligned base interactions in the first motif instance must be consistent to that in the second motif instance.

With these two constraints, we can summarize the two input RNA structural motif graphs into one *compatibility graph* (Figure 3.2). Each vertex in the compatibility graph represents a base-interaction matching. Consider that the RNA structural motif A contains $|\mathcal{P}^A|$ base pairs and $|\mathcal{T}^A|$ non-adjacent base-stacking interactions. ($|\mathcal{P}^B|$ and $|\mathcal{T}^B|$ are defined accordingly.) The total number of vertices in the compatibility graph is $|\mathcal{P}^A| * |\mathcal{P}^B| + |\mathcal{T}^A| * |\mathcal{T}^B|$. For example, in Figure 3.2, the first motif instance contains 3 base-pairing and a single base-stacking interactions, while the second motif instance contains 2 base-pairing and 2 base-stacking interactions. The corresponding compatibility graph thus have 3 * 2 + 1 * 2 = 8 vertices. Two base-interaction matchings are compatible if the two constraints stated in the previous paragraph are satisfied. In this case, we add an edge between the two corresponding vertices. For any valid alignment, all of its base-interaction matchings must be compatible with each other, which would form a clique (completely connected graph) in the

compatibility graph. Therefore, finding the optimal alignment between two RNA structural motif instances is equivalent to identifying the optimal clique in the compatibility graph that corresponds to the highest alignment score (Equation 3.1).

Previously, Rahrig *et al.* formulated the RNA 3D structure alignment problem into a maximum clique finding problem, and implemented its solution into a tool called `R3D Align` [105]. This algorithm shares the high-level objective with our RNA structural motif graph alignment problem, but differs significantly in the following two aspects. First, `R3D Align` aims to identify the *maximum* clique in the local alignment graph to include as many matchings as possible, while the optimal clique for `RNAMotifScanX` can be neither *maximum* nor *maximal*. This is because `RNAMotifScanX` aims to find the local alignment between two RNA structural motif instances by optimizing a more sophisticated object function (Equation 3.1). For example, an isosteric base-pair matching may result in much higher score than several matchings of non-isosteric or canonical base pairs. In this case, all cliques in the compatibility graph must be systematically traversed to guarantee optimality. Second, R3D Align implements a greedy algorithm to find the maximum clique for the sake of computational efficiency, while `RNAMotifScanX` adopts a branch-and-bound algorithm that guarantees the global optimal solution. The branch-and-bound solution is appropriate for aligning RNA structural motifs for that their sizes are usually small. In this case, `RNAMotifScanX` is different from `R3D Align` in terms of both problem formulation and algorithm design.

3.2.3 Identification of the Optimal Alignment Clique

Bron and Kerbosch [23] devised an algorithm to enumerate all possible cliques in a given graph, and we adopt the major idea of this algorithm to find the optimal clique in the compatibility graph. The algorithm maintains two vertex sets $\mathcal{R}$ and $\mathcal{C}$, for holding the vertices that have *already* been included in the current clique and the candidate vertices that *will* potentially be included in the optimal clique, respectively. Each vertex in the candidate set $\mathcal{C}$ is required to connect with all vertices in the identified set $\mathcal{R}$, so as to fulfill the complete-connection definition of clique. This constraint avoids unnecessary computations by only considering vertices in the set $\mathcal{C}$. The algorithm proceeds by picking a vertex in $\mathcal{C}$ and adding it to $\mathcal{R}$, and updating the set $\mathcal{C}$ with the complete-connection constraint based on the updated $\mathcal{R}$. This procedure is recursively executed with the updated sets $\mathcal{R}$ and $\mathcal{C}$ until $\mathcal{C}$ is exhausted. For each identified clique, we evaluate the corresponding alignment score using the object function described in Equation 3.1 and record the maximum score that have achieved so far. The optimal alignment for the two RNA structural motif instance can be identified after the traversal of all possible cliques.

We devise a branch-and-bound technique to speedup the naive Bron and Kerbosch algorithm. Observe that for the candidate set $\mathcal{C}$, if k vertices in C are finally added to $\mathcal{R}$, there should be at least $k(k-1)/2$ edges formed by the vertices in $\mathcal{C}$ (complete-connection definition of clique). Conversely, if we simply count how many edges are formed between the candidate vertices, we will be able to determine the size of the maximal clique for this branch. As we also know the matching score for all base-interaction matchings (which can be directly looked up from the scoring matrix), we can compute the corresponding upper bound by assuming that the high-score base-interaction matchings are taken as parts of the optimal clique. The

initial lower bound is computed using a heuristic algorithm called `CLCL` (will be discussed in Chapter 5) that finds the maximum clique in a graph, and is updated whenever a higher score is achieved as the algorithm proceeds.

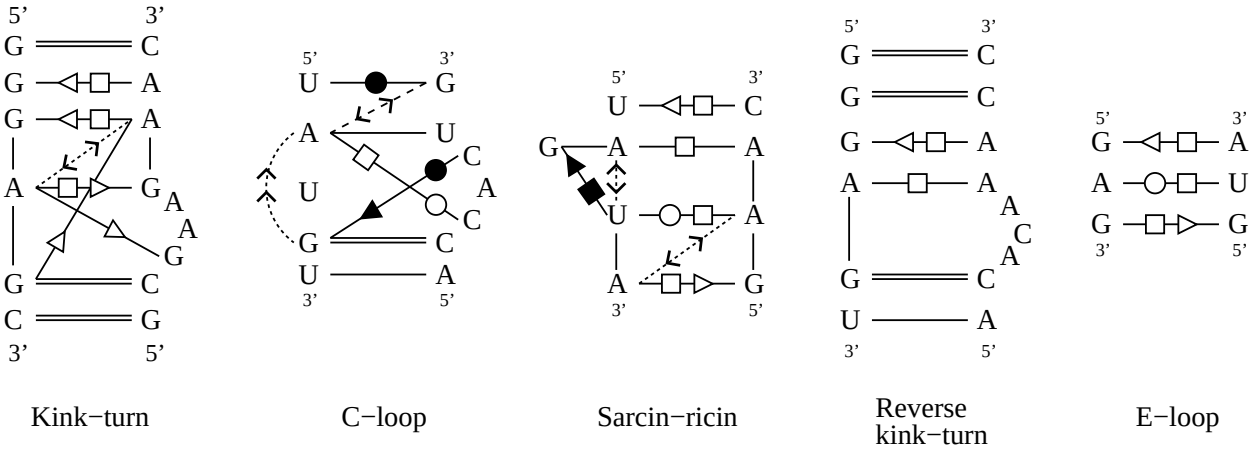

Figure 3.4: The revised consensus base-interaction patterns used by `RNAMotifScanX` to search for related motif instances.

3.2.4 Comparison of Base-stacking Interactions

In this section, we discuss how the base-stacking interactions are handled in `RNAMotifScanX`. Recall that the comparison of non-canonical base pairs is based on evaluating their isosteric-ity [80, 120]. A base-pair substitution scoring function can be devised (similar to those used by `RNAMotifScan`) to prioritize isosteric base-pair matchings. Similarly, the conservation of base-stacking interactions can also be evaluated using the classification proposed by Major and Thibault [89, 101] as we have outlined in the Introduction section. Correspondingly, we can devise an *ad hoc* scoring matrix to evaluate the substitution between base-stacking interactions in the four different categories (upward, downward, inward, and outward). The

setup of the base-stacking substitution matrix is generic, where the substitution between the same category is given a universal (for all four categories) bonus score, and the substitution between different categories is given a universal (for all combinations of any two different categories) penalty score. We use this generic scoring setup to demonstrate the importance of base-stacking information, rather than emphasizing parameter tuning. Nevertheless, more realistic scoring functions are strongly encouraged.

We distinguish the base-stacking interactions in a given RNA structural motif instance as either *non-adjacent* or *adjacent*. We are more interested in evaluating the substitution of non-adjacent base-stacking interactions, as they might suggest unusually structural configurations at the corresponding region. The non-adjacent base stacking are processed as general base pairs, but with their specific substitution scoring function. Also, note that a base-stacking interaction can only be matched with another base-stacking interaction (see Figure 3.2). For the adjacent base-stacking interactions, we evaluate their substitution effects during the sequence alignment process. That is, if two consecutive nucleotides are aligned, and they happen to form a conserved base-stacking interaction in both motif instances, then a corresponding bonus score will be assigned to prioritize such sequence alignment.

3.2.5 P-value Computation

The estimation of statistical significance for the alignment scores is critical towards automatic detection of RNA structural motif instances. We expect to improve the P-value computation strategy of `RNAMotifScan` by the following two aspects. First, the P-value estimation of `RNAMotifScan` makes it difficult to find a universal cutoff that performs well on all types of

motifs, and different cutoffs were suggested for each specific motif. This is because each motif adopts highly different structural configurations, and allows different degrees of variation. Second, the original strategy requires massive execution of the program on the randomly generated data set to simulate the distribution of alignment scores. This is computationally infeasible for `RNAMotifScanX`, as it is a branch-and-bound algorithm that runs slower than `RNAMotifScan`.

To solve these two problems, we observe that under the current scoring setup, randomly generated motif instances that have high degree of structural variation often result in low alignment scores. Also, the alignment scores on random motif instances with conserved structural configuration but un-conserved base-interaction substitutions and nucleotide insertions/deletions appears to fit the hypothesized extreme value distribution better. This observation suggests that we might be able to compute more universal P-values estimates by taking random motifs with conserved structural configuration as the background. Note that this strategy also helps to solve the second problem. As we ensure that the random instances have the same structural configuration with the query motif, we can assess the corresponding alignment score immediately without actually running the program. In this case, online estimation of the P-value becomes feasible. We will show the comparison of the P-value estimation strategy between `RNAMotifScan` and `RNAMotifScanX` in the Results Section.

Table 3.1: Comparison between `RNAMotifScan` and `RNAMotifScanX` in identifying Kink-turn motifs from ribosomal RNA 1S72

	`RNAMotifScan`				`RNAMotifScanX`			
Ranking	chain	Location	Score	*P*-value	chain	Location	Score	*P*-value
1	'0'	**77-82/92-100**	70.2	0.009	'0'	**77-82/92-100**	167.4	0.009
2	'0'	**1211-1217/1146-1156**	62.1	0.014	'0'	**936-941/1025-1034**	138.4	0.014
3	'0'	**936-941/1025-1034**	55.8	0.022	'0'	**2911-2914/2667-2669/2820-2829**	130.6	0.017
4	'0'	**1338-1343/1311-1319**	54.7	0.024	'0'	**1211-1217/1146-1156**	128.0	0.018
5	'0'	**1586-1593/1601-1609**	45.4	0.062	'0'	**1338-1343/1311-1319**	126.6	0.019
6	'0'	**244-250/259-267**	44.4	0.072	'0'	**1586-1593/1601-1609**	92.6	0.050
7	'0'	**2903-2906/2845-2855**	43.8	0.078	'0'	**244-250/259-267**	88.0	0.060
8	'0'	815-822/792-798	43.0	0.088	'0'	**111-113/148-149/42-50**	70.0	0.151
9	-	-	-	-	'0'	**2903-2906/2845-2855**	67.6	0.178
10	-	-	-	-	'0'	**1068-1075/1084-1088/1045-1046**	63.8	0.238
11	-	-	-	-	'9'	815-822/792-798	56.9	0.463

True motif instances are bolded.

Table 3.2: Comparison between `RNAMotifScan` and `RNAMotifScanX` in identifying C-loop motifs from ribosomal RNA 1S72

	`RNAMotifScan`				`RNAMotifScanX`			
Ranking	chain	Location	Score	P-value	chain	Location	Score	P-value
1	'0'	**1436-1440/1424-1430**	40.9	0.033	'0'	**2760-2764/2716-2722**	90.8	0.014
2	'0'	**2760-2764/2716-2722**	39.1	0.041	'0'	**1004-1009/957-964**	83.8	0.018
3	'0'	1939-1945/1892-1898	38.4	0.044	'0'	**1436-1440/1424-1430**	71.6	0.034
4	'0'	**1004-1009/957-964**	34.4	0.081	'0'	1939-1945/1892-1898	44.7	1.000

True motif instances are bolded.

Table 3.3: Comparison between `RNAMotifScan` and `RNAMotifScanX` in identifying Sarcin-ricin motifs from ribosomal RNA 1S72

	`RNAMotifScan`				`RNAMotifScanX`			
Ranking	chain	Location	Score	P-value	chain	Location	Score	P-value
1	'0'	**211-215/225-228**	42.8	0.007	'0'	**211-215/225-228**	141.0	0.005
2	'0'	**1368-1372/2053-2056**	42.8	0.007	'0'	**1368-1372/2053-2056**	141.0	0.005
3	'0'	**2690-2694/2701-2704**	42.8	0.007	'0'	**2690-2694/2701-2704**	141.0	0.005
4	'9'	**76-80/102-105**	42.0	0.007	'9'	**76-80/102-105**	127.6	0.006
5	'0'	**461-466/475-478**	37.5	0.010	'0'	**173-177/159-162**	118.6	0.008
6	'0'	**380-383/406-408**	34.4	0.013	'0'	**380-383/406-408**	115.2	0.008
7	'0'	**951-955/1012-1016**	33.4	0.015	'0'	**461-466/475-478**	114.6	0.008
8	'0'	**173-177/159-162**	29.8	0.022	'0'	**951-955/1012-1016**	84.8	0.019
9	'0'	2090-2094/2651-2654	26.2	0.037	'0'	**585-590/568-572**	84.4	0.019
10	'0'	1775-1779/1765-1768	25.5	0.042	'0'	**355-360/292-296**	83.8	0.019
11	'0'	1542-1545/1640-1643	21.0	0.117	'0'	**1971-1973/2009-2010**	83.4	0.020
12	'0'	**585-590/568-572**	20.8	0.126	'0'	**1292-1294/911-912**	81.8	0.021
13	'0'	**355-360/292-296**	20.8	0.126	'0'	1775-1779/1765-1768	47.2	0.144

True motif instances are bolded.

Table 3.4: Comparison between `RNAMotifScan` and `RNAMotifScanX` in identifying Reverse kink-turn motifs from ribosomal RNA 1S72

	`RNAMotifScan`				`RNAMotifScanX`			
Ranking	chain	Location	Score	P-value	chain	Location	Score	P-value
1	'0'	**1661-1666/1520-1530**	48.6	0.114	'0'	**1661-1666/1520-1530**	94.7	0.014
2	'0'	**1530-1536/1649-1661**	46.8	0.145	'0'	**1530-1536/1649-1661**	84.1	0.021
3	'9'	74-82/100-107	46.2	0.160	'9'	74-82/100-107	82.9	0.022

True motif instances are bolded.

Table 3.5: Comparison between `RNAMotifScan` and `RNAMotifScanX` in identifying E-loop motifs from ribosomal RNA 1S72

	`RNAMotifScan`				`RNAMotifScanX`			
Ranking	chain	Location	Score	P-value	chain	Location	Score	P-value
1	'0'	**706-708/720-722**	21.2	0.052	'0'	**1543-1545/1640-1642**	64.6	0.010
2	'0'	**1543-1545/1640-1642**	20.6	0.061	'0'	**706-708/720-722**	64.4	0.010
3	'0'	**174-177/159-161**	18.7	0.098	'9'	**100-104/77-82**	54.0	0.016
4	'0'	**663-666/680-683**	18.6	0.100	'0'	**1369-1372/2053-2055**	53.6	0.016
5	'0'	**586-590/568-571**	18.0	0.120	'0'	**214-215/225-226**	53.6	0.016
6	'0'	**356-360/292-295**	18.0	0.120	'0'	**2691-2694/2701-2703**	53.6	0.016
7	'0'	**2691-2694/2701-2703**	17.8	0.130	'0'	**356-360/292-295**	52.9	0.016
8	'0'	**1369-1372/2053-2055**	17.8	0.130	'0'	**952-955/1012-1015**	52.5	0.017
9	'0'	**463-466/475-477**	17.8	0.130	'0'	**1293-1294/911-912**	52.3	0.017
10	'0'	**380-383/406-408**	17.8	0.130	'0'	**174-177/159-161**	52.3	0.017
11	'9'	**77-82/100-104**	17.8	0.130	'0'	**586-590/568-571**	52.3	0.017
12	'0'	2773-2776/2799-2801	17.8	0.133	'0'	**1972-1973/2009-2010**	52.3	0.017
13	-	-	-	-	'0'	**380-383/406-408**	52.3	0.017
14	-	-	-	-	'0'	**463-466/475-477**	49.6	0.019
15	-	-	-	-	'0'	**663-666/680-683**	48.0	0.021
16	-	-	-	-	'0'	2782-2784/2788-2792	46.0	0.023

True motif instances are bolded.

3.3 Results

We have benchmarked the performance of `RNAMotifScanX` by searching five important RNA structural motifs, including the kink-turn [75], C-loop [11, 27, 130, 144], sarcin-ricin [60, 94, 118, 125], reverse kink-turn [1, 2, 122], and E-loop [28, 81] motif, against the *H. marismortui* 50S rRNA [74]. We manually examined the known instances for these motifs, and revised their base-interaction patterns by adding the conserved base-stacking interactions (see Figure 3.4). The 3D structure of the *H. marismortui* 50S is downloaded from PDB [17], with the accession number of 1S72. This 50S rRNA contains a 23S rRNA (chain '0') and a 5S rRNA (chain '9'). The base-interaction annotation for this 50S rRNA is generated by `MC-Annotate` [53] and `RNAVIEW` [147].

The search results of `RNAMotifScan` and `RNAMotifScanX` on the five RNA structural motif families are summarized in Table 3.1 - Table 3.5. In this table, the *bona fide* motif instances are shown in bold, and the motif instances that are newly detected by `RNAMotifScanX` are underlined. Here we give a brief reasoning for accepting these new instances as true predictions. For the kink-turn motif family (Table 3.1), the instances ranked in the 3^{rd} and the 8^{th} place are known motif instances, as identified and described by FR3D [113]. We will discuss the 10^{th} motif instance for more details in later sections. For the sarcin-ricin motif family (Table 3.3), the two instances (the 11^{th} and 12^{th}) have been discovered through a *de novo* clustering approach as partial sarcin-ricin motif instances with ultra conserved structure at the bulged-G region (will be discussed in Chapter 4). For the E-loop motif family (Table 3.5), all newly identified motif instances (the 5^{th}, 8^{th}, 9^{th}, and 12^{th}) are overlapped with known sarcin-ricin motif instances. This is because the E-loop motif family and the sarcin-ricin motif family share significant similarities in both base-interaction pattern

and 3D geometry. For example, the 9th E-loop motif instance was identified as E-loop by `LENCS` (reference [37], a base-pairing pattern-based approach), and it is overlapped with the 12th sarcin-ricin motif instance. The 5th E-loop instance was identified as E-loop by the shape histogram method (reference [6], a 3D geometry-based approach), and it is overlapped with the 1st sarcin-ricin motif instance. Therefore, we consider these motif instances are E-loop related motif instances.

In the following sections, we will show the improvement of `RNAMotifScanX` over its predecessor `RNAMotifScan`. We will first show that `RNAMotifScanX` is able to rank known motif instances on top without including any unrelated motif instances. We will then show that by using the improved P-value estimation, we are able to suggest a universal P-value cutoff that performs well (over 90% of F-measure) on all RNA structural motifs that have been tested. Finally, we will discuss a tandem kink-turn motif instance and a novel kink-turn-like motif instance found by `RNAMotifScanX`, which provide new insights into the understanding of the kink-turn motif family.

3.3.1 Prioritizing the Rankings of True RNA Structural Motif Instances

We summarize the benchmark results of searching kink-turn, C-loop, sarcin-ricin, reverse kink-turn and E-loop motifs in Table 3.1 - Table 3.5, respectively. We can observe that `RNAMotifScanX` is able to prioritize the rankings of the real motif instances before other unrelated motif instances, especially for the C-loop and the sarcin-ricin motif families. Originally, `RNAMotifScan` ranked an unrelated motif instance chain '0', 1939-1945/1892-1898 before a real motif instance chain '0', 1004-1009/957-964, while `RNAMotifScanX` is able to

remove the unrelated motif instance from its top list. Similarly, RNAMotifScan included three unrelated motif instances in its sarcin-ricin motif search results (which are ranked 9th, 10th, and 11th), while RNAMotifScanX has also removed these unrelated motif instances. In this case, RNAMotifScanX has significantly improved the identification accuracy upon its predecessor RNAMotifScan.

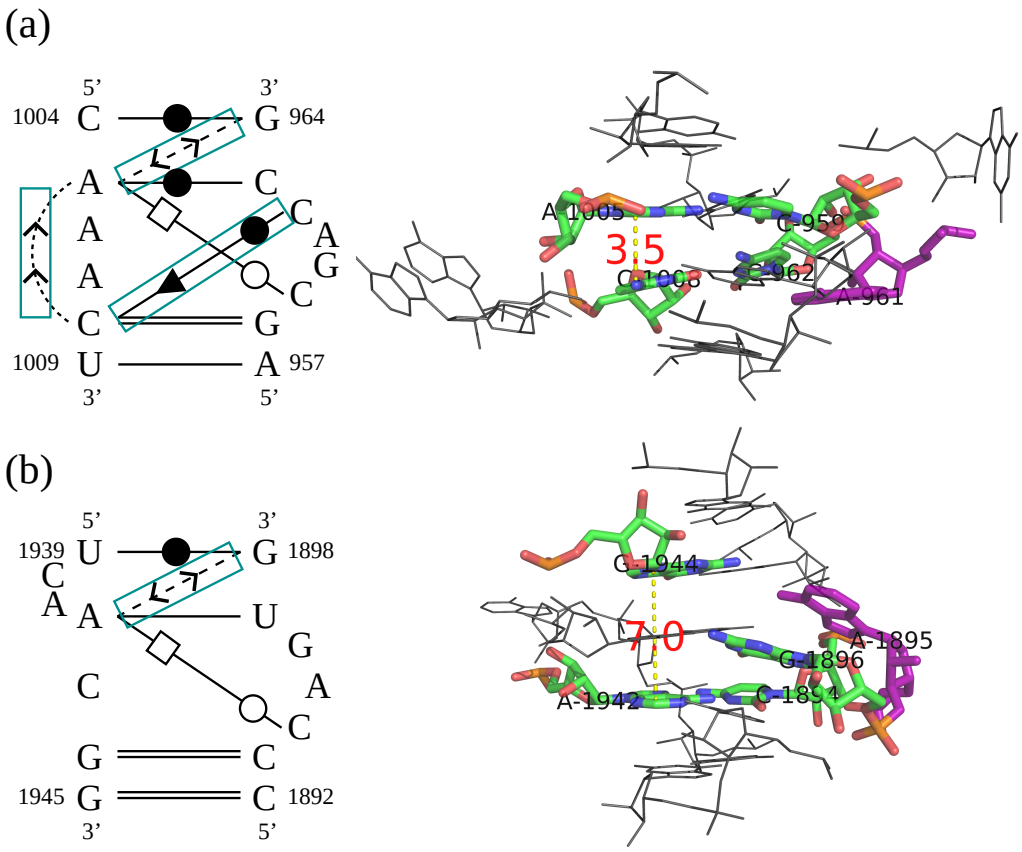

Figure 3.5: A real example showing that the base-stacking information and the optimal alignment of crossing base pairs help to improve C-loop motif identification accuracy. The base-interaction pattern (left panel) and 3D structure (right panel) of a C-loop motif instance found in (a) 1S72, chain '0', 1004-1009/957-964 and an unrelated motif instance found in (b) 1S72, chain '0', 1939-1945/1892-1898. In the left panels, the boxed base interactions are newly detected by RNAMotifScanX (but not RNAMotifScan). In the right panels, the nucleotides in green are the stacking nucleotides in (a), or their corresponding nucleotides in (b). The red measurements indicate the distances (3.5 Å in (a) and 7.0 Å in (b)) between the corresponding nucleotides. The nucleotides in purple correspond to the adenine residues that should be positioned in the minor groove.

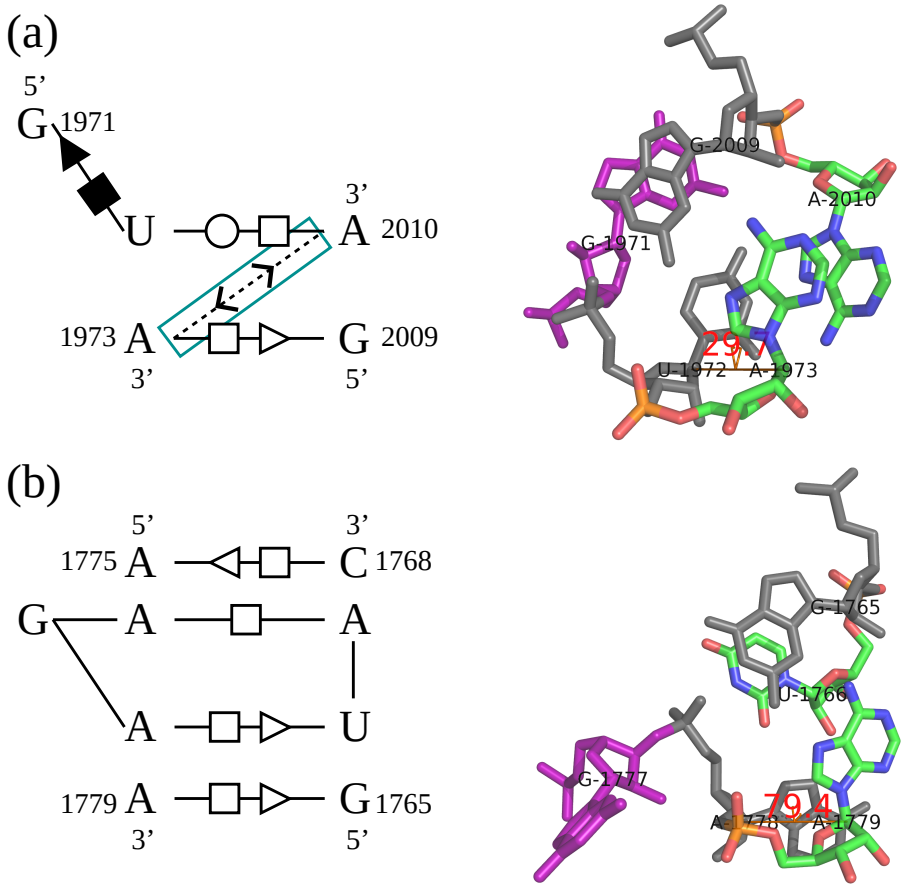

Figure 3.6: A real example showing that the base-stacking information is capable of improving sarcin-ricin motif identification accuracy. The base-interaction pattern (left panel) and 3D structure (right panel) of a sarcin-ricin motif instance found in (a) 1S72, chain '0', 1971-1973/2009-2010 and an unrelated motif instance found in (b) 1S72, chain '0', 1775-1779/1765-1768. In the left panels, the boxed base interactions are newly detected by `RNAMotifScanX` (but not `RNAMotifScan`). In the right panels, the nucleotides in green color are the stacking nucleotides in (a), or their corresponding nucleotides in (b). The red measurements indicate the dihedral angles (29.7° in (a) and 79.4° in (b)) formed between the two vectors defined by the C1' atom and its bonding nitrogen atoms of two consecutive nucleotides (U1972, A1973 in (a) and A1778, A1779 in (b)). The nucleotides in purple correspond to the bulged guanine residues.

Table 3.6: The optimal performance of `RNAMotifScan` and `RNAMotifScanX` with a universal P-value cutoff

	`RNAMotifScan`			`RNAMotifScanX`		
Motif	Specificity	Sensitivity	F-measure	Specificity	Sensitivity	F-measure
Kink-turn	0.875 (7/8)	1.000 (7/7)	**0.933**	1.000 (4/4)	0.571 (4/7)	0.727
C-loop	0.375 (3/8)	1.000 (3/3)	0.545	1.000 (2/2)	0.667 (2/3)	**0.800**
Sarcin-ricin	0.769 (10/13)	1.000 (10/10)	0.869	1.000 (10/10)	1.000 (10/10)	**1.000**
Reverse kink-turn	1.000 (1/1)	0.500 (1/2)	0.667	1.000 (2/2)	1.000 (2/2)	**1.000**
E-loop	1.000 (11/11)	1.000 (11/11)	**1.000**	1.000 (11/11)	1.000 (11/11)	**1.000**
Average performance	0.780 (32/41)	0.970 (32/33)	0.865	1.000 (29/29)	0.879 (29/33)	**0.935**

The P-value cutoff for `RNAMotifScan` is 0.130, and for `RNAMotifScan` is 0.021.
F-measure is computed as the follow: F-measure $= \frac{2 * \text{Specificity} * \text{Sensitivity}}{\text{Sensitivity} + \text{Specificity}}$. The higher performance of F-measure is bolded.
The novel motif instance found by `RNAMotifScanX` is not considered for the performances shown in this table.

The identification improvement is made by the inclusion of base-stacking information and the new branch-and-bound algorithm that optimally align the crossing base interactions. We use examples of the C-loop and the sarcin-ricin motif search to demonstrate these two advantages of `RNAMotifScanX`. In Figure 3.5, we show the true C-loop instance chain '0', 1004-1009/957-964 that was ranked 4$^{\text{th}}$ by `RNAMotifScan` (Figure 3.5a), and the unrelated instance chain '0', 1939-1945/1892-1898 that was ranked 3$^{\text{rd}}$ by `RNAMotifScan` (Figure 3.5b). Originally, `RNAMotifScan` identified four isosteric base pairs (three canonical and one non-canonical) in the first instance, but five isosteric base pairs (four canonical and one non-canonical) in the second instance. Therefore, the first instance is ranked below the second instance. While base-stacking information is incorporated, and the crossing base interactions are optimally aligned, `RNAMotifScanX` identified two more conserved base-stacking interactions and one more isosteric non-canonical base pair from the first instance, but only one more conserved base-stacking interaction from the second (see the boxed base interactions in left panels of Figure 3.5).

The base-stacking interaction (outward A1005-C1008) and the non-canonical base pair (*cis* S/W C1008-C962) aligned in the first instance but not in the second instance are critical for the formation of the C-loop motif instance. The presence of the base-stacking interaction in the first instance indicates that the spatial distance between A1005 and C1008 is small (3.5Å as shown in Figure 3.5a, right panel). The small distance between these two nucleotides facilitates the formation of the two crossing base pairs (*trans* H/W A1005-C959 and *cis* S/W C1008-C962, as shown by the green nucleotides in Figure 3.5a, right panel). Notice that one of these crossing base pairs *cis* S/W C1008-C962) happens to be the base pair that only conserves in the first instance. The crossing base pairs position the adenine base (A961, as shown in purple) within the minor groove (below the green nucleotides, as

shown in Figure 3.5a, right panel), which is critical for the molecular function of the C-loop motif [130]. On the other hand, the absence of the base-stacking interaction in the second motif instance indicates that the distance between the two corresponding nucleotides (A1942-G1944) is large (7.0Å as shown in Figure 3.5b, right panel). In this case, one of the two crossing base pairs is missing, and the corresponding adenine base (A1895, as shown in purple) is not properly positioned in the minor groove (above the green nucleotides, as shown in Figure 3.5b, right panel). In this case, `RNAMotifScanX` has distinguished true and unrelated motif instances through the consideration of base-stacking information and the optimal alignment of crossing base pairs.

Similarly, the following example shows how `RNAMotifScanX` improves the identification accuracy of the sarcin-ricin motif through the incorporation of base-stacking information. We show a true instance that is newly identified by `RNAMotifScanX` in Figure 3.6a, and an unrelated motif instance that was ranked higher than two true sarcin-ricin instance in Figure 3.6b. Originally, `RNAMotifScan` aligned only three non-canonical base pairs in the first instance (all are isosteric) in the first instance, but four (three isosteric and one non-isosteric) in the second instance. Therefore, `RNAMotifScan` ranks the second instance before the first instance. When the base-stacking information is incorporated, `RNAMotifScanX` is able to identify one more conserved base-stacking interaction (A1973-A2010, see Figure 3.6a, left panel) from the first instance. In this case, the true motif instance is ranked higher than the unrelated motif instance by `RNAMotifScanX`.

The identified base-stacking interaction is also important in defining the sarcin-ricin motif. In the first motif instance, two non-canonical base pairs (*trans* W/H U1972-A2010 and *trans* H/S A1973-G2009) are formed consecutively. The base-stacking interaction is formed by two nucleotides A2010 and A1973, each from one of these two base pairs. In order to form

such interaction, the relative rotation of the two consecutive nucleotides (U1972 and A1973) around the backbone should be small. We measured the dihedral angle between the two vectors defined by the two nucleotides' C1' atoms and their bonding nitrogen atoms in the bases. The dihedral angle in the first instance is 29.7° (see Figure 3.6a, right panel), which is consistent with our conjecture. On the other hand, without the pressure to form such base-stacking interaction, the corresponding dihedral angle in the second instance is 79.4°(see Figure 3.6b, right panel). The significant difference in the torsion angles between the two instances also affect the positioning of its directly adjacent guanine residues (G1971 and G1777). In this first instance, G1971 is folded inward to form the *cis* S/H base pair with U1972. While in the second instance, G1777 is flipped outward, as shown in Figure 3.6b, right panel. The guanine nucleotide is critical for the molecular function of the sarcin-ricin motif, as indicated by its alternative name: the G-bulge motif. As a result, the second instance is unlikely to be a real sarcin-ricin motif instance. Such example shows the importance of the base-stacking information in modeling the sarcin-ricin motif families.

3.3.2 Universal P-value Cutoff Towards Automatic Identification of RNA Structural Motif Instances

Besides the performance improvement through prioritizing the ranking of related motif instances, we also expect to show the new P-value estimation is more reasonable, and a universal P-value cutoff will generate satisfying results for all types of RNA structural motif families.

We summarize the optimal performances of `RNAMotifScan` and `RNAMotifScanX` under a universal P-value cutoff in Table 3.6. The optimal performance of `RNAMotifScan` is achieved at a P-value cutoff of 0.130, and that of `RNAMotifScanX` is achieved at a P-value cutoff of 0.021. Note that the reference data set we used to compute the sensitivity and specificity was generated based on the search results of `RNAMotifScan`, and the related motif instances that are newly detected by `RNAMotifScanX` were not counted. In this case, the final benchmark results will favor `RNAMotifScan`. Even with such benchmark design, we can still observe a significant improvement on the overall F-measure (see Table 3.6).

Notably, `RNAMotifScanX` is able to achieve over 93% of average accuracy with universal P-value cutoff, and at the same time achieve 100% specificity. This means that when the correct P-value cutoff is provided, `RNAMotifScanX` will identify motif instances with high confidence. In this case, the time-consuming manual validation can be avoided, and such advantage is highly desirable for full automatic identification of RNA structural motifs. On the other hand, the universal P-value cutoff for `RNAMotifScan` still includes several unrelated motifs, making the manual validation step inevitable. Note that we are unable to apply a more stringent cutoff to `RNAMotifScan` search results without dramatic decreasing the overall performance (using any P-value cutoff less than 0.130 will miss at least 5 true E-loop motif instances, as shown in Table 3.5). In summary, the P-value estimation strategy used by `RNAMotifScanX` is capable of providing a universal P-value cutoff for all types of RNA structural motif families with high sensitivity and specificity.

Figure 3.7: A tandem kink-turn motif instance found at 1S72, chain '0', 2818-2856/2901-2930/2667-2671. The two 'kink' regions in both kink-turn motif instances are colored orange. The tandem kink-turn motif instance form RNA-protein interaction with the ribosomal protein L3P (1S72, chain B), which is shown in blue. Both individual kink-turn motif instances interact with the L3P protein, suggesting the aggregation of these two motif instances is necessary for the binding of L3P.

3.3.3 New Insights into the Kink-turn Motif Family

3.3.3.1 Tandem Kink-turn Motif Instance

We identified a tandem kink-turn motif instance from the kink-turn search results generated by `RNAMotifScanX`. Two individual kink-turn motif instances found at locations 1S72, chain '0', 2911-2914/2667-2669/2820-2829 and 1S72, chain '0', 2903-2906/2845-2855 forms a tandem instance, with their NC helices (non-canonical helix [75]) connecting to each other coaxially (Figure 3.7). The C helices (canonical helix [75]) of the individual kink-turn motif instances rotate around the NC helix axis in different directions, leaving a $\sim 90^{\circ}$ torsion angle between them. The majority of the nucleotide residues in the two C helices form RNA-protein interaction with the ribosomal protein L3P (1S72, chain B, see Figure 3.7). In addition, 27.6% (21/76) of total nucleotide residues that interact with L3P can be found within the two C helices, indicating an important role of this tandem kink-turn motif instance in the binding of the L3P ribosomal protein (RNA-protein interaction annotation is taken from the Comparative RNA Website (CRW), reference [24]). Similar tandem motif instance has also been observed for the reverse kink-turn motif, suggesting that such motif aggregation phenomenon may not be random. Nevertheless, whether the cooperation of the individual motif instances will lead to novel molecular function requires further experimental studies.

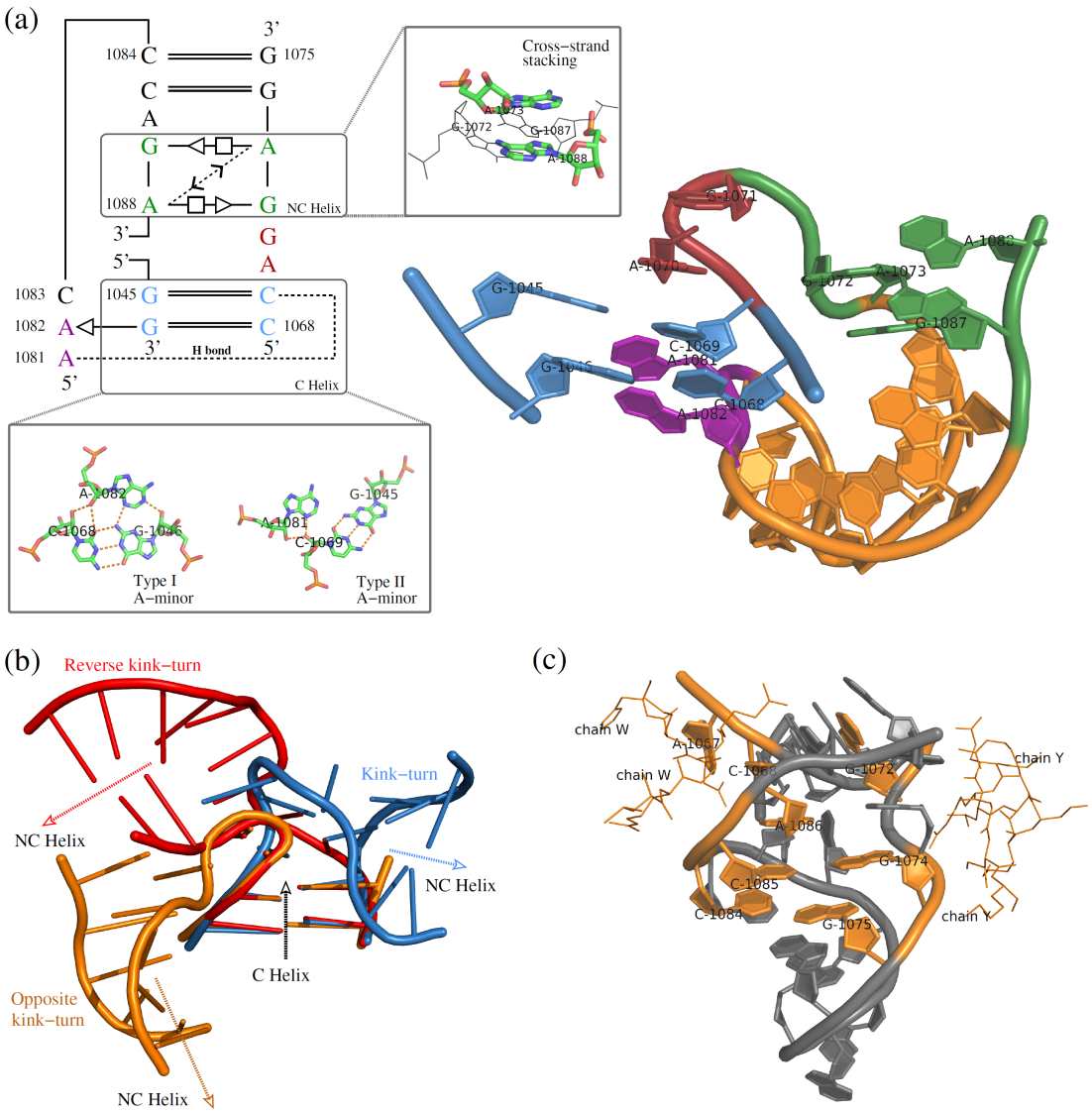

Figure 3.8: The new kink-turn-like motif instance found at 1S72, chain '0', 1068-1075/1081-1088/1045-1046. (a) The base-interaction pattern and the 3D structure of this motif instance. Different colors in the base-interaction pattern and 3D structure depicts different regions of this motif instances: Green - the NC helix where the cross-strand A-A stacking is found; Red - the bulge loop that corresponds to the kink region of the motif instance; Blue - the C helix where two A-minor interactions are found (a type I and a type II A-minor interaction); Purple - the adenine residues that participate in the two A-minor interactions. (b) Superimposition of the C helices of a kink-turn (blue), a reverse kink-turn (red), and the new kink-turn-like motif instance (orange). The NC helix of the kink-turn turns leftwards, while that of the reverse kink-turn turns rightwards and that of the new kink-turn-like turns downwards. (c) The new kink-turn-like motif instance interacts with two ribosomal proteins, L30P (1S72, chain W) and L32E (1S72, chain Y), simultaneously. The nucleic acid residues that interact with the ribosomal proteins are colored orange.

3.3.3.2 The New Kink-turn-like Motif: Opposite Kink-turn

During the search of kink-turn motif instances in 1S72, we discovered a kink-turn-like motif instance at 1S72, chain '0', 1068-1075/1084-1088/1045-1046. The motif instance is ranked in the 10^{th} place, indicating its large structural variance compared to regular kink-turn motif instances. However, several key features of the kink-turn motif is preserved in this instance, and according to which we consider this motif as a related motif instance. First, both the C helix (colored blue in Figure 3.8a) and NC helix (colored green in Figure 3.8a) can be found in this motif instance. As shown in Figure 3.8a, the C helix contains two G-C canonical base pairs and the NC helix contains two G-A sheared base pairs, and both of which are consistent with the kink-turn consensus structure. Second, similar to the kink-turn motif, this motif instance is also stabilized through the cross-strand A-A stacking in the NC helix and A-minor interaction in the C helix. The difference is that the new kink-turn-like motif instance forms two A-minor interactions (a type I A-minor interaction, as shown by the *trans* S/S A1082-G1046 base pair in Figure 3.8a, and a type II A-minor interaction, as shown by the hydrogen bond interaction between A1081 and C1069 in Figure 3.8a), instead of only one in the regular kink-turn motif instances. Third, the kink region is formed by two unpaired nucleotides (A1070 and G1071, the red residues in Figure 3.8a). The kink region of the kink-turn-like motif is, however, not found in the asymmetric internal loop motif as regular kink-turn motif, as the shorter strand of the asymmetric internal loop is interrupted by the discontinuity of the corresponding strands.

The superimposition of the C helices of a kink-turn, a reverse kink-turn, and the kink-turn-like motif instance clearly depicts the direction of the turn of its two helices. In Figure 3.8b, the kink-turn motif instance is colored blue, the reverse kink-turn motif instance is colored

red, and the new kink-turn-like motif instance is colored orange. Surprisingly, the turn of the new kink-turn-like motif instance neither follows that of the kink-turn motif nor the reverse kink-turn motif. Instead of turning to the left like the reverse kink-turn (red arrow in Figure 3.8b) or to the right like the kink-turn motif (blue arrow in Figure 3.8b), the NC helix of the new kink-turn-like motif turns downwards (orange arrow in Figure 3.8b), and to the opposite direction of its C helix. In this sense, we name the new kink-turn-like motif the 'opposite kink-turn' motif.

Similar to the kink-turn motif, the opposite kink-turn motif also exhibit potential molecular function in protein binding. The opposite kink-turn motif instance interacts with two ribosomal protein simultaneously, i.e. the ribosomal protein L30P (1S72, chain W) and L32E (1S72, chain Y) (Figure 3.8c). Interestingly, the nucleotide residues that participate in the RNA-protein interactions are mostly found near the NC helix where the cross-strand A-A stacking interaction is present. For example, C1084, C1085 and A1086 interact with the L30P protein, and G1072, G1074 and G1075 interact with the L32E protein. In this case, the protein binding scheme of this opposite kink-turn motif instance is consistent with that of the kink-turn motif, which may facilitate the RNA-protein interaction through its fattened minor groove of the NC helix [75].

These evidences suggest the strong similarity between the opposite kink-turn motif and the kink-turn motif. The large structural variation of the opposite kink-turn motif can be explained using the plasticity of the kink-turn motif [5, 32]. In this case, it would be interesting to experimentally verify the molecular function of the opposite kink-turn motif, and search for other kink-turn-like motifs that exhibit different structural configurations.

3.4 Discussion

In this chapter, we have developed a new RNA structural motif search tool `RNAMotifScanX` by incorporating the base-stacking information. The current implementation of `RNAMotifScanX` adopts a branch-and-bound technique to maintain its execution in a reasonable time. Our test on searching the kink-turn motif consensus structure against the 50S rRNA 1S72 took less than 1 hour to finish (single-core configuration). We expect to improve the running time of `RNAMotifScanX` for future online service purpose using the following strategies. First, we will apply a filtering step, where the candidate motif instances that share no isosteric base pair or conserved base-stacking interaction will be discarded without detailed alignment. Second, as the alignment between the consensus structure and the candidate instances are independent, we will introduce a multi-threaded feature to the future version of `RNAMotifScanX` so as to process the candidates in parallel. Once these two speedup techniques are implemented, we will use `RNAMoitfScanX` to scan the PDB [17] and update our registration of motif instances.

More importantly, we have proved the importance of base-stacking information in modeling RNA structural motifs. As we have shown in Table 3.1 - Table 3.5 in the Results section, the score difference between the last true motif instance and the first unrelated motif instance has also been significantly increased as compared to `RNAMotifScan`. In addition to the advantage of easy separation of true and unrelated motif instances, such experimental results also suggest that the base-stacking interaction is highly specific for the given motif families. Otherwise, we should observe concurrent alignment score increment in both of the true and the unrelated motif instances. We can also see that the score difference for the kink-turn (Table 3.1), C-loop (Table 3.2), and sarcin-ricin (Table 3.3) motif is large, while the

difference for reverse kink-turn (Table 3.4) and E-loop (Table 3.5) motif is marginal. This is because we have not identified any conserved base-stacking interaction in the reverse kink-turn and E-loop motif families, and their search models are not revised (see Figure 3.4). We can also observe that the score difference is highly correlated with the number of base-stacking interactions that have been incorporated into the search model. For example, one base-stacking interaction is introduced into the kink-turn consensus structure and the score difference is 6.9. When two base-stacking interactions are introduced into the C-loop and sarcin-ricin consensus structures, the score difference is increased to 26.9 and 34.6, respectively. These evidences suggest that base-stacking information is highly specific in defining RNA structural motif families, and is perhaps more powerful in distinguishing true and unrelated motif instances than the base-pairing information.

CHAPTER 4: *DE NOVO* CLUSTERING OF RNA STRUCTURAL MOTIFS

In Chapter 3, we have presented `RNAMotifScanX`, an enhanced version of `RNAMotifScan` which considers both base-pairing and base-stacking information. Using `RNAMotifScanX`, we can search query RNA structural motifs with much higher accuracy and sensitivity. However, a limitation of the query-based search approach is that it heavily relies on the query model, thus it cannot identify novel RNA structural motif families. To address this issue, we introduce the *de novo* RNA structural motif clustering problem in this chapter, which has the ability to discover novel RNA structural motif families. We devise a clustering pipeline called `RNAMSC` based on `RNAMotifScan` (for its higher computational efficiency compared to `RNAMotifScanX`). Novel RNA structural motif families that have been discovered through the clustering analysis of ribosomal RNAs (5S, 16S, and 23S) will also be discussed in details.

4.1 RNA Structural Motif Identification without Explicit Query

The computational identification of RNA structural motifs can become much more difficult when there is no explicitly defined query. This problem, also referred as the *de novo* motif identification problem, is usually solved using clustering approaches that require no explicit query information. `COMPADRES` [136], a *de novo* clustering method developed based

on PRIMOS [44], has successfully identified four new structural motif families from the resolved RNA 3D structures in Protein Data Bank (PDB) [17]. However, the motif families identified by COMPADRES are mostly short motifs with rigid 3D topologies, while larger and more complicated motifs were not considered. In addition, the lack of conserved base interaction pattern for the newly identified motifs makes further modeling, search and functional inference of these motifs rather difficult [37]. As a result, base-pairing patterns should also be considered in *de novo* structural motif identification.

Recently, Djelloul and Denise have devised a clustering approach that purely considers base-pairing pattern for *de novo* RNA structural motif identification [37]. In this chapter, we refer this method as the LENCS (Longest Extensible Non-Canonical Substructure) method. They transformed each candidate structural motif instance into a base-pairing graph, and applied graph isomorphism algorithm to identify maximum common subgraphs. After pairwise comparison, the structural fragments were organized using hierarchical clustering, and potential motif clusters were extracted by applying a universal cutoff. Although LENCS has successfully rediscovered many known motifs and suggested potential novel motifs, the graph isomorphism restriction makes it impossible to consider RNA structural motifs with base-pair variations. Besides, the LENCS method completely ignored the sequences of the motifs, hence difficult to correctly incorporate base-pair isostericity information [80].

We have developed RNAMotifScan to account for these problems (discussed in Chapter 2), and expect to develop a more accurate clustering framework by incorporating RNAMotifScan. In addition, we also try to tackle three important issues in RNA structural motif clustering. First, it is well known that the annotation tools may make mistakes in base-pair prediction due to inadequate resolution. Although this may not be an issue in model-based search application (as the query model is hand-curated and thus can represent the complete base-

pairing pattern), it can significantly affect clustering analysis since the erroneous base-pair predictions may happen in both motif instances that are being compared. Second, the `LENCS` method only considers the fraction of matched base pairs between motif instances, but does not distinguish the importance of the matching. For example, the *trans* H/SE pair can be found in many motifs such as kink-turn, sarcin-ricin and tandem-sheared motifs, while the *cis* H/SE pair is much less frequent. In this case, matching *cis* H/SE pairs should be more informative than matching *trans* H/SE pairs. Finally, the hierarchical clustering approach applied by the `LENCS` method is not suitable for large-sized data sets, since it would be difficult to manually examine the huge hierarchical tree to determine the optimal cutting level.

To account for the first issue, we combined the base-pair predictions made by two popular annotation tools: `RNAVIEW` [147] and `MC-Annotate` [53]. In this way, we were likely to include all true base-pairing interactions into the compiled candidate motif instances. `RNAMotifScan` is then responsible for identifying the optimal matching between these predictions and discarding additional base pairs with moderate penalty. To solve the second issue, we developed a statistical inference framework that can be used to measure the significance of the matchings. Each candidate motif instance was aligned to a set of artificial motif instances that simulate random structural segments from ribosomal RNAs. Consider the example in the previous paragraph, although we do not distinguish the alignment score between matching *trans* and *cis* H/SE pairs, we can expect lower P-value assigned to the matching of *cis* H/SE pairs. This is because *cis* H/SE pairs are much less frequently found, resulting in lower background alignment scores associated with the motif instances that contain this base pair and, therefore, more significant P-values for a match. Finally, to make the clustering analysis extensible to large-size data sets, we applied the `CAST` (Cluster Affinity Search Technique)-

like [16] clique finding algorithm that can automatically generate individual clusters given only a universal P-value cutoff.

We applied our new clustering framework on two data sets (one for hairpin loop instances and the other for internal loop, bulge loop and junction loop instances, see Materials and Methods section) that contain 5S (*Haloarcula marismortui*, PDBid: 1S72, chain '9'), 16S (*Thermus thermophilus*, PDBid: 1J5E, chain A) and 23S (*Haloarcula marismortui*, PDBid: 1S72, chain '0') ribosomal RNAs. We have identified totally 44 clusters (8 from the hairpin loop data set and 36 from the internal loop data set). These clusters define many known RNA structural motifs such as GNRA tetraloop [145], kink-turn [75], C-loop [11, 27, 130, 144], sarcin-ricin [60, 94, 118, 125], reverse kink-turn [122], hook-turn [124], E-loop [28, 81] and tandem-sheared [31] motifs. The performance of our clustering framework shows significant improvement over the `LENCS` method. Specifically, the F-measure has been increased from 69.1% to 82.6%. Besides, we also identified several new occurrences of these known motifs. Finally, we also present three clusters corresponding to novel motif families that have not been characterized before. All clusters are sorted based on average P-values that indicate the in-cluster structural similarities.

4.2 Materials and Methods

4.2.1 Data Preparation

The resolved ribosomal RNA subunit structures (1S72 and 1J5E) were downloaded from PDB [17]. The base pairs were annotated by `RNAVIEW` [147] and `MC-Annotate` [53]. We com-

bined (union) the annotations from both tools to generate the final annotation. The conflict predictions (different edge or orientation annotations for the same base pair) were resolved by taking the annotations from `MC-Annotate`. All non-canonical base pairs were temporarily discarded to reveal the general sketch of the A-form helices in the structures. Pseudo-knots were then removed using `K2N` web server [117]. Lone pairs were further removed to avoid accidental destruction of potential motifs. Finally, regions corresponding to hairpin loops, internal loops, bulge loops or junction loops [84] were identified from the resulting nested structures and all base pairs within these regions were recovered to construct candidate motif instances (similar to `LENCS` [37]). The candidate instances that contain no non-canonical base pair were removed.

Candidate motif instances from 5S, 16S and 23S rRNAs were compiled into two data sets, one for hairpin loops and the other for internal loops, bulge loops and junction loops (we will call this data set internal loop data set for short). Since sequence conservation in hairpin loop motifs is also very important in defining their functionalities, higher sequence weight should be applied for this data set. The hairpin loop data set contains 33 candidate instances and the internal loop data set contains 157 candidate instances. To account for different concatenation orders the strands, the symmetric counterpart of each motif instance in internal loop data set is also included.

4.2.2 Aligning Structural Components using `RNAMotifScan`

We applied `RNAMotifScan` to measure the structural similarity between two candidate motif instances. `RNAMotifScan` matches two motifs instances by a dynamic programming approach

which takes into account base-pair isostericity. For the internal loop data set, the sequence weight was set to 0.2 and the structure weight was set to 0.8. while for the hairpin loop data set, we raised the sequence weight to 0.4 and lowered the structure weight to 0.6. Because the hairpin loop motifs are usually defined by their lengths (e.g., tetraloop and hexaloop), we also doubled the default gap penalty for hairpin loop clustering. Other parameters were set to default.

4.2.3 Generating Random Structural Motif Instances

Given a candidate instance, we aim at generating a number of random motif instances that have similar length (allowing ±20% fluctuation) with the candidate instance and base-pairing pattern with the ribosomal RNAs background. Our statistics indicate that in ribosomal RNAs, the base pair ratio (the number of canonical and non-canonical base pairs over the length of the sequence) is ∼50% (specifically, 51.7% for 5S rRNA, 50.0% for 16S rRNA, and 50.2% for 23S rRNA). Among these base pairs, ∼15% of them correspond to non-nested base pairs (specifically, 15.5% for 5S rRNA, 14.6% for 16S rRNA and 16.9% for 23S rRNA), while the others form nested base pairs. (This statistic is solely based on MC-Annotate predicted base pairs.)

Since random sampling of existing structural segments from database may not result in enough randomness and sometimes introduce bias [61], we developed the following method to generate random motif instances. Given the base-pair distribution for the ribosomal RNAs and assume the length of the random motif instance is n (predetermined based on the length of the candidate instance), we first build a perfectly stacked helix with $85\% * n/2$

base pairs (with the same base-pair frequency as the background). Then we randomly insert $15\% * n$ unpaired nucleotides into the helix (with the same nucleotide frequency as the background). Finally, we add $15\% * n/2$ non-nested base pairs (also with the same base-pair frequency as the background) by randomly selecting two nucleotides from the constructed motif instance.

4.2.4 Extracting Significant Clusters

Upon the finishing of all-against-all pairwise alignments, a P-value was assigned for each alignment score. Alignment score distribution regarding each candidate instance was simulated by aligning it to a number of random instances generated using the method described above. The P-values were computed using optimal fitting that assumed general extreme value distribution (with MATLAB built-in function 'gevfit'). Since each alignment score is associated with two P-values (that are computed from both candidate instances' background score distributions), the higher P-value was assigned to ensure specificity.

After the computation of P-values, the all-against-all alignment scores were summarized into a graph, where the nodes represent the candidate motif instances and the edges indicate pairwise structural similarities (denoted by P-values). We extracted all strongly connected subgraphs by applying a `CAST`-like clique finding algorithm [16]. The P-value cutoff was set to $10^{-3.5}$ (empirically determined) for both hairpin loop data set and internal loop data set.

4.3 Results

We have identified 8 clusters from the hairpin loop data set and 36 clusters from the internal loop data set. (If two clusters are completely symmetric due to the inclusion of both strand orientations, only one of them is retained.) The clusters are sorted by their average P-values. To describe the results more clearly, we represent each cluster with a label of the data set ('CH' for the hairpin loop data set and 'CL' for the internal loop data set) followed by its ranking. For example, the kink-turn cluster, CL15, indicates that it was identified from the internal loop data set and ranked 15^{th} by its average P-value. All naming and representation of base pairs follow the fashion proposed by Leontis and Westhof [82]. The 3D structure figures were prepared using `PyMol` (`http://www.pymol.org`).

In this section, we will first discuss the clustering results regarding currently known motifs and present discovery of their new instances. We will then show three potential novel motif families revealed by our clustering analysis. Due to the limitation of space, many meaningful clusters were not discussed in this section. For instance, cluster CH2 represents the UUCG tetraloop motif [47], and cluster CL3 represents an extremely complex base-pairing pattern where four base pairs are formed within only four nucleotides. We anticipate that these clusters can also provide useful information for RNA structural motif studies.

Table 4.1: Comparison between two base-pairing pattern based clustering methods: `RNAMSC` (`RNAMotifScan` based Clustering) and `LENCS`

			`RNAMSC`			`LENCS`		
Motif	Cluster ID	Novel[1]	Sensitivity[2]	Specificity[3]	F-m.[4]	Sensitivity	Specificity	F-m.
GNGA Tetraloop	CH1	0	72.7% (8/11)	100% (8/8)	**84.2%**	-	-	-
GNAA Tetraloop	CH3	1	63.6% (14/22)	93.3% (14/15)	**75.7%**	-	-	-
Kink-turn	CL15	0	50.0% (5/10)	100% (5/5)	**66.7%**	20.0% (2/10)	100% (2/2)	33.3%
C-loop	CL24	0	75.0% (3/4)	100% (3/3)	**85.7%**	50.0% (2/4)	100% (2/2)	66.7%
Sarcin-ricin	CL13	3	100% (12/12)	100% (12/12)	**100%**	66.7% (8/12)	100% (8/8)	80.0%
Reverse Kink-turn	CL18	0	100% (3/3)	100% (3/3)	**100%**	100% (3/3)	42.8% (3/7)	59.9%
Hook-turn	CL17	0	66.7% (2/3)	100% (2/2)	**80.2%**	100% (3/3)	60.0% (3/5)	75.0%
E-loop	CL19	0	100% (4/4)	66.7% (4/6)	**80.0%**	100% (4/4)	57.1% (4/7)	72.7%
Tandem-sheared	CL23	1	33.3% (2/6)	100% (2/2)	49.6%	100% (6/6)	75.0% (6/8)	**85.7%**
Average performance[5]			73.8% (31/42)	93.9% (31/33)	**82.6%**	66.7% (28/42)	71.8% (28/39)	69.1%

1: The novel instances are discussed in detail in corresponding sections. These instances are not counted for performance assessment.

2: Expression in parenthesis corresponds to number of true positive over all known instances.

3: Expression in parenthesis corresponds to number of true positive over cluster size.

4: F-m. (F-measure) = 2 $*$ Sensitivity $*$ Specificity / (Sensitivity + Specificity). The higher performance is bolded.

5: The average performance assessment does not include GNGA and GNAA tetraloop, since they were not identified by `LENCS` method.

4.3.1 Clustering of Known Motifs and Their New Instances

We have identified several clusters that correspond to known motifs including GNRA tetraloop, kink-turn, C-loop, sarcin-ricin, reversed kink-turn, hook-turn, E-loop and tandem sheared motifs. The clustering results of these known motifs and corresponding results generated by `LENCS` method are summarized in Table 4.1. Our clustering method, `RNAMSC` (`RNAMotifScan` based Clustering), shows generally higher performance comparing to the `LENCS` method. The clustering results for these known motif families will be discussed separately below.

4.3.1.1 GNRA tetraloop

The GNRA tetraloop is an RNA structural motif in the hairpin loop region featured by its consensus sequence. The motif is found to interact with proteins [146] or other RNA structural elements [39, 99]. `FR3D` identified 21 GNRA tetraloop motif instances from 1S72 23S rRNA and 12 from 1J5E 16S rRNA. Our clustering method separates the GNRA tetraloop into two clusters: CH1 and CH3. The cluster CH1 contains tetraloops with consensus sequence 'GNGA' and the cluster CH3 contains tetraloops with consensus sequence 'GNAA'. The separation of the GNRA tetraloop motif is due to the strict universal P-value cutoff applied. The clustering performances of the two sets of GNRA tetraloop motif are summarized in Table 4.1. One potential novel GNAA tetraloop instance has been identified in cluster CH3. This novel instance and a well-established GNRA tetraloop instance are shown in Figure 4.1. The base-pairing patterns and 3D geometries of these two instances are very similar.

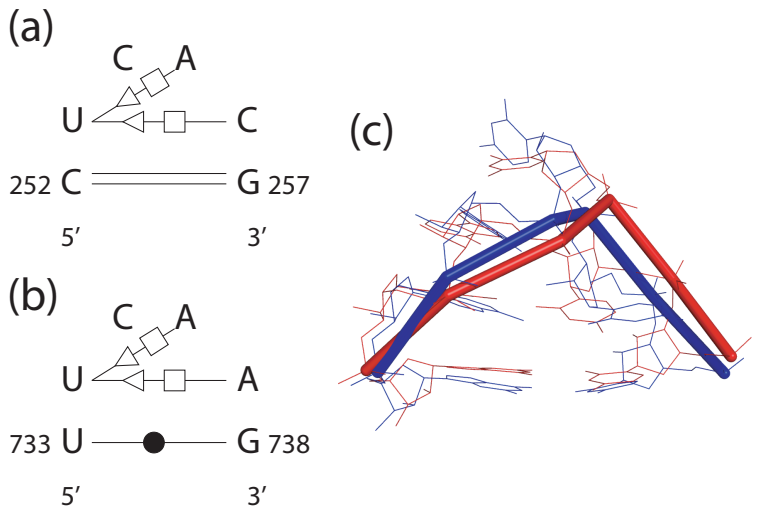

Figure 4.1: The base-pairing patterns and superimposition of two GNRA tetraloop motif instances clustered in CH3. (a) A known GNRA tetraloop instance in 1S72, chain '0', 252-257. (b) The novel GNRA tetraloop instance in 1S72, chain '0', 733-738. (c) The superimposition between these two motif instances (red: (a); blue: (b)).

Several GNRA instances were missed due to two major reasons: unusual base-pair replacement and nucleotide insertion. For example, the GNRA tetraloop instance 1S72, chain '0', 1326-1331 was missed due to the fact that the G1327-A1330 sheared pair is replaced by *trans* W/H pair, while the instances 1S72, chain '0', 1706-1712 and 1J5E, chain A, 691-696 were missed because the closing canonical pair is replaced by sheared pairs. Furthermore, the instance 1J5E, chain A, 726-731 was missed due to the deletion of base pair G727-A729. The GNRA tetraloop instances 1S72, chain '0', 481-487, 493-499, 1054-1060, 1275-1281, 1468-1474 and 1793-1799 were missed due to one nucleotide insertion within the hairpin loop. The other missed instances, 1J5E, chain A, 1030A-1030D, was not included into the candidate set for its irregular nucleotide indexing.

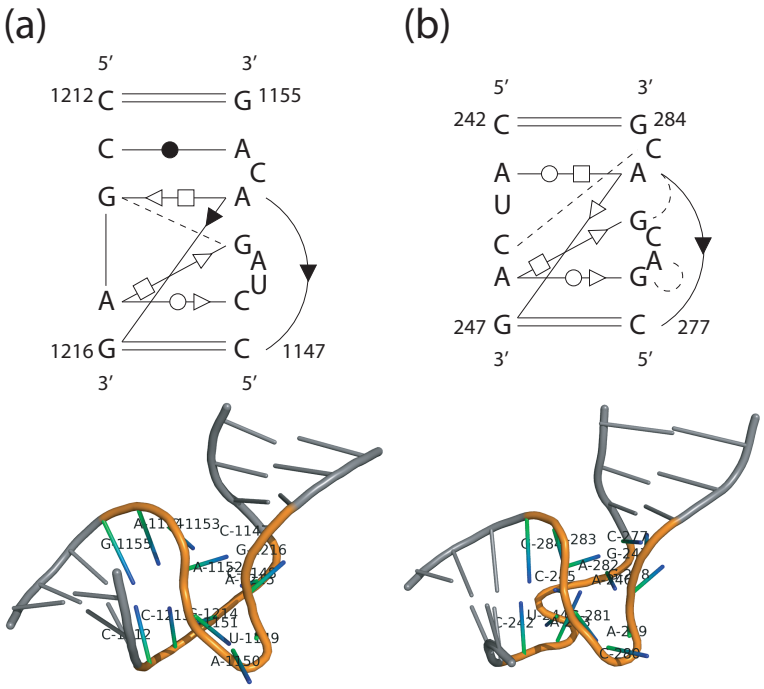

Figure 4.2: The base-pairing patterns and structures of the two kink-turn motif instances clustered in CL7. (a) A known kink-turn instance found in 1S72, chain '0', 1147-1155/1212-1216. (b) The potential novel kink-turn instance found in 1J5E, chain A, 242-247/277-284. The dashed edges in the base-pairing patterns (both in this figure and in the remaining figures of this chapter) correspond to additional base pairs annotated but not included into the consensus structure. The regions that are not part of the motif are colored gray (both in this figure and in the remaining figures of this chapter).

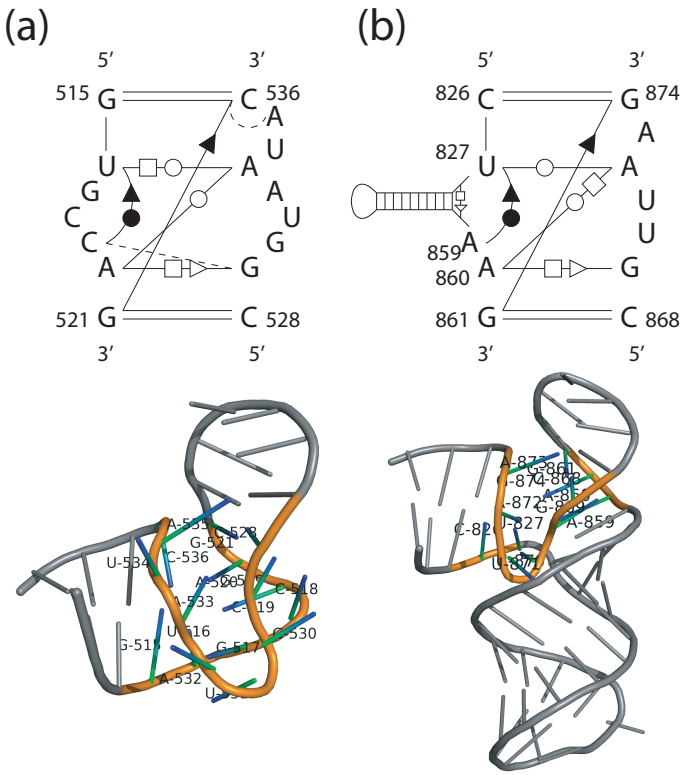

Figure 4.3: The base-pairing patterns and structures of the two kink-turn motif instances clustered in CL6. (a) A novel kink-turn instance found in 1J5E, chain A, 515-521/528-536. (b) A novel kink-turn instance found in 1J5E, chain A, 826-861/868-874.

4.3.1.2 Kink-turn

The kink-turn motif is an asymmetric internal loop characterized by the 'kink' observed in its longer strand which causes a sharp turn between its two supporting helices [83, 84]. It is known to be an important recognition site for interaction with proteins or other RNA elements [75, 135]. We have identified four out of nine known kink-turn instances in 1S72 23S rRNA and the known instance in 1J5E 16S rRNA in cluster CL15 with no false positive prediction (see Table 4.1). Base-pair variations are frequently observed in kink-turn motif instances, making the sensitivity of both base-pairing pattern based clustering methods relatively low. Therefore, some potential novel kink-turn instances can also be found in other clusters besides cluster CL15, as we will describe in details below.

One potential novel kink-turn motif instance is clustered with a known kink-turn motif instance in CL7. The highly conserved bulged nucleotides that correspond to the 'kink' can be found at U1149-A1150 in Figure 4.2 (a) and A279-C280 in Figure 4.2 (b). Interestingly, two nucleotides (U244, C245) are inserted in the novel instance, which induces an 'S' shaped bend at the opposite strand of the 'kink' (see Figure 4.2 (b)). The insertion has altered both base-pairing pattern and geometry of the instance with unknown corresponding biological impact. However, we can still categorize this instance as kink-turn motif based on its base-pairing and geometric similarity with the known kink-turn instance.

Another kink-turn cluster, CL6, contains two potential novel kink-turn instances. The base-pairing patterns and 3D geometries of both instances are very similar to known kink-turn instances. However, both instances contain two pairs of cross-strand base-triples (see Figure 4.3). These base-triples form two 'Z' shaped interactions (G515-C536-G521-C528, U516-

A533-A520-G529 in Figure 4.3 (a) and C826-G874-G861-C868, U827-A872-A860-G869 in Figure 4.3 (b)). Unlike regular kink-turn instances, the two pairs of cross-strand base-triples extrude two bulge regions, one at each strand. In the first instance, G517-C519 are also bulged out in addition to G530-A532 that corresponds to the 'kink', making a much more severe turn at the companion strand comparing to regular kink-turn instances (see Figure 4.3 (a)). More interestingly, in the second instance, an A-form helix of ten canonical base pairs is inserted at this region and interrupts the kink-turn instance (see Figure 4.3 (b)). These two motif instances reveal a potential new form of kink-turn motif where two bulges are extruded. It is also interesting to study the impact of the insertions on the binding activity of kink-turn motif.

4.3.1.3 C-loop

The C-loop motif is an asymmetric internal loop characterized by the base triple induced from the cytosine residue [83]. We clustered two out of three known C-loop motif instances in 1S72 23S rRNA and the only known C-loop motif in 1J5E 16S rRNA in cluster CL24 (see Table 4.1). We missed one known C-loop motif instance in 1S72, chain '0', 958-963/1005-1008 because of two nucleotide insertions, one at each strand (G960 and A1006). Also, four additional base-pairs are annotated in this instance, which indicates unusual properties of this C-loop motif instance.

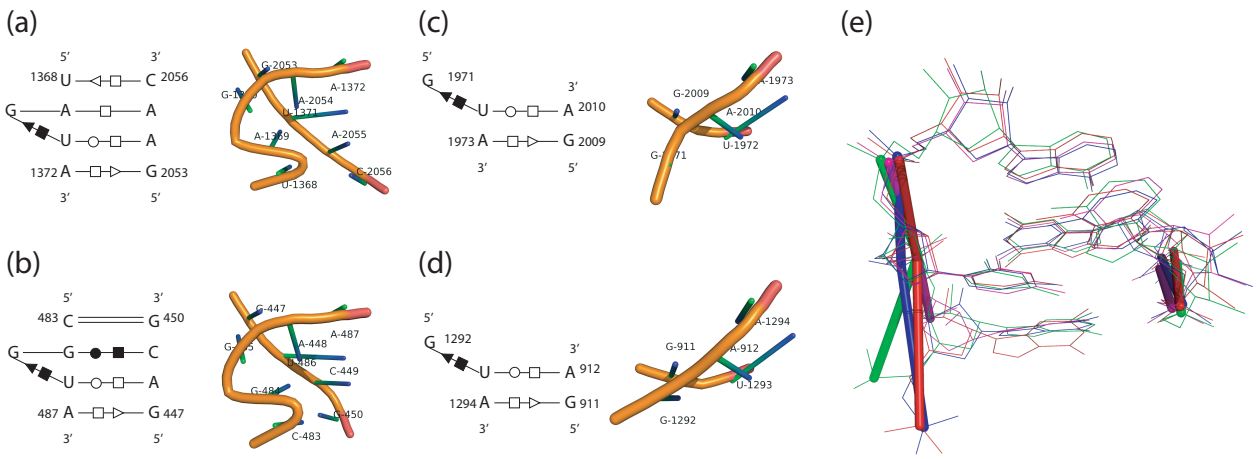

Figure 4.4: The base-pairing patterns, structures and superimposition of the three base pairs formed near the bulged 'G' of four sarcin-ricin motif instances clustered in CL13. (a) A known sarcin-ricin instance found in 1S72, chain '0', 1368-1372/2053-2056. Three novel sarcin-ricin instances: (b) 1J5E, chain A, 483-487/447-450, (c) 1S72, chain '0', 1971-1974/2009-2010 and (d) 1S72, chain '0', 1251-1254/911-912. (e) The superimposition of three base pairs that characterize the sarcin-ricin motif in these four motif instances (red: (a); blue: (b); green: (c); magenta: (d)).

4.3.1.4 Sarcin-ricin

The sarcin-ricin motif (or sometimes referred as the G-bulge motif) is an asymmetric internal loop that is known to be involved in the interaction between the ribosomal RNA and elongation factors [125]. There are ten known sarcin-ricin motif instances in 1S72 (nine in 23S and one in 5S rRNA) and two in 1J5E. We have successfully clustered all twelve known sarcin-ricin instances in cluster CL13, while the `LENCS` method only clustered eight of them (six in 1S72 and two in 1J5E, see Table 4.1). Three potential novel instances are also included in cluster CL13, which are presented in Figure 4.4.

Figure 4.4 (a) shows a well-established sarcin-ricin motif instance in CL13. In its base-pairing pattern, we can observe that the characterized bulged G1370 is interacting with its consecutive nucleotide U1371 using *cis* SE/H pair, followed by two non-canonical base

pairs: *trans* W/H U1371-A2054 and *trans* W/SE A1372-G2053. These three base pairs have been used to characterize the sarcin-ricin motif [28, 33, 116]. Figure 4.4 (b) shows the first potential novel sarcin-ricin instance found in cluster CL13. This potential instance shows base-pair variations in the two pairs before the bulged G (*cis* W/W C483-G450 and *cis* W/H G484-C459) comparing to the known instance. However, it is conserved for the three characteristic base pairs. The 3D geometry of this potential instance also shows high similarity comparing to the known sarcin-ricin motif instance, where an 'S' shape turn can be observed.

The two potential sarcin-ricin motif instances, shown in Figure 4.4(c) and (d), were identified from the junction loop regions instead of internal loop regions (where sarcin-ricin motif instances are usually found). It is worth noting that some known sarcin-ricin motif instances can also be found in the junction loop regions (e.g., the known sarcin-ricin motif instance at 1S72, chain '0', 380-384/405-408). These two potential sarcin-ricin instances are conserved in the three characteristic base pairs but without the other two base pairs. The absence of the other two base pairs makes the two instances smaller than regular sarcin-ricin motif instances and results in large geometric variations (i.e., the 'S' shape turn cannot be observed for these two instances). However, the local geometries associated with the three characteristic base pairs are still highly conserved in these two motif instances (see Figure 4.4 (e)), suggesting potential functional similarity between these two motif instances and regular sarcin-ricin motif instances. Nevertheless, the specific functions of these potential motif instances still need to be experimentally investigated.

4.3.1.5 Reverse Kink-turn

The reverse kink-turn motif is also an asymmetric internal loop that produces a turn between two supporting helices such as kink-turn motif but towards the opposite direction [78]. There are three known reverse kink-turn motif instances in 1S72. We have clustered all three known instances in cluster CL18 with no false positive predictions (see Table 4.1). The `LENCS` method has also clustered these three known reverse kink-turn instances, however, with four unrelated instances. The reason for the false positive predictions is that the `LENCS` method does not consider nucleotide when determining base-pair isostericity. For example, a false prediction made by `LENCS` in 1S72, chain '0', 2307-2310/2298-2300 contains a *trans* H/SE U2308-G2299 base pair. This base pair is matched to the *trans* H/SE A-C or A-G pair in the true reverse kink-turn instances. Although these base pairs have the same orientation and interacting edges, *trans* H/SE U-G pair is not isosteric with *trans* H/SE A-C or A-G pair. In our clustering framework, strict definition of base-pair isostericity is applied to avoid such unexpected false predictions.

Interestingly, two of the known reverse kink-turn instances (1S72, chain '0' 1527-1529/1662-1664 and 1531-1533/1658-1660) appear to be located close to each other, and manual inspection of the region suggests an instance of tandem reverse kink-turn (see Figure 4.5). As there are only three known reverse kink-turn instances in the entire 23S rRNA, the chance of finding a tandem case is extremely low. Therefore, the tandem reverse kink-turn is likely to be required for certain biological functions. On the other hand, we investigated the other known reverse kink-turn instance (1S72, chain '0' 1132-1134/1228-1230) but did not found a tandem counterpart, which implies different functional roles played by single and tandem reverse kink-turn motif instances.

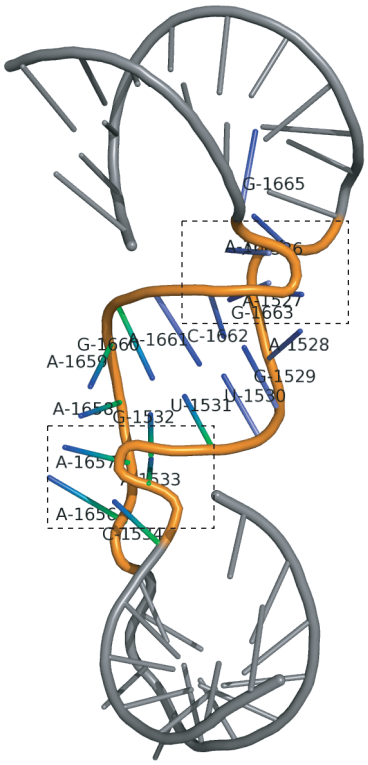

Figure 4.5: The tandem reverse kink-turn motif instance found in 1S72, chain '0', 1515-1540/1645-1670. The two reverse kink-turn instances are colored. The 'kink' regions are indicated by the two boxes.

4.3.1.6 Hook-turn

The hook-turn motif is found at regular A-form helix regions, where one of the nucleotide chain sharply folds back towards the opposite direction [124]. We identified two out of three known hook-turn motif instances in 1S72 23S rRNA with no false prediction (see Table 4.1). The `LENCS` method identified all three known hook-turn instances but include two unrelated motif instances. Figure 4.6 shows the two known hook-turn motif instances clustered in CL17, where conserved base-triples can be observed in both instances (G2267-C2243-A2244 and G2810-G2674-A2675). These two base-triples are both annotated solely by MC-Annotate, which indicates that these base-triples are likely to be real instead of being artifacts of combining `RNAVIEW` and `MC-Annotate` annotations (see Figure 4.6 (c) for

their superimposition). However, RNAVIEW does not predict these base-triples, making the LENCS method (which solely considers RNAVIEW annotations) include the two unrelated motif instances. This base-triple is not predicted by either RNAVIEW or MC-Annotate in the other known hook-turn motif instance, 1S72, chain '0', 1457-1460/1483-1485, hence it was missed by our clustering method.

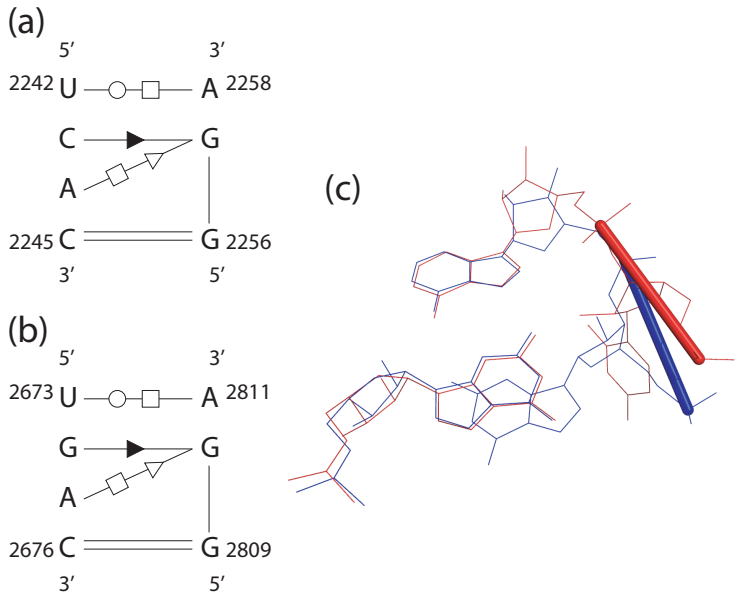

Figure 4.6: The base-pairing patterns and superimposition of the base-triple interactions of the two known hook-turn instances identified in cluster CL17. (a) 1S72, chain '0', 2242-2245/2256-2258. (b) 1S72, chain '0', 2673-2676/2809-2811. (c) The superimposition of the base triples in these two motif instances shown in (a) and (b) (red: (a); blue: (b)).

4.3.1.7 E-loop

The E-loop motif is a symmetric internal loop that contains the following base pairs: a *trans* H/SE base pair, a *trans* W/H or *trans* SE/H base pair, and a *cis* bifurcated or *trans* SE/H base pair as summarized by Leontis *et al.* [79]. We notice that there are confusions

in distinguishing E-loop and sarcin-ricin motifs since they share similar base-pairing pattern (i.e., the three base pairs that define the E-loop motif). Another reason can be that bacterial 5S rRNA contains an E-loop motif while the corresponding region in *H. marismortui* appears to be sarcin-ricin motif. In this chapter, we consider an instance without the bulged G (and the base pair formed with its consecutive nucleotide) to be E-loop motif and otherwise sarcin-ricin motif.

Using this criterion, there are two E-loop motif instances in 1S72 23S rRNA and two in 1J5E 16S rRNA. We clustered all four instances in cluster CL19, with two false positive predictions that appear to be tandem-sheared motif instances (see Table 4.1). The `LENCS` method has also successfully identified all four instances, but include three other unrelated motif instances, where one of them appears to be a sarcin-ricin motif instance (1J5E, chain A, 446-450/483-488) and the other two are kink-turn motif instances (1S72, chain '0', 241-244/267-270 and 1J5E, chain A, 683-687/703-707). The inclusion of false positive prediction by both methods even under strict P-value cutoff and graph isomorphism indicates that the universal cutoff which can optimize the overall clustering performance may not be strict enough for E-loop motif.

4.3.1.8 Tandem-sheared

The tandem-sheared motif consists of two consecutive sheared base pairs and is frequently observed in regular helix regions [31]. There are four known tandem sheared motif instances in 1S72 23S rRNA and two in 1J5E 16S rRNA. The `LENCS` method has identified all six known tandem sheared motif instances but included two kink-turn motif instances. We identified

two out of six known instances but with no false positive prediction in cluster CL23 (see Table 4.1). The tandem-sheared instances identified by us are strictly closed by canonical base pairs at both ends, while the other missed instances are surrounded by additional non-canonical base pairs. We have also identified a potential novel tandem-sheared motif instance (also strictly closed by canonical base pairs) in cluster CL23. The base-pairing patterns and structures of a known tandem-sheared motif instance and the potential novel instance are shown in Figure 4.7. The colored backbone regions correspond to the tandem-sheared base pairs, and slight inward turns can be observed in these regions from both instances.

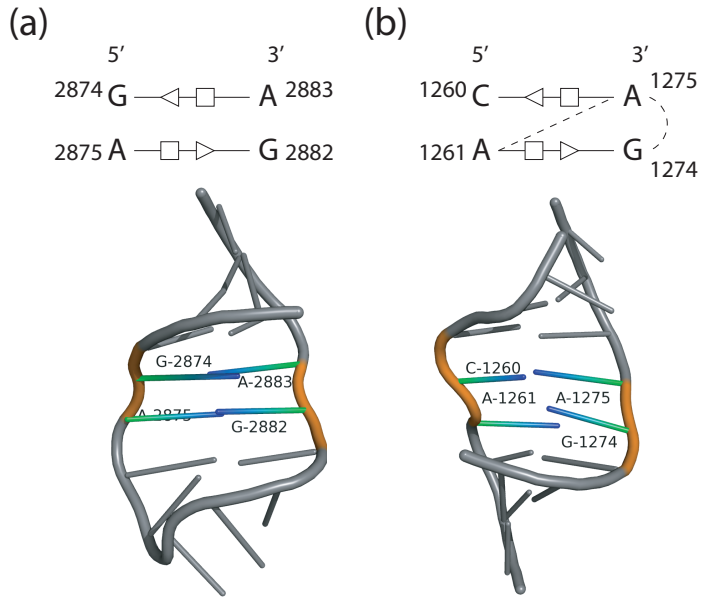

Figure 4.7: The base-pairing patterns and structures of two tandem-sheared instances identified in cluster CL23. (a) A known tandem-sheared instance found in 1S72, chain '0', 2874-2875/2882-2883. (b) The novel tandem-sheared instance found in 1J5E, chain A, 1260-1261/1274-1275.

4.3.2.1 The 'Rope Sling' Motif

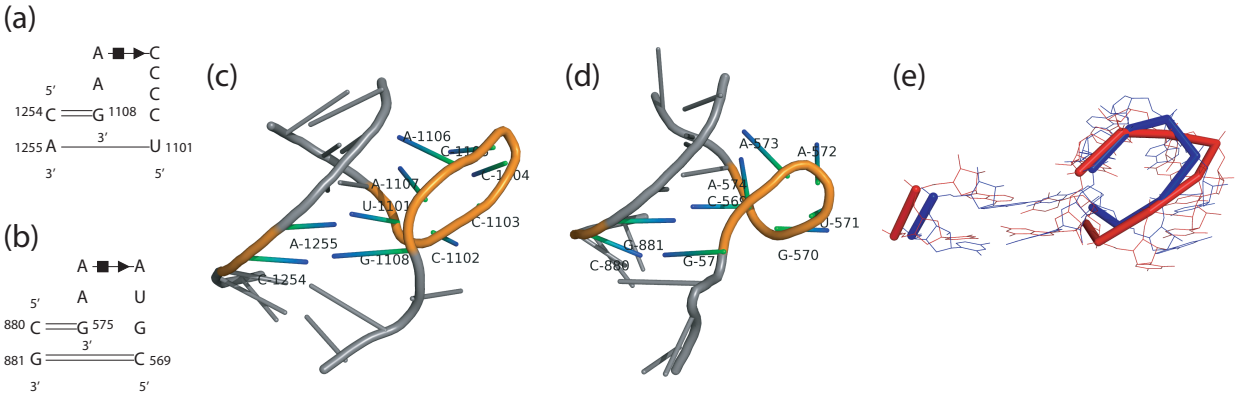

Figure 4.8: Potential novel motif family that resembles the rope sling. (a) and (b) The base pairing patterns of structural components found in 1S72, chain '0', 1254-1255/1101-1108 and 1J5E, chain A, 880-881/569-575, respectively. (c) and (d) Local structures around motif instances shown in (a) and (b), respectively. (e) The superimposition between these two motif instances (red: (a); blue: (b)).

We have discovered a highly asymmetric bulge loop motif family that resembles the rope sling. The corresponding motif cluster (CL1), which has the lowest average P-value, consists of two motif instances: one from 1S72 23S rRNA and the other from 1J5E 16S rRNA. The base-pairing patterns and structures of these two motif instances are shown in Figure 4.8. Both motif instances consist of two highly asymmetric strands, where the longer ones have seven to eight nucleotides while the shorter ones have only two nucleotides. The first and last nucleotides of the longer strands form canonical base pairs with the two nucleotides in the shorter strands, leaving the other nucleotides in the longer strands bulged out from the main helix and resulting in a loop similar to rope sling (see Figure 4.8). Two consecutive nucleotides (C1105-A1106 and A572-A573) within the bulged chains form *cis* SE/H non-canonical interactions.

Several evidences indicate that the functionalities of the rope sling motif are carried out by its longer strand. First, a non-canonical *cis* SE/H base pair can observed in the longer stand of both motif instances (C1105-A1106 and A572-A573). The nucleotide mutation (C1105 to A572) in these two base pairs is compensated by their isostericity. Second, two nucleotides in the longer strand of both motif instances also participate in non-nested canonical interactions (C1102-G1241 and C1103-G1240 in the first motif instance and G570-C866 and U571-A865 in the second motif instance). These conserved non-nested interactions also indicate the structural importance of these regions. Finally, high geometric similarity of the longer strands can also be observed from the superimposition between these two motif instances (see Figure 4.8 (e)). Therefore, we conjecture that the longer strands may determine the functionalities of the rope sling motif. Using `RNAMotifScan`, we also identified this motif from both 16S and 23S rRNA in *H. marismortui*, *T. thermophilus* and *E. coli*. The recurrence of this motif further indicates its structural or functional importance for ribosomal RNAs.

4.3.2.2 Motif that Increases the Twist at the Helical Region

Two internal loop motif instances, both closed by an A-U and a C-G canonical base pairs, were clustered in CL2. The conserved non-canonical base pairs between the two motif instances are the *cis* W/SE pairs formed at C1383-A935 and C36-A47 (see Figure 4.9 (a) and (b)). The sequences are highly conserved at the left strand, where only one nucleotide substitution at the unpaired region can be observed (U1381 mutated to A34). On the other hand, a two-nucleotide deletion (between G933 and C934) is found at the right strand in the first motif instance. The nucleotide deletion alters the interaction between G933 and the

left strand, violating the *trans* SE/H U33-G43 pair that can be observed in the second motif instance. The *trans* SE/W G43-C46 pair cannot be formed either, since G933 and C934 are too close to each other.

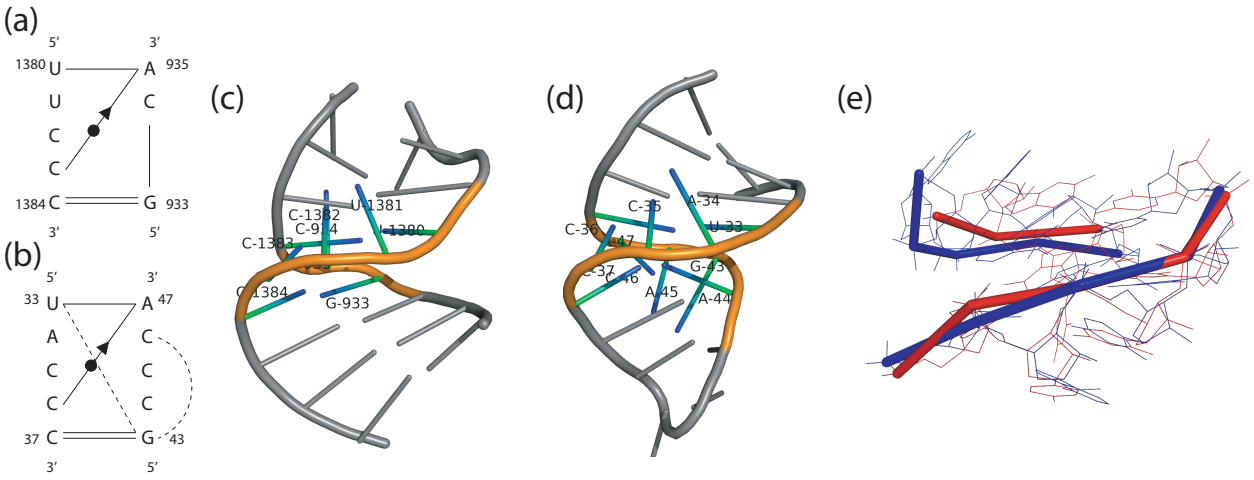

Figure 4.9: Potential novel motif family that increases the twists at the helical region. (a) and (b) The base pairing patterns of structural components found in 1J5E, chain A, 933-935/1380-1384 and 1S72, chain '9', 33-37/43-47, respectively. (c) and (d) Local structures around the motif instances shown in (a) and (b), respectively. (e) The superimposition between these two motif instances (red: (a); blue: (b)).

Superimposition of the two motif instances clearly reveals high structural similarity between the left strands (see Figure 4.9 (e)), where two nucleotides (U1380, C1383 in the first motif and U33, C36 in the second motif) participate in the conserved base triple. The base triple indicates that the two nucleotides in the left strand are spatially close to each other. As a result, the left strand is likely to exhibit an unusual backbone conformation, such as a tight bend that can bring these two nucleotides together. Visualization of the local structures around the motif instances clearly shows increased twists at the corresponding regions (see Figure 4.9 (c) and (d)). The two strands of the motif instances are nearly parallel to each other and form planes that are perpendicular to the main helical axes, suggesting rather acute twists induced by this motif.

The functionalities of this motif family remain unclear without further experimental investigations. However, some evidences suggest potential binding activity of the motif. The twists deepen the groove where the potentially bound biomolecules can reside. At the same time, they also narrow down the helix, which can tightly clip the biomolecules that would have been embedded. Moreover, both motif instances are located at the surfaces of the ribosomal RNAs, which further suggests binding potentials.

4.3.2.3 New Subfamily of Hexaloop Motif

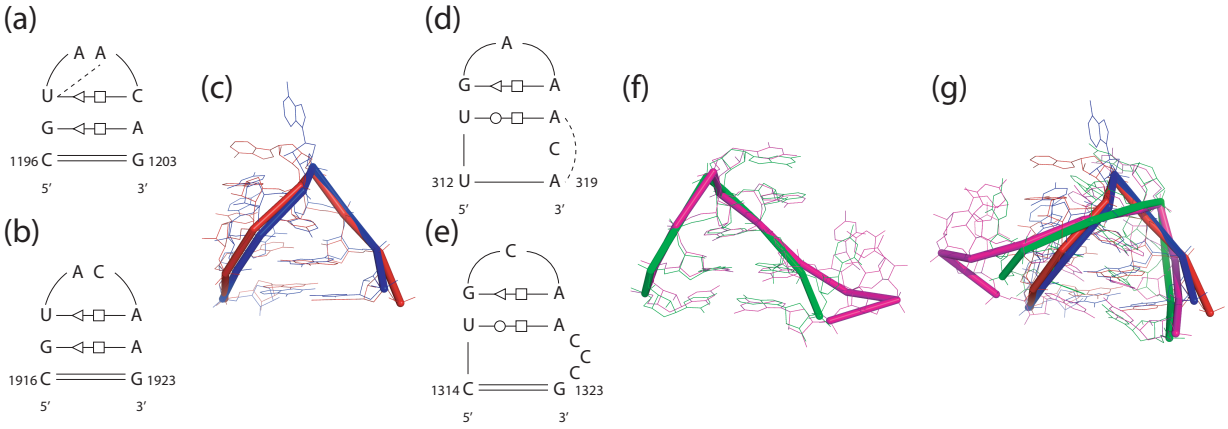

Figure 4.10: A novel type of hexaloop motif subfamily detected by our clustering method. (a) and (b) Base pairing pattern of the two hexaloop motif instances identified in CH6: 1S72, chain '0', 1196-1203 and 1S72, chain '0', 1916-1923, respectively. (c) Superimposition between the motif instances shown in (a) and (b) (red: (a), blue: (b)). (d) and (e) Base pairing pattern of the two hexaloop motif instances identified in CH8: 1S72, chain '0', 312-319 and 1J5E, chain A, 1314-1323, respectively. (f) Superimposition between the motif instances shown in (d) and (e) (green: (d), magenta: (e)). (g) Superimposition of the four hexaloop motif instances.

We have identified two clusters that correspond to the hexaloop motif (CH6 and CH8). Cluster CH6 contains two hexaloop instances from 1S72 23S rRNA, both of which share the common base-pairing pattern that two *trans* SE/H base pairs stack together. The nucleotide

U1198 in the first motif instance is also annotated to be pairing with A1200, while this base pair is absent in the second motif instance (see Figure 4.10 (a) and (b)). This base-pairing variation results in the geometric difference that A1199 in the first motif and A1919 in the second motif are extruded towards different directions (see Figure 4.10 (c)). Other than this difference, the backbones and the rest of nucleotides can be well superimposed, indicating true motif recurrence.

Cluster CH8 contains one motif instance from 1J5E 16S rRNA and one from 1S72 23S rRNA, both of which share the base-pairing pattern that *trans* SE/H G-A pair (G314-A316 and G1316-A1318) stacks on *trans* W/H U-A pair (U313-A317 and U1315-A1319). The second motif instance contains two inserted cytosine residues between C1320 and G1323, which likely destruct the *trans* SE/H A-A pair (A317-A319) that can be observed in the first motif instance (see Figure 4.10 (d) and (e)). However, superimposition between the two motif instances reveals that the nucleotide insertions are well accommodated (see Figure 4.10 (f)). Therefore, although the insertion increases the hairpin loop length and the motif instance cannot be literally called 'hexaloop', we consider this instance to be true hexaloop motif due to its conservation in both base pairing pattern and 3D geometry.

The hexaloop motif family has been previously registered in the SCOR database [127], which defines only one hexaloop cluster in contrast to two subfamilies of hexaloop motif as suggested by our clustering results. SCOR identified all hexaloop motif instances found by us except the one with eight nucleotides. We consider that the two clusters of hexaloop motif have different sequence signatures and more importantly, different base pairing patterns (the *trans* SE/H G-A pair in CH6 comparing to the *trans* W/H U-A pair in CH8), therefore, should be classified into two different subfamilies. Indeed, superimposition of the four hexaloop motif instances clearly reveals two subfamilies of the motif that are consistent with our clustering

predictions (see Figure 4.10 (g)). In this case, motif characterization should involve thorough consideration of both base pairing pattern and geometry, and classification of motif solely based on their sizes should be revised to incorporate such information.

4.4 Discussion

In this chapter, we studied RNA structural motifs in ribosomal RNAs using a *de novo* clustering method based on base-pairing patterns. The similarities between RNA structural motifs were evaluated by `RNAMotifScan`, which is a secondary structural alignment tool that considers non-canonical base pairs and their isostericity. We have significantly improved the existing clustering performance (see Table 4.1) achieved by the `LENCS` method through addressing the three issues raised in the Introduction section. The clustering framework can benefit future RNA structural motif analysis.

The newly identified motif instances were not discovered by previous base-pairing pattern based search methods since they contain base-pair variations. The base pairs that are conserved in these instances can be critical in forming the motifs, and further studies should be conducted to elucidate their roles in maintaining proper functionalities of the motifs. On the other hand, the base-pair variations should also be investigated to study functional evolution. Finally, more comprehensive consensus models can be built to facilitate future model-based searches by combining both information. The discoveries of novel motif families are also exciting. These new motifs may lead to the discovery of unknown structure-function relationships and define new building blocks for the RNA architecture, significantly improving our understanding of the RNA structural motifs.

CHAPTER 5: GENOME-WIDE STRUCTURAL CLUSTERING OF RNA SECONDARY STRUCTURES

In Chapter 4, we present an RNA structural motif clustering pipeline called `RNAMSC`. The new pipeline is highly accurate and robust, and has significantly improved over the existing hierarchical clustering methods. In this case, it is desirable that we can take advantage of this clustering pipeline, and conduct a genome-wide clustering analysis of the ncRNA elements. To achieve this goal, we replace `RNAMotifScan` with an RNA secondary structure alignment tool. We apply this new RNA secondary structure clustering pipeline to analyze the post-transcriptional control elements from fly 3'-UTR.

Post-transcriptional control elements regulate the expression of genes after the transcription of the genes, and such mechanism is considered to offer an additional layer of regulation to fine-tune the gene expressions in the biological system. Many of the post-transcriptional controls elements locate at the 3'-UTR of the mRNA, and recruit corresponding protein factors through their sequence motif or secondary structures (i.e. Nanos [30] and Histone [143]). In this case, clustering the secondary structures from the 3'-UTRs has a great potential to find co-regulated or co-expressed gene clusters whose expressions are controlled by their 3'-UTR elements. These gene clusters will provide invaluable information for us to further understand the cellular functions of these genes. In this chapter, we present the work flow of the new RNA secondary structure clustering pipeline, its application on the *Drosophila melanogaster* 3'-UTR elements, and detailed functional analysis of the resulting gene clusters.

5.1 Limitation of Clustering Analysis of Post-transcriptional Control Elements

Post-transcriptional control is the regulation at the protein level through the existing mRNAs by modifying their stability, translation efficiency and subcellular locations. Many of the regulations are found to be triggered by RNA-protein or RNA-RNA interaction, which usually occur in the 3' untranslated regions (3'-UTRs) of the mRNA [19, 90, 92]. In eukaryotes, the sequence or structural elements in the 3'-UTR of some genes under regulation serve as 'zip-code', determining the fate of their corresponding mRNAs through interaction with transportation or entrapment proteins, or signalling molecules [67]. For instance, the NOS translational control element, *cis*-regulates the expression of Nanos protein through binding with the Smaug protein, which in turn determines the proper morphogenesis of the *Drosophila* embryo [30]. The sequence and structure features of the translational control elements, which determine the fate of the corresponding mRNA through specific recognition of partner RNAs or proteins, are thus critical in understanding the expression pattern and functionalities of the corresponding genes. For example, the conserved histone 3'-UTR stem loop [41] suggests that the histone genes are co-regulated and co-expressed, which implies their potential collaborations in nucleosome packing. In this work, we are particularly interested in identifying common non-coding RNA (ncRNA) elements from the 3'-UTRs, and using such information to infer the corresponding genes' co-regulation or co-expression patterns.

Recently, Rabani *et al.* identified a number of 3'-UTR ncRNA elements from *Drosophila melanogaster* genome [104] using improved stochastic context-free grammar (SCFG) [46]. They detected several structured ncRNA elements from experimentally verified co-localized genes [77]. Because experimental determination of the gene expression patterns (both tem-

poral and spatial) can be expensive, we propose to computationally infer the genes' potential co-regulation pattern through structural clustering before conducting real experiments. Currently, there exist many computational tools for *de novo* identification of ncRNA elements from multiple alignments, such as `RNAz` [137], `Evofold` [103], `MSARI` [29], `QRNA` [108] and `ddbRNA` [36] *etc.* We will first use these ncRNA identification tools to reveal the candidate structured regions in the 3'-UTRs, and then use pairwise structural alignment tools such as `LocARNA` [142], which implements the alignment of pairing-probability matrices [65, 93], to compute the structural similarities between the candidate ncRNA elements. Finally, we will cluster the candidate ncRNA elements from 3'-UTRs based on their sequence and structural similarity, and predict the co-expression patterns of the genes whose 3'-UTR RNA elements are clustered.

However, the clustering performance, despite the fact that high-quality pairwise alignments can be generated by many state-of-the-art alignment tools (i.e. `LocARNA` achieves over 80% sum-of-pair score even for RNA sequences with <40% identity), remain relatively low (the F-measure for clustering pipeline based on `LocARNA` is only 64.8%). We conjecture that the performance bottleneck may exist in the clustering algorithm itself, rather than the structural alignment quality. Specifically, we notice that the local structural alignment scores, which appear to be length-dependent, are fed into the hierarchical clustering algorithm without normalization. The consequence is that hierarchical clustering may merge longer ncRNA candidates with higher priority, rather than those with higher structural similarity. Such problems also exist in many of the existing clustering pipelines, such as [69, 107, 129, 131]. To normalize the structural alignment scores, we simulate the RNA structure alignment score distribution through a number of randomly generated alignment scores. We then compute statistically meaningful P-values for the structural similarity scores. We also take advantage

of the normalized measures, and devise a more efficient and robust CLique finding CLustering algorithm (CLCL), to replace the traditional hierarchical clustering. In addition, CLCL is also capable of outputting disjoint clusters without further human interaction, which is a highly desirable feature when analyzing a large data set.

We have conducted benchmark experiments against the LocARNA clustering pipeline on Rfam [57] to demonstrate the performance gains made by our proposed clustering method improvement. We chose the same data set (see Materials and methods section) and structural alignment tool (LocARNA) for the comparison. We have seen that by incorporating the clique clustering method, we are able to increase the F-measure, a comprehensive measurement for recall and precision, from 64.8% to 74.9%. A more detailed analysis suggests that the score normalization is responsible for $\sim 70\%$ of the performance gain, and the application of CLCL is responsible for $\sim 30\%$ of the performance gain. Note that in order to reach the LocARNA clustering performance, the correct Rfam classification is required to parse the hierarchical tree and determine the optimal cutting level with the specified recall rate. Such information is not usually available, and the optimal cutting level for the benchmark data set is not necessarily optimal for the data set of interest. On the other hand, our results can be achieved completely automatically and require no additional information. As a result, we have provided a novel clustering pipeline which is more efficient, automatic, and accurate.

We then have applied our clique clustering method to the 3'-UTR of *D. melanogaster* genes and have found 184 3'-UTR ncRNA families, among which 91.3% are predicted to contain a structural element by RNAz. It implies that most clusters identified in this study contain RNA elements with conserved sequences and structures, which further implies that they can possibly be co-regulated. The histone stem-loop are rediscovered among these clusters with high accuracy, in addition to many other gene clusters whose cooperations under certain

physiological processes are suggested by existing studies. In addition, we also present two other gene clusters, where one cluster contains genes that are highly expressed in male *Drosophila*, and the other contains genes that are essential for septate junction function in *Drosophila*.

5.2 Methods

5.2.1 Generating Random RNA Structural Alignment Scores

We propose that the valid random ncRNA structures should have the following two properties: (1) low free energy such that they can be considered to be stable under natural conditions, and (2) the same length to rule out the length bias. Therefore, given the ncRNA sequence of interest, we generate the random RNA sequences that preserve the original dinucleotide frequency and length using the Altschul-Erickson algorithm [4]. Then, we use `RNAfold` [66] to compute the base-pairing probabilities of the random ncRNA sequences. Finally, we aligned pairing probability matrices of the random sequences with the probability matrix of the sequence of interest using `LocARNA`. We consider the resulting alignment scores as the background score distribution associated with the sequence of interest.

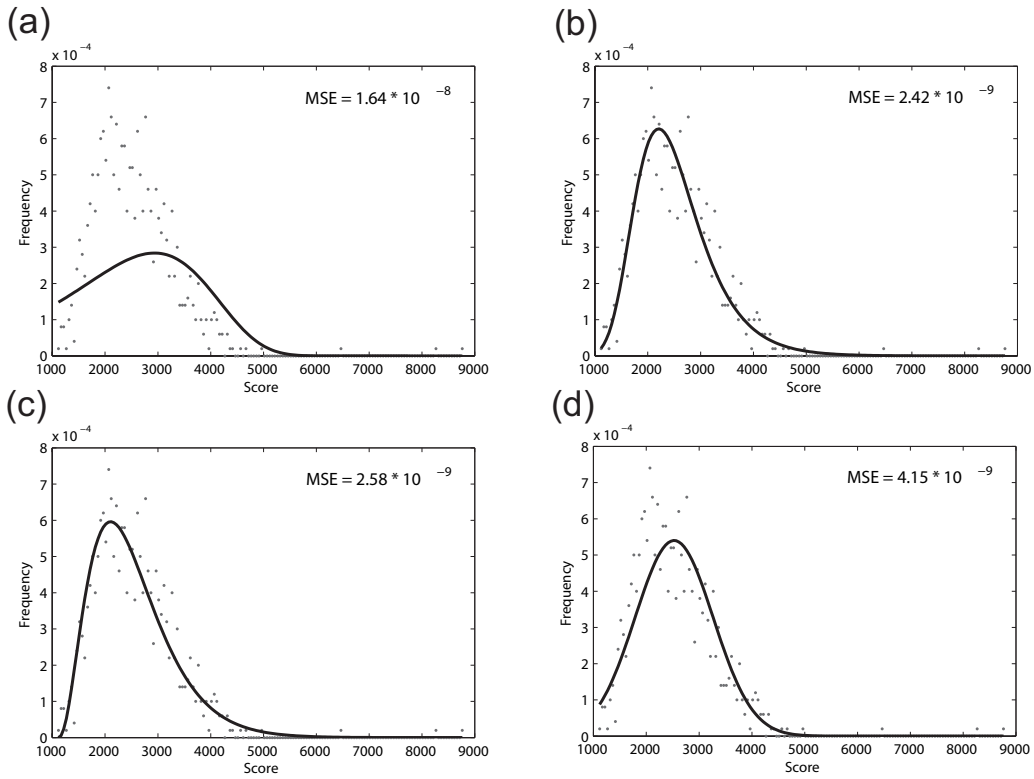

Figure 5.1: Four distributions that have been used to model the RNA structure alignment scores. (a) Gumbel's distribution. (b) general extreme value distribution. (c) Gamma distribution. (d) normal distribution. The mean square error (MSE) is used to measure the goodness of fit. The general extreme value distribution can optimally model the local structural alignment scores.

5.2.2 Optimal Parameters Fitting

We intend to find a distribution that can be used to model the simulated background alignment scores. Note that the local sequence alignment scores have been shown to follow the extreme value distribution [70] while the behavior of local structural alignment score has not yet been studied. To investigate the local structural alignment score distribution, we tested two forms of extreme value distributions. The first one is the widely used two-parameter Gumbel's distribution, and the second one is the three-parameter general extreme value distribution (using MATLAB built-in functions `evfit` and `gevfit`). We also fit the observed alignment score frequency with Gamma distribution and normal distribution (using MATLAB built-in functions `gamfit` and `normfit`), as they have also been previously used to model sequence alignment scores [100]. The fitting results of these four distributions with background alignment scores associated with the Rfam 5S rRNA consensus structure, are shown in Figure 5.1.

The goodness of fit is calculated using the mean square error (MSE) between the sampled alignment score frequencies and the theoretical frequencies under certain distribution assumptions. The experiment results suggest that Gumbel's distribution may not be a model for the local sequence alignment score distribution. Therefore, the more sophisticated three-parameter general extreme value distribution is used for all successive analysis.

5.2.3 Extracting ncRNA Clusters

After curve fitting, we can estimate the statistical significance of the pairwise alignment scores through the computation of their P-values. We denote the alignment score distribution associated with the ncRNA element i as $\mathcal{D}_i$. Given the two-dimensional matrix S, where $S_{i,j}$ is the pairwise structural alignment score between ncRNA element i and j, denote $P(S_{i,j}|\mathcal{D}_i)$ as the P-value of the alignment score $S_{i,j}$ when assuming $\mathcal{D}_i$ as background. Let P_c be an empirical P-value cutoff, we can convert S into a boolean matrix I, where $I_{i,j}$ indicates whether the ncRNA elements i and j are significantly structurally similar to each other:

$$I_{i,j} = \begin{cases} 1 \text{ if } \max(P(S_{i,j}|\mathcal{D}_i), P(S_{i,j}|\mathcal{D}_j)) \leq P_c, \\ 0 \text{ otherwise.} \end{cases} \tag{5.1}$$

Using this conversion, we are able to remove most of the insignificant edges between candidate structures and speedup the successive clustering analysis. The traditional hierarchical clustering generates a hierarchical tree and requires human intervention to output disjoint clusters. Since the number of candidate RNA elements in genome-wide analysis can be large, it is desirable to devise an algorithm that can automatically output disjoint clusters without human intervention. We formulate the cluster extraction problem into a clique-finding problem. Inspired by Bron-Kerbosch's algorithm [23] and Cluster Affinity Search Technique (`CAST`) algorithm [16], we devised a heuristic algorithm named CLique finding for CLustering (`CLCL`) to solve this problem. The pseudo-code for each stage of the `CLCL` algorithm, which finds the potential maximum clique in a given graph, is outlined in Figure 5.2.

The major idea of the algorithm is the following. We keep a set (the set $\mathcal{C}$ in Figure 5.2) which stores vertices that form a clique (i.e each vertex in the set is connected to all other vertices in the set). As the algorithm proceeds, we add a new vertex (v_i) to $\mathcal{C}$ at each phase. The new vertex has to connect to all vertices in $\mathcal{C}$. To ensure this property, we associate each vertex with its clique connectivity ($cc(v_i)$ in Figure 5.2), which depicts the number of edges between v_i and the vertices in $\mathcal{C}$. If v_i connects to all vertices in $\mathcal{C}$, it will be a valid candidate for expanding $\mathcal{C}$. Since we try to identify a clique that is as large as possible, we will select the candidate vertex that has the largest degree, which implies higher potential of connecting to other vertices that have not yet been added. The algorithm will terminate when no candidate vertex is found.

To analyze the time efficiency of this algorithm, denote the number of vertices in the graph as $|V|$, the edges in the graph as $|E|$, the size of the maximum clique as z. We claim that the algorithm outlined in Figure 5.2 can be finished in $O(z|E|)$ time. To see the time complexity, we can divide the algorithm into phases, with each phase corresponding to an execution of the 'while' loop. Each phase contains two 'for' loops, and both 'for' loops are indexed by existing edges in the graph. Therefore the running time for each phase is bounded by $O(|E|)$. Since each phase includes exactly one vertex into the clique, the total number of phases is clearly $O(z)$. As a result, the time complexity of the algorithm shown in Figure 5.2 is $O(z|E|)$.

After analyzing the time complexity for extracting one clique from a given graph, we can extend the analysis to the algorithm's application in extracting all cliques from a given graph. As soon as a clique has been identified, the corresponding vertices will be removed from the original graph, and the same algorithm will be applied to the remaining graph to identify the next clique. Let the size of the ith clique be z_i, and the time required for extracting the

ith clique is T_i, the total time T that is required for extracting all cliques can be written as:

$$T = \sum T_i = \sum O(z_i|E|) = O(\sum z_i|E|) = O(|V||E|). \tag{5.2}$$

Since most of the biological graphs are scale-free [12], we can expect that $O(|E|) = O(|V|)$, and `CLCL` will be finished in quadratic time. The `CLCL` algorithm thus outperforms the traditional hierarchical clustering algorithm with respect to both the running time and the capability of automatically generating disjoint clusters.

The algorithm will output disjoint cliques in the graph. However, the complete connection restriction of clique definition may be too stringent, such that in some cases it separates an RNA family into many subfamilies. To compensate for this drawback, we merged the output cliques which have high connectivity. Similar to clustering coefficient, the connectivity $k_{U,V}$ between cliques U and V can be written as:

$$k_{U,V} = \frac{\sum_{i,j} IsConnect(v_i^U, v_j^V)}{|U| * |V|}, \tag{5.3}$$

where v_i^U is ith vertex in clique U, and $|U|$ is the size of the clique U. *IsConnect* is a boolean function defined as the following:

$$IsConnect(v_i, v_j) = \begin{cases} 1 \text{ if vertex } v_i \text{ connects with vertex } v_j, \\ 0 \text{ otherwise.} \end{cases} \tag{5.4}$$

$k_{U,V}$ is empirically set to 0.4 for all experiments.

Algorithm 1 Finding the largest clique in $G(V, E)$

Compute the degree for each vertex (in $O(|E|)$ time)
$\mathcal{C} \leftarrow \emptyset$
for each vertex $v \in V$ **do**
  $cc(v) \leftarrow 0$ {clique connectivity, the number of vertices in the clique $\mathcal{C}$ that connect to v}
end for
$i \leftarrow$ the vertex that has the maximum degree (in $O(|V|)$ time)
while i is defined **do**
  $\mathcal{C} \leftarrow \mathcal{C} \cup i$
  $maxV \leftarrow undefined$, $maxDegree \leftarrow 0$
  for each edge e from i **do**
    $j \leftarrow$ the other vertex connected to i through edge e
    $cc(j) \leftarrow cc(j) + 1$
    Delete edge e, $degree(j) \leftarrow degree(j) - 1$
    if $degree(j) > maxDegree$ **and** $cc(j) == |\mathcal{C}|$ **then**
      $maxDegree \leftarrow degree(j)$, $maxV \leftarrow j$
    end if
  end for
  $i \leftarrow maxV$
  for each edge $e \in E$ **do**
    $k, l \leftarrow$ the two vertices connected through edge e
    if $cc(k) == |\mathcal{C}|$ **and** $cc(l) < |\mathcal{C}|$ **then**
      Delete edge e, $degree(k) = degree(k) - 1$
    else if $cc(k) < |\mathcal{C}|$ **and** $cc(l) == |\mathcal{C}|$ **then**
      Delete edge e, $degree(l) = degree(l) - 1$
    else if $cc(k) < |\mathcal{C}|$ **and** $cc(l) < |\mathcal{C}|$ **then**
      Delete edge e
    end if
  end for
end while
Output $\mathcal{C}$ as a clique

Figure 5.2: An overview of the CLCL algorithm. At each stage, the heuristic algorithm tries to identify the clique with largest size from the given unit-weighted, undirected graph. Notation: (v_i, v_j) denotes an edge connecting the vertices v_i and v_j; $adj(v)$ denotes the set of vertices that are adjacent to vertex v.

5.2.4 Rfam Data Set

We generated two data sets to investigate the performance of the clique clustering method. The first data set is exactly the same as the one used in the `LocARNA` clustering benchmark. It contains 3,901 individual RNA structures from 499 families in the Rfam [57] seed alignment (with sequences longer than 400 bp and having >80% sequence identities filtered out). This data set is referred to as '*Rfam*' data set in the following sections. The second data set contains 263 individual RNA structures from seven families in Rfam seed alignment whose average sequence identities are <50%. These families include 6S, RNase_MRP, RNaseP_nuc, SECIS, T-box, tmRNA and yybp-ykoy. We compiled this data set to confirm that the clique clustering pipeline will also work well on ncRNA families with low sequence identity. This data set is referred to as '*Rfam_LowID*' data set in the following sections.

5.2.5 D. Melanogaster 3'-UTR Candidate ncRNA Elements

The *D. melanogaster* genome and multiple alignments were downloaded from UCSC genome browser (version dm3). The gene annotation was taken from FlyBase (*D. melanogaster* version 5.12) [43]. The multiple alignments of the 3'-UTR of each gene were cut and fed into standard `RNAz` [137] analysis pipeline (using 120 bp window size and 40 bp step size). Sequences with `RNAz` RNA class probability value greater then 0.5 were taken as potential candidate regions. In total, 3,657 candidate regions were collected. Their base-pairing probability matrices were computed using `RNAfold` [66].

5.3 Results

5.3.1 Benchmarking using Rfam Database

Here we compare the clustering performance of our clique clustering method, to the traditional hierarchical clustering method (as used in the `LocARNA` pipeline). The F-measure, which is the harmonic mean of recall and precision, are compared between the two clustering experiments. Figure 5.3 (a) shows the F-measure for `LocARNA` hierarchical clustering on *Rfam* data set (red) and the clique clustering on *Rfam* (green). It is observed that the clique clustering pipeline outperforms the hierarchical clustering by over 10% of F-measure (74.9% compared to 64.8%). The peak performance of the clique clustering method is observed around P-value cutoff 0.01. This P-value cutoff is then used in the real-world application of this clustering pipeline in analyzing *Drosophila* 3'-UTR. The benchmark results confirm our conjecture that improving the clustering performance itself is as important as developing accurate pairwise structural alignment methods.

Table 5.1: Detailed clustering results on *Rfam_LowID* data set

Rfam ID	Family	Ave. Identity	Ave. Length	Count	# Clusters	Sensitivity	Specificity
RF00013	6S	45%	180.10	5	1	100%	71.4%
RF00030	RNase_MRP	42%	321.70	18	1	72.2%	100%
RF00009	RNaseP_nuc	45%	312.40	38	2	84.2%	100%
RF00031	SECIS_1	45%	64.50	44	2	95.5%	100%
RF00230	T-box	49%	225.70	40	1	99.8%	97.5%
RF00023	tmRNA	48%	356.60	61	2	90.2%	100%
RF00080	yybp-ykoy	49%	121.80	57	1	91.2%	94.5%

Ave. Identity: average sequence identity of the ncRNA family. Count: total number of individual ncRNAs in the family that have been included in the benchmark experiment. Ave. Length: average sequence length of the ncRNA family. # Clusters: number of major clusters for the ncRNA family. Sensitivity: number of clustered ncRNAs over total size of the family. Specificity: number of ncRNAs of the same family over total size of the cluster.

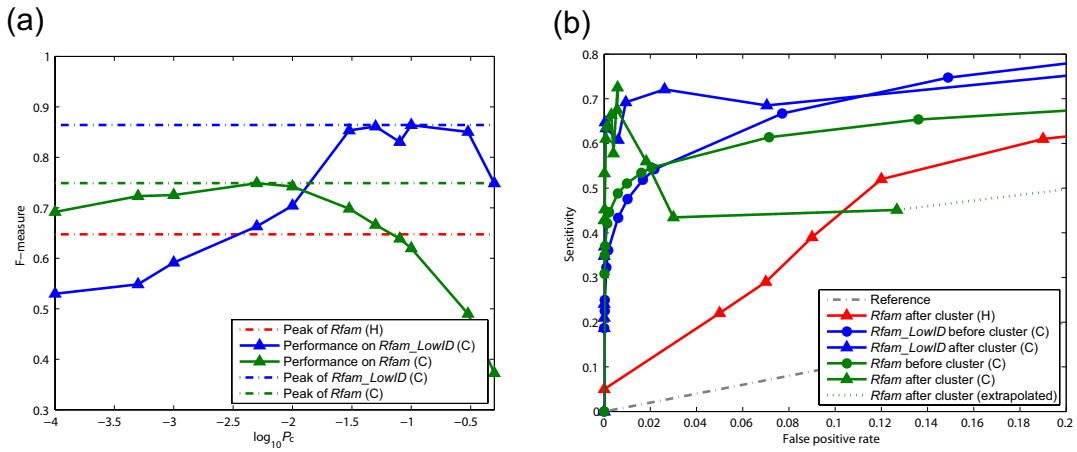

Figure 5.3: Comparison of the clustering performance between `CLCL` and hierarchical clustering. Red series: hierarchical clustering with *Rfam* data set by Will *et al.* [142]. Green series: clique clustering pipeline with *Rfam* data set. Blue series: clique clustering pipeline with *Rfam_LowID* data set. (a) F-measure of the clustering performance on different data sets. The peak performances of the three series are 64.8%, 74.9% and 86.4%, respectively (denoted by broken lines). Note that the cutoff used by Will *et al.* [142] is recall rate, for which the corresponding P-value cutoff is difficult to estimate. Therefore, only the peak performance is presented. (b) ROC curves of clique and hierarchical clustering pipelines for different data sets. The term 'before cluster' refers to the performance of clustering before clique extraction (only score normalization has been applied). The term 'after cluster' refers to the performance of clustering after clique extraction (both score normalization and clique extraction have been applied). When the best overall performance is achieved (with corresponding FPR $8*10^{-3}$), the score normalization contributes to the $\sim 70\%$ of the performance gain, while the clique extraction contributes the other $\sim 30\%$.

Surprisingly, the performance of the clique clustering pipeline on the *Rfam_LowID* data set is even better than that on the *Rfam* data set. Figure 5.3 (a) shows the F-measure of clique clustering on *Rfam_LowID* (blue) data set, which has achieved 86.4% for its peak performance. Table 5.1 shows the more detailed family-wise performance of the clique clustering. The results indicate that our clique clustering method is capable of handling low-identity ncRNA families with high accuracy. We have carefully examined the clustering results and conclude that the high performance of the *Rfam_LowID* (blue) data set is due to the exclusion of ncRNAs families that are highly similar to each other. For example, the microRNAs and snoRNAs are divided into tens of subfamilies in Rfam, which greatly reduces the clustering performance if those belonging to different subfamilies are clustered together.

The improvement of our clustering pipeline is made by normalizing the structure alignment scores and incorporating the clique finding algorithm in clustering. It is important to understand the contribution of each step to the improvement of overall performance, as the answer may provide insights of this problem and lead to more desirable applications of the pipeline. To separate the contributions of these two steps, we use the ROC (Receiver Operating Characteristic) curves, which is generated by plotting true positive rate versus false positive rate, to represent the clustering performances that are: 1) after structure alignment score normalization; and 2) after score normalization and clique finding clustering. We named the first performance as 'before cluster', and the second performance as 'after cluster'. To draw the ROC curve, we define true positive for 'before cluster' as the number of edges that connects two vertices whose corresponding ncRNAs are clustered in the same RNA family (as defined by Rfam) in the original graph, and for 'after cluster' as the number of ncRNA pairs that are clustered (by us) in the same group and in the same RNA family (as defined by Rfam). The false positive, true negative, and false negative are defined correspondingly.

We show the ROC curves in Figure 5.3 (b). In Figure 5.3 (b), we can observe that when the best overall performance is achieved (where the FPR is $8 * 10^{-3}$), the score normalization contributes $\sim$ 70% of the performance gain (subtracting the value of the red line with triangular labels from the value of the green line with round labels), while the clique extraction contributes the other $\sim$ 30% of the performance gain (subtracting the value of the green line with round labels from the value of the green line with triangular labels). We can also observe that the performances for 'after cluster' are higher than 'before cluster' at the low false positive rate range for both *Rfam* and *Rfam_LowID* data sets. This is because with stringent P-value cutoff, the merging step of the `CLCL` algorithm can correct some false negatives. On the other hand, with a loose P-value cutoff, the merging step will produce more false positives than the false negatives which it may reduce. As a result, it is more desirable to apply relatively strict P-value cutoff to the clustering pipeline.

5.3.2 Finding ncRNA Elements in D. melanogaster 3'-UTR

After benchmarking the clique clustering pipeline on the Rfam data sets, we applied it to the real ncRNA candidates generated from *D. melanogaster* 3'-UTR (with P-value cutoff 0.01). We identified 524 significant clusters that contain at least three structural elements at the beginning. To further assure the clusters' quality, we first removed the overlapping sequences, which are included by the candidate screening strategy used by `RNAz` discovery pipeline. We also ensured that the local region aligned within each cluster is consistent. To extract the consistently aligned local regions, we re-performed the pairwise alignments on the clustered ncRNA candidates. We represented each candidate by its longest local region that was commonly (aligned to all other candidates in the cluster) and structurally (annotated

as structured region) aligned. If such region is too short (<60% of the longest local common structural region within the cluster) or does not exist, we removed the corresponding candidate from the cluster. This process was carried out iteratively until a high quality consensus local structural region was identified, or the number of potential candidates dropped below three. Finally, we collected 184 ncRNA clusters with high confidence.

We sorted the 184 clusters based on their average in-cluster P-values. For each cluster, we used `mLocARNA` to generate the corresponding multiple alignments on their commonly aligned local regions without structural constraint. We also used `RNAz` to evaluate the quality of the multiple alignments. Since the multiple alignments were generated using a structural alignment approach, we chose a di-nucleotide background model and a structural RNA alignment quality decision model of the `RNAz` for evaluation [58]. We identified 168 (91.3% of all identified clusters) clusters that have `RNAz` RNA class probability value >0.95, indicating potential true structural elements in these clusters. (For more detailed information including consensus structures of the clusters and GO term analysis please refer to our supplementary website: `http://genome.ucf.edu/fly3UTRcluster`.) In addition, we have also provided the differentiated expression information of each cluster of genes in terms of different tissues, based on the experimental results and T-test performed by FlyAtlas [26].

5.3.2.1 Histone Stem-loop Clusters

The two clusters that are ranked top among all 184 clusters correspond to the histone 3'-UTR stem-loop structures [41]. The histone genes are divided into five major subfamilies: His1, His2A, His2B, His3 and His4. There are 23, 20, 23, 23 and 22 genes annotated as the

five subfamilies by FlyBase, respectively. Only 13 His1 genes' and 18 His2A genes' 3'-UTR were included in the candidate regions after `RNAz` screening (possibly due to the flanking sequence contamination). The first cluster (C1) contains 10 out of 13 annotated His1 genes and one other gene, while the second cluster (C2) contains 18 out of 18 annotated His2A genes and three other genes. The three missed His1 genes are clustered together in cluster C7.

While the known histone 3'-UTR structural elements have been rediscovered with high accuracy, the annotation of the remaining clusters is more challenging as they contain many un-annotated genes. However, we were still able to identify several interesting clusters with significant functional enrichments, as we will present in the following.

5.3.2.2 Cluster of Genes that are Preferentially Expressed in Drosophila Testis

Gene cluster C19 is a striking example of a cluster of 20 transcripts with functionally related genes (see Table 5.2). Many of the genes in this cluster show either a male-biased and/or testes-enriched expression pattern (see Figure 5.4 (a)), and/or localized expression in post-meiotic spermatids. Of the genes for which data is available, 65% (11/17) show male biased expression (fold enrichment: min 5-fold, max 6,762-fold, median 734-fold), 69% (9/13) show expression enriched in testes compared to ovaries (fold enrichment: min 3-fold, max 772-fold, median 175-fold), and 80% (4/5) show a highly specific expression pattern in spermatids (see Table 5.2). The spermatid expression is very specific with transcription occurring in post-meiotic spermatids and subcellular localization of the mRNA (described as either 'cup' or 'comet') to the distal region of spermatids [13]. This expression pattern is also highly un-

usual and was only observed in 24 testes-expressed genes (among 529 genes that have been investigated). Given the fact that our cluster contains only five genes which have been investigated, and four of them exhibit the 'cup' or 'comet' localization pattern (see Figure 5.4 (b)), hypergeometric test indicates that the probability to observe this result by chance is less than $1.6 * 10^{-5}$. The enrichment of genes with male-biased expression pattern in this cluster and their highly specific localization patterns, suggest the potential post-transcriptional regulation induced by their common 3'-UTR ncRNA elements.

To further confirm the correlation between the 3'-UTR ncRNA element and the genes' expression patterns, we conducted a search for genes with similar 3'-UTR elements. We used `cmsearch` [97] to search the 3'-UTR ncRNA element profile against the entire 3'-UTR of the *D. Melanogaster* genome. We identified two candidate genes: CG12993 and CG15059. The first ncRNA element lies in 105bp downstream of the translational ending site of CG12993. The gene CG12993 is called *presidents-cup*, which also shows the 'cup' expression pattern in spermatids [13]. The expression of the gene is highly male-biased as well, with 1,549 expression level for adult male of 5 days, and 2 for adult female of 5 days. Furthermore, this gene is annotated to be highly expressed in testis by FlyBase. The second ncRNA element strides over the translational ending site of CG15059. The gene CG15059 is also highly male-biased expressed, showing expression level of 1,497 for adult male of 5 days, and 0 for adult female of 5 days. These evidences further support the correlation between the 3'-UTR ncRNA element and these genes' expressions and functionalities. The multiple structural alignment of the 3'-UTR structured elements of these genes, and the consensus secondary structure, are shown in Figure 5.4 (c).

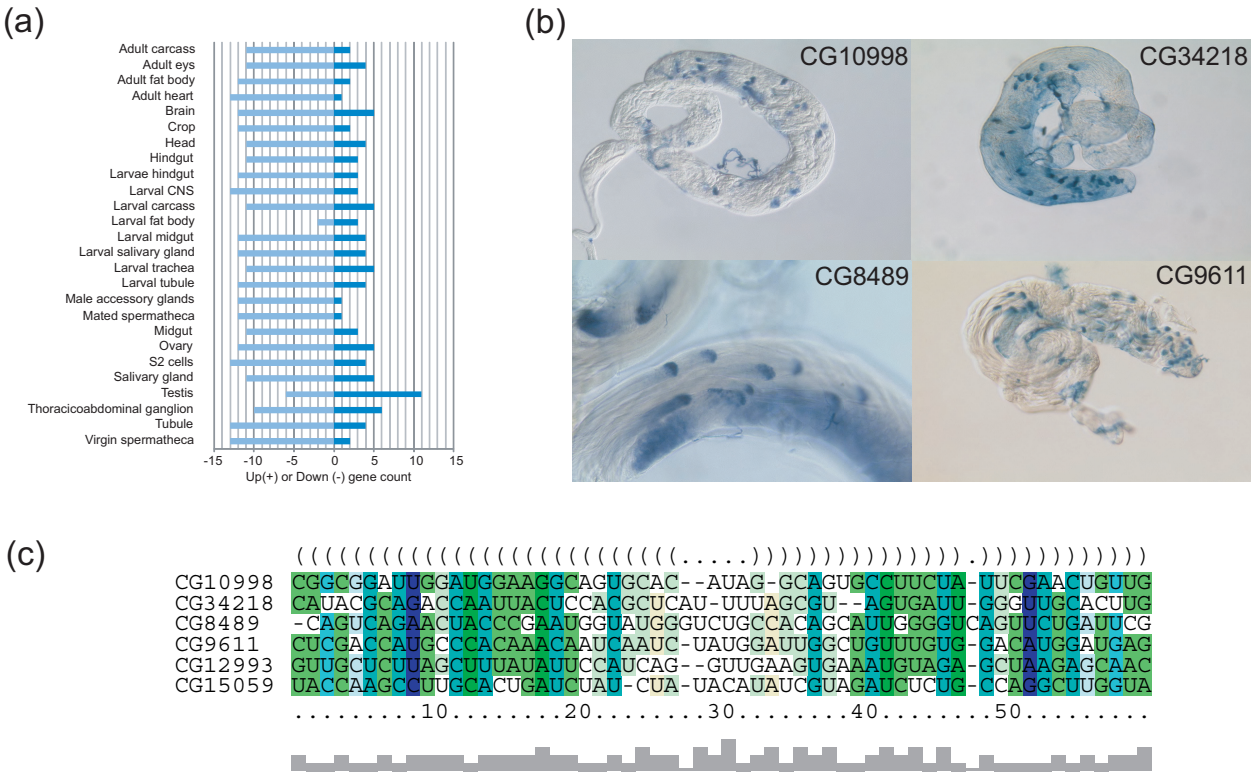

Figure 5.4: Functional inferences of the genes clustered in C19. (a) FlyAtlas expression levels of the genes clustered in C19 in different tissues. (This figure is generated by searching FlyMine [86] with all genes that are clustered in C19.) A majority (eleven) of these genes are highly expressed in fly testis, while no similar pattern can be observed for the other tissues. (b) The 'cup' or 'comet' localization patterns of four genes identified by 3'-UTR RNA clustering in fly testes. These four images were created in the laboratory of Dr. Helen White-Cooper, are copyright © Helen White-Cooper, and were first published in FlyTED, the *Drosophila* Testis gene Expression Database (`http://flyted.zoo.ox.ac.uk/`), from which these copies were obtained [151]. (c) The consensus secondary structure and multiple alignments of the 3'-UTR RNA elements of the four genes that are shown in (b), and two high-score hits that have been identified by searching the secondary structure profile against 3'-UTR of *Drosophila melanogaster* genome using `cmsearch` [97].

Table 5.2: The expression profile of genes clustered in C19 and the consensus structure and multiple alignments of their conserved 3'-UTR RNA elements

FlyBase ID	CG ID	Symbol	Expression Profile				
			modENCODE[1]		FlyAtlas[2]		FlyTED[3]
			Adult males 5 days	Adult females 5 days	Testis	Ovary	Spermatogenesis
FBgn0004403	CG1524	*RpS14a*	32115	53897	705	2785	nd
FBgn0010316	CG1772	*dap*	309	1813	45	1117	nd
FBgn0028487	CG9611	*f-cup*	5786	755	1419	148	cup
FBgn0029809	CG15767	CG15767	734	0	134*	1	nd
FBgn0031142	CG10998	*r-cup*	2008	14	nd	nd	cup
FBgn0031546	CG8851	CG8851	4241	2	nd	nd	nd
FBgn0032176	CG13127	CG13127	360	0	175*	1	nd
FBgn0033848	CG13330	CG13330	nd	nd	895*	3	nd
FBgn0034374	CG15086	CG15086	5501	0	1237*	2	nd
FBgn0036687	CG6652	CG6652	9250	6	1544*	2	spermatocytes
FBgn0038170	CG14367	CG14367	1889	364	29	11	nd
FBgn0038225	CG8489	*soti*	6762	0	143*	2	comet
FBgn0038499	CG31256	*Brf*	470	932	9	94	nd
FBgn0038683	CG11779	CG11779	4905	2739	nd	nd	nd
FBgn0062517	CG16984	CG16984	6630	230	1393*	3	nd
FBgn0086358	CG7417	*Tab2*	1554	3470	87	382	nd
FBgn0250827	CG34218	*whip*	5358	1	nd	nd	comet
FBgn0261799	CG32159	*dsx-c73A*	nd	nd	nd	nd	nd
FBgn0262515	CG8029	*VhaAC45*	nd	nd	nd	nd	nd
FBgn0262740	CG11727	CG11727	nd	nd	nd	nd	nd

[1]modENCODE RNA-Seq data were downloaded from Flybase (average RNA-Seq RPKM reported in FlyBase Annotation Release 5.26) [56]. [2]FlyAtlas microarray expression data was downloaded from FlyBase (Annotation Release 5.26) [26]. *Genes with strong expression are confined to the testis and low expression in the fat body. [3]RNA tissue *in situ* hybridization data obtained from Fly-TED [151].

Table 5.3: Expression profile of the gene cluster C37

FlyBase ID	CG ID	Symbol	mRNA signal level (fold enrichment to whole fly)					
			Head	Eye	Crop	Male acc.[1]	Virgin sp.[2]	Mated sp.[3]
FBgn0083975	CG34139	CG34139	4 (2.4)	2 (1.5)	2 (1.3)	3 (2.3)	1 (0.7)	1 (0.7)
FBgn0001987	CG3903	*Gli*	234 (2.6)	378 (4.2)	311 (3.4)	157 (1.7)	219 (2.4)	343 (3.8)
FBgn0260659	CG4196	CG4196	481 (1.3)	749 (2.1)	398 (1.1)	694 (1.9)	412 (1.1)	416 (1.2)
FBgn0001219	CG4264	*Hsc70-4*	3873 (1.0)	6556 (1.7)	6037 (1.5)	4610 (1.2)	4690 (1.2)	4930 (1.3)
FBgn0035914	CG6282	CG6282	278 (8.5)	611 (18.6)	31 (0.9)	72 (2.2)	6 (0.2)	7 (0.2)
FBgn0031515	CG9664	CG9664	74 (2.8)	55 (2.1)	78 (2.9)	9 (0.4)	115 (4.3)	73 (2.7)

The shaded cells in the table indicate the genes that are significantly (based on FlyAtlas T-test) enriched in the specific tissues. FlyAtlas microarray expression data was downloaded from FlyBase (Annotation Release 5.26) [26]. [1]Male accessory gland. [2]Virgin spermetheca. [3]Mated spermetheca.

5.3.2.3 Clusters of Genes that are Essential for the Functions of Septate Junction

Gene cluster C37 contains six genes that share a common 3'-UTR element shown in Figure 5.5. These genes may play important roles for maintaining the proper function of septate junction in *Drosophila*, which is responsible for the formation of paracellular diffusion barrier. The first gene CG34139 is suggested to code for a transmembrane protein neuroligin by FlyBase report, based on its sequence homology to human neuroligin gene. Neuroligin acts as ligands for neurexin, which is also a transmembrane protein that is known to glue together neurons at the synapse. Alternations of these two genes will cause a cognitive disease in human [123]. The second gene CG3903 (also known as *Gli*), codes for gliotactin protein, which is critical in forming blood-nerve barrier [7]. This protein is almost exclusively expressed in neuroglia cells which maintain the proper external environment and provide support and protection for the neurons in the brain. The third gene CG9664 is annotated with the biological function of lipid metabolic process and lipid transport [110]. The gene has also been suggested by OrthoDB [138] to code for a membrane protein that has ATP binding potential and ATPase activity. These genes (i.e. neurexin, gliotactin, and ATPase) are responsible for maintaining the extracellular environment through the formation of paracellular diffusion barrier, and are essential for septate junction function in *Drosophila* [54]. The fourth protein CG4264 (Heat shock 70-kDa protein cognate 4 or Hsc70-4) has also been found to express in neuroglia cells [114]. This gene is responsible for the protection of synapse under high temperature [71], and it is possible that the protein is also responsible for protection paracellular diffusion barrier in other tissues. The functions of other two genes, CG4196 and CG6282, are not annotated, but they are inferred as membrane and lipid metabolic process related proteins by FlyAtlas curators, which are possibly also responsible for maintaining the paracellular diffusion barrier.

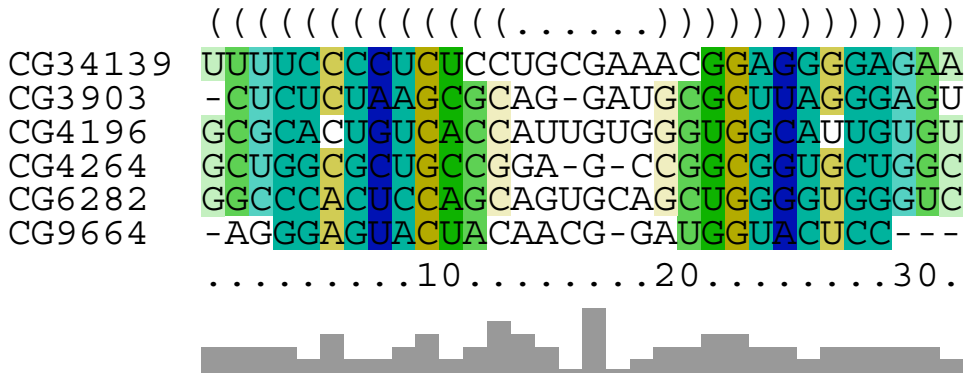

Figure 5.5: The consensus secondary structure and multiple alignments of the 3'-UTR RNA elements of all six genes that have been clustered in C37.

We investigated the expression profiles of the genes in C37 from FlyAtlas [26], and outline their expressions in head, eye, crop, male accessory gland, and spermetheca (both virgin and mated) in Table 5.3. The gene CG34139 has extremely low expressions in all tissues, whose exact expression level may be difficult to measure by microarray technique. Therefore, we exclude this gene from our studies. We found that 80% (4/5) of the genes in this cluster show enriched expression in head. On the other hand, only 40% (2/5) of them show increased expressions in brain. This indicates that the genes in this cluster may participate in the maintenance of paracellular diffusion barrier in the head rather than the central nervous system, for example, in the eye where all genes (5/5) show significant enrichment. Besides its important functions in the nervous system, paracellular diffusion barrier is also known to be required for proper nutrition absorption or secretion [48, 50]. Indeed, these genes also show enriched expression in crop, male accessory gland, and spermetheca (both virgin and mated) where secretion appear to be important for maintaining the proper physiological environment (see Table 5.3). Investigating the commonalities of the physiological environments in these tissues may help to elucidate these gene's specific functions and interactions.

5.4 Conclusions

In this work, we are particularly interested in finding 3'-UTR ncRNA elements that may direct post-transcriptional regulation in the *D. melanogaster* genome. We have improved the existing clustering pipeline by normalizing the structural alignment scores through simulation and adopting the clique-finding style clustering algorithm. We performed benchmark tests against the `LocARNA` hierarchical clustering pipeline to demonstrate the performance improvement made by our new clustering method. Then we applied the improved clustering pipeline to 3'-UTR of the *D. melanogaster* genome and revealed 184 ncRNA element clusters. We identified two interesting clusters, where one cluster contains genes that are highly expressed in male *Drosophila*, and the other contains genes that are essential for septate junction function in *Drosophila*. These findings have significantly enriched our current understanding of the 3'-UTR ncRNA elements and their correlation with post-transcriptional regulation.

Although structural conservation scored by `RNAz` indicates high clustering accuracy, it remains challenging to conduct functional analysis for the identified clusters. The mechanism of localization can be very sophisticated, and 3'-UTR element may not be the only one that directs the regulation. For example, in Rabani *et al.*'s study [104], only 9 conserved 3'-UTR RNA elements were identified from 94 sets of genes that are experimentally verified to be co-localized. We plan to apply this clustering pipeline to other genomic locations that may affect localization, for example 5'-UTR, to discover more RNA elements. The difficulty of annotation is also due to the presence of many un-annotated genes. For example, we tried to use functional enrichment analysis tools such as `g:profiler` [106] and `Ontologizer` [15], and pathway searching tools such as `IPA` (Ingenuity Pathway Analysis), to reveal poten-

tial correlations between the genes within a cluster. But most of the queries failed due to incomplete gene annotation. We also tried to map the gene clusters using experimental co-localization data [77], yet similarly, only a few of the genes appear to be well studied. As the functionalities of these genes are elucidated, we expect that more clusters can also be biologically explained. We also expect that researchers will refer and design experiments to confirm our predictions.

Finally, we observed that two issues still await to be solved to improve the existing clustering pipeline. First, the candidate regions for ncRNAs may be mis-predicted, which will likely reduce the clustering accuracy. For example, `RNAz` is known to have a high false positive rate [58], which may include many non-RNA elements in the candidate set and contaminate the clustering analysis. We can improve the clustering pipeline at this point by incorporating next-generation sequencing data, where the regions in the genome that are actively transcribed can be experimentally detected. Second, the computational bottleneck of the entire clustering process lies at the pairwise alignment of all candidate RNA elements. Existing alignment tools either have limited accuracy, or satisfying accuracy but with a high computational overhead. To resolve this issue, we propose to incorporate the sparse dynamic programming technique used in RNA folding [141] and co-folding [8, 152] to speedup existing alignment algorithms with high accuracy, and devise a more efficient alignment algorithm for clustering analysis. We anticipate that these improvements will enable clustering analysis on larger and more sophisticated data sets, and lead to further interesting discoveries.

CHAPTER 6: EFFICIENT ALIGNMENT OF RNA SECONDARY STRUCTURES

In Chapter 5, we have described a clustering pipeline for genome-wide clustering of RNA secondary structure elements, and developed the clique extraction algorithm called `CLCL`. The bottleneck for this clustering pipeline lies on the all-against-all pairwise alignments of the candidate RNA secondary structures. The computational efficiency of the RNA secondary structure alignment algorithm is thus of high importance for applying this clustering pipeline to long ncRNAs and large data sets.

In this chapter, we describe an efficient RNA secondary structure alignment algorithm to solve this issue. The new algorithm is developed using a sparse dynamic programming technique. Importantly, the speedup is achieved without sacrificing the alignment quality. We benchmark our new RNA secondary structure alignment tool, called `ERA`, with other state-of-the-art RNA secondary structure alignment algorithms. The benchmark results indicate that `ERA` is capable of producing high-quality alignment with significantly improved computational efficiency.

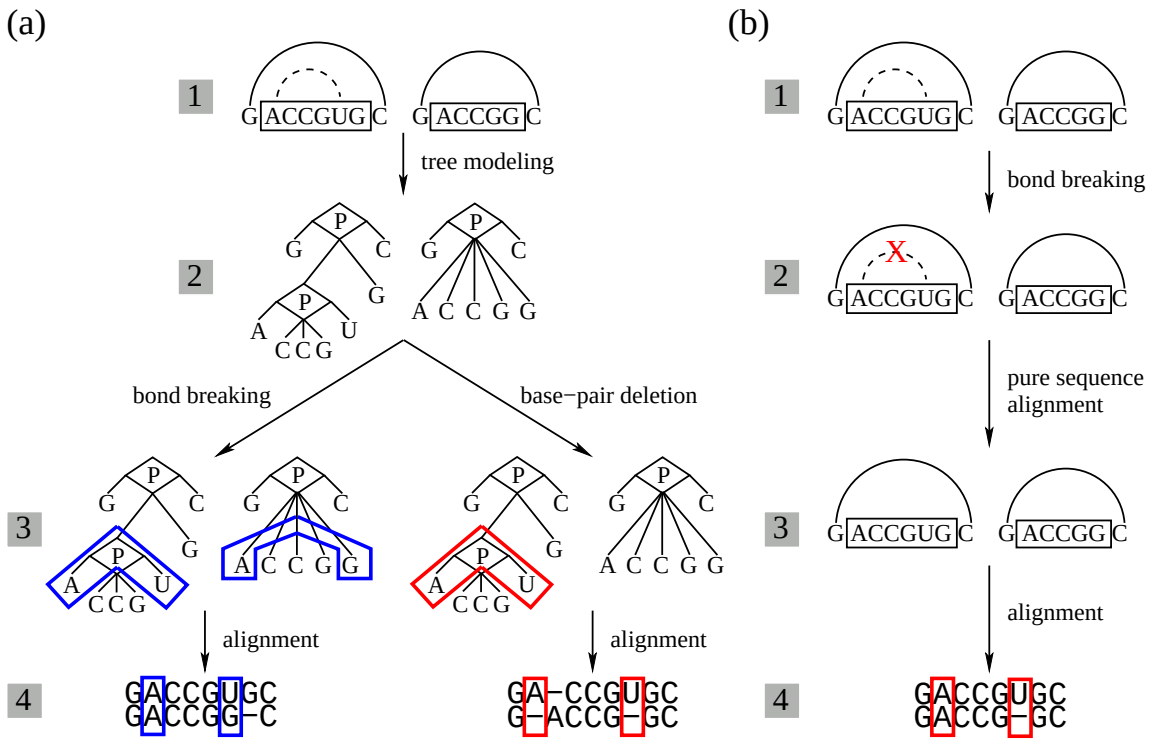

Figure 6.1: Comparison between the tree-based alignment approach and the SAF-style alignment approach in handling mis-predicted base pairs. (a) The tree-based alignment algorithm in handling mis-predicted base pairs. Row 1: The arcs on the sequences indicate the base pairs (solid arc indicates real base pairs, while dashed arc indicates mis-predicted base pairs). The structure regions indicated by the boxes are being aligned. Row 2: The two RNA structures are modeled into trees according to `RNAforester` [63]. The 'P' node was introduced to represent a base pair. Row 3: Either the bond breaking or the base-pair deletion operation is taken. The blue boxes indicate the aligned nucleotides in the bond-breaking case. The red box indicates the base pair (including its nucleotides) being deleted in the base-pair deletion case. Row 4: The corresponding alignments resulted from both operations. The boxes in the alignments correspond to those in the RNA structure trees. Neither of the alignments is correct. (b) The SAF-style alignment algorithm in handling mis-predicted base pairs. Row 1: The same RNA structures are being aligned. Row 2: The base-pair interaction is deleted (red cross), leaving two free nucleotides. Row 3: The sequence similarity between the boxed regions is assessed using a traditional sequence alignment algorithm [98]. Row 4: The corresponding alignment is generated correctly. The boxes correspond to nucleotides that form the mis-predicted base pair.

6.1 General Edit-Distance RNA Secondary Structure Alignment with Sparse Dynamic Programming

Existing computational approaches for RNA structure-structure alignment can be classified into two major categories: the tree-based alignment algorithms [63, 68, 126, 148] and the structure alignment application of the simultaneously alignment and folding (SAF) algorithms [10, 38, 65, 73, 91, 111, 129, 142]. Despite their original intention of solving the SAF problem, many of the SAF algorithms can align fixed RNA structures by simply modifying the inputs. Therefore, we refer to these SAF algorithms as the *SAF-style alignment algorithms* throughout this chapter to emphasize their structure alignment applications. The tree-based alignment algorithms model each RNA structure into a tree, and adopt the tree edit-distance algorithm to find the optimal alignment between the RNA structure trees. The time complexity of this category of algorithm is $O(l^3)$ (where l is the average sequence length of the RNA structures). Such complexity is shown by the optimal decomposition technique proposed by Demaine *et al* [35]. On the other hand, the SAF-style alignment algorithms generate the RNA structure-structure alignment by simply restricting the inputs to fixed RNA structures instead of RNA structure ensembles. The time complexity of these algorithms is $O(l^4)$ [9], and is achieved by assuming $n = O(l)$ (where n is the average number of base pairs in the structures). Depending on specific implementations, some of the SAF alignment algorithms have a $O(l^4 + n^2l^2)$ time complexity (such as `PMcomp` [65], `LocARNA` [142], and `FOLDALIGNM` [129]), while the others have a $O(n^4 + n^2l^2)$ time complexity (such as `RNAscf` [10]).

Despite their higher time complexities, the SAF-style alignment algorithms usually generate high-quality alignment results compared to the tree-based alignment algorithms. This is

because the tree-based RNA alignment algorithms are sensitive to the mis-predicted base pairs. Recall that the RNA structure trees are built prior to the application of the tree edit-distance DP algorithm. In this case, once the RNA structure trees are built, they are impossible to repair under the DP scheme (which assumes subproblem optimality). We demonstrate such a problem with an artificial example, shown as follows.

Consider that the two RNA structures shown in Figure 6.1 (a) are being aligned using a tree-based alignment algorithm. In the first structure, due to the insertion of a uracil (U), an additional base pair is predicted (dashed arc, Row 1). Both structures are enclosed by G-C base pairs, and we focus on the alignment of their inner regions (boxed regions, Row 1). Following `RNAforester`'s extended tree representation [63], the two RNA structures can be transformed into two trees (Row 2). The 'P' node represents a base pair formed between the two corresponding nucleotides. Because there is no base pair in the second structure, the only allowed operations are bond breaking and base-pair deletion (Row 3). For the bond breaking operation, the base pair formed between A and U is broken, leaving them aligned to A and G in the second structure, respectively (blue boxes, Row 3). The alignment between the U (first structure) and G (second structure) introduces an unnecessary mismatch, making the alignment incorrect (blue boxes, Row 4). For the base-pair deletion operation, the entire base pair (including the two nucleotides A and U) is deleted (red box, Row 3). This operation opens two unnecessary gaps in the alignment (red boxes, Row 4), making it underestimate the real structural similarity.

In contrary, the SAF-style alignment algorithms handle the mis-predicted base pairs in a more straightforward way. As shown in Figure 6.1 (b), they simply break the the base pair interaction and disassociate the two corresponding nucleotides completely (red cross, Row 2). These two nucleotides are then treated as regular unpaired nucleotides. The SAF-style

alignment algorithm uses the standard sequence alignment algorithm [98] to evaluate the pure sequence similarity between the boxed hairpin-loop regions (Row 3). The resulting alignment contains only one gap, and correctly interprets the true structural difference between the two RNA structures (red boxes, Row 4).

The above example shows that the SAF-style alignment algorithms can produce more accurate alignments than the tree-based alignment algorithms. However, they do trade such advantage with higher time complexity ($O(l^4)$ compared to $O(l^3)$). In this case, an ideal scenario to devise an $O(l^3)$ SAF-style algorithm that can generate accurate alignment results. To achieve this goal, we incorporate the sparse DP technique into the SAF algorithm `RNAscf` [10]. Using this technique, we can reduce the original time complexity by a factor of n^2 to z, where $n < z \ll n^2$ under the assumption of the *polymer-zeta* property of RNA molecules [141]. In this case, the new SAF-style RNA structure-structure alignment algorithm will have a time complexity of $O(zn^2 + zl^2)$. The new time complexity has an expected cubic ($z = O(n) = O(l)$) growth behavior, and is similar to those of the tree-based alignment algorithms. In addition, we also devise a novel online pruning technique to further speedup the new algorithm, which deletes obsolete candidates on-the-fly. By combining both speedup techniques, the new SAF-style RNA structure alignment algorithm is capable of comparing RNA secondary structures efficiently and accurately.

We have implemented the proposed RNA structure alignment algorithm into a program called `ERA` (Efficient RNA Alignment). The benchmark results showed that `ERA` has the expected $O(zl^2)$ time complexity. We showed the $O(zl^2)$ time complexity of `ERA` through aligning Rfam [57] RNA structures that were carefully chosen to represent a wide rage of input sizes. We also used BraliBase II [52] to benchmark the alignment quality between `ERA`, `LocARNA` and `RNAforester`. Nearly identical alignment quality can be observed for

the SAF-style alignment tools ERA and LocARNA, while both of them are more accurate than the tree-based alignment algorithm RNAforester. Finally, we also concluded that ERA is efficiently implemented by observing an average of 10 fold speedup over LocARNA, and RNAforester in terms of real RNA structure alignments. Based on these results, we confirmed that the sparse DP technique and the online pruning technique are successfully incorporated into the original RNAscf algorithm. We also anticipate that ERA will become an important bioinformatics tool for comparative RNA structure analysis.

6.2 Methods

In this section, we will present a novel SAF-style RNA structure alignment algorithm by incorporating the sparse DP technique into the RNAscf algorithm. RNAscf was originally designed to identify the consensus structure between two RNA sequences. It guides the DP process though stacks and has a time complexity of $O(n^4+n^2l^2)$. Comparing to LocARNA (which has a time complexity of $O(l^4+n^2l^2)$), the indexing scheme used by RNAscf makes it easier to incorporate the sparse DP technique, which aims to reduce the size of n instead of l. In addition to the sparse DP technique, we will also present an online pruning technique, which tries to reduce the search space of the algorithm as the DP proceeds. Through combining these two speedup techniques, the novel algorithm will have an expected $O(zl^2)$ time complexity, where $n < z \ll n^2$.

The Methods section is organized as follows: In Section 6.2.1, we will give the basic definition of RNA structures and the RNA alignment problem. In Section 6.2.2, we will reintroduce the RNAscf algorithm as a basis to understand the novel algorithm that is developed in this

work. In Section 6.2.3, we will present the triangular inequality in RNA alignment with necessary proofs, which serves as a theoretical foundation for the sparse DP technique. In Section 6.2.4, we will further discuss the implementation details of incorporating the sparse DP technique. In Section 6.2.5, we will present the novel RNA alignment algorithm with the incorporation of the sparse DP technique. In Section 6.2.6, we will present the online pruning technique as an additional speedup step to the novel algorithm. Finally, in Section 6.2.7, we will summarize the new algorithm using pseudo-code that can be directed implemented.

6.2.1 Preliminaries and Definitions

We will begin with the introduction of the basic symbols and notations. The secondary structure of an RNA A of length l_A is represented by a set of base pairs in A, denoted as $\mathcal{P}^A$. A base pair $p^A \in \mathcal{P}^A$ is an interaction formed between two nucleotides in the sequence of A, whose positions are denoted by $l(p^A)$ and $r(p^A)$ (without loss of generality, we assume $l(p^A) < r(p^A)$). The base pair p^A can also be represented as $(l(p^A), r(p^A))$. The base pairs are partially ordered by the increasing order of their ending nucleotides, i.e. $p_i^A < p_j^A$ if and only if $r(p_i^A) < r(p_j^A)$. Since we do not consider RNA ensembles, no crossing base pair is allowed. That is, we do not allow $l(p_i^A) < l(p_j^A) < r(p_i^A) < r(p_j^A)$. The two base pairs p_i^A and p_j^A are either *enclosing* or *juxtaposing to* each other. The base pair p_j^A encloses p_i^A if $l(p_j^A) < l(p_i^A) < r(p_i^A) < r(p_j^A)$, denoted as $p_i^A <_I p_j^A$. The base pair p_i^A juxtaposes to and *before* p_j^A if $r(p_i^A) < l(p_j^A)$, and is denoted by $p_i^A <_J p_j^A$.

We also define loop regions (i.e. hairpin loop, internal/bulge loop, and multi-branch loop) whose sequence similarities are assessed by the alignment. The loop regions can be viewed as

the unpaired regions in the RNA sequence that are segregated by the paired nucleotides. Let $A[i...j]$ denote a continuous sequence region in RNA A, which begins with the ith nucleotide and ends with the jth nucleotide. Define $L(p^A)$ as the sequence $A[l(p^A)+1...r(p^A)-1]$ (hairpin loop). If $p_i^A <_I p_j^A$, define $L_l(p_i^A, p_j^A)$ as the sequence $A[l(p_j^A)+1...l(p_i^A)-1]$, and $L_r(p_i^A, p_j^A)$ as the sequence $A[r(p_i^A)+1...r(p_j^A)-1]$ (internal or bulge loop). If $p_i^A <_J p_j^A$, define $L(p_i^A, p_j^A)$ as the sequence $A[r(p_i^A)+1...l(p_j^A)-1]$ (multi-branch loop).

The structure alignment between RNA A and B is the optimal matching between their base-pair sets $\mathcal{P}^A$ and $\mathcal{P}^B$ and the corresponding loop similarities. In other words, the alignment between RNAs A and B is a one-to-one binary relation $\mathcal{A}$ on the base-pair sets $\mathcal{P}^A$ and $\mathcal{P}^B$. To ensure that the alignment will not lead to conflicting base-pair matchings, for any $(p_i^A, p_{i'}^B) \in \mathcal{A}$ and $(p_j^A, p_{j'}^B) \in \mathcal{A}$, either $p_i^A <_I p_j^A$ and $p_{i'}^B <_I p_{j'}^B$, or $p_i^A <_J p_j^A$ and $p_{i'}^B <_J p_{j'}^B$. Given the alignment $\mathcal{A}$, the matched base pairs in $\mathcal{A}$ will partition the RNA sequences A and B into two sets of loop regions, $\mathcal{L}_{\mathcal{A}}^A$ and $\mathcal{L}_{\mathcal{A}}^B$, respectively. The sequence similarity between these two sets of loop regions is added to compute the overall alignment score. The optimal alignment is the relation $\mathcal{A}$ that maximizes overall alignment score M that combines both structure and sequence similarities:

$$M = w_1 * \sum_{(p^A, p^B) \in \mathcal{A}} S_{str}(p^A, p^B) + w_2 * \sum S_{seq}(\mathcal{L}_{\mathcal{A}}^A, \mathcal{L}_{\mathcal{A}}^B). \tag{6.1}$$

Here, the first term is the summation of all structural similarities (S_{str}) between the annotated base pairs. The structural similarity score for base-pair substitution is set using the RIBOSUM matrix [76], denoting such base-pair substitution matrix as R. We do not give penalty for base-pair deletion or insertion, as we may expect incorrectly predicted base pairs in the input RNA structures. The second term is the summation of the sequence similarities (S_{seq}) on all loop (unpaired) regions that are determined by base-pair matchings in $\mathcal{A}$.

The sequence similarity between two sequence regions is computed as traditional sequence alignment, with D as a 4-by-4 matrix that accounts for nucleotide substitution (set using the RIBOSUM matrix), g as the gap opening penalty, and e as the gap extension penalty [96] (g and e are both set to negative values and $g < e$). The weights w_1 and w_2 are used to balance the structural and sequence contribution to the overall alignment score, and we set $w_1 > w_2$ to emphasize structural similarity. To simplify the expressions, in the rest of this chapter, we assume that w_1 has been multiplied by all structural similarity terms (R), and w_2 has been multiplied by all sequence similarity terms (D, g, and e).

We will now define the matrices that are used by the DP algorithm. Denote $M[p^A, p^B]$ as the optimal structure alignment score between the regions enclosed by p^A and p^B, given that p^A is matched with p^B. Denote $M_h[p^A, p^B]$ as the optimal alignment score when the matching of p^A and p^B corresponds to a hairpin loop in the consensus structure. Similarly, $M_l[p^A, p^B]$ stores the optimal alignment score when the matching of p^A and p^B corresponds to an internal, a bulge, or a multi loop in the consensus structure. Assume that $p_i^A <_I p^A$, and $p_{i'}^B <_I p^B$, $M_l[p^A, p^B]$ can be computed by referring to the matrix $M_c[p_i^A, p_{i'}^B]$, which stores the optimal alignment score between the juxtaposed base-pair *chains* (each chain contains at least one base pair) that end with p_i^A and $p_{i'}^B$, respectively. The optimal alignment between A and B can be retrieved from $M[p_0^A, p_0^B]$, where p_0^A and p_0^B are pseudo base pairs such that $p_0^A = (0, |A| - 1)$, $p_0^B = (0, |B| - 1)$, and $S_{str}(p_0^A, p_0^B) = 0$ [10].

6.2.2 The Original $O(n^4 + n^2l^2)$ Algorithm

In this section, we briefly reintroduce the `RNAscf` [10] algorithm for RNA consensus structure prediction as a basis for understanding the novel algorithm developed in this work. The recursive functions for the `RNAscf` algorithm are outlined as follows:

$$M[p^A, p^B] = \max \begin{cases} M_h[p^A, p^B], \\ M_l[p^A, p^B]. \end{cases} \tag{6.2}$$

$$M_h[p^A, p^B] = S_{str}(p^A, p^B) + S_{seq}(L(p^A), L(p^B)). \tag{6.3}$$

$$M_l[p^A, p^B] = S_{str}(p^A, p^B) + \max_{i,i'} \left\{ M_c[p_i^A, p_{i'}^B] + S_{seq}(L_r(p_i^A, p^A), L_r(p_{i'}^B, p^B)) \right\}. \tag{6.4}$$

$$M_c[p_i^A, p_{i'}^B] = \max_{\substack{p_j^A \in \mathcal{F}(p_i^A) \\ p_{j'}^B \in \mathcal{F}(p_{i'}^B)}} \begin{cases} M[p_i^A, p_{i'}^B] + S_{seq}(L_l(p_i^A, p^A), L_l(p_{i'}^B, p^B)), \\ M_c[p_j^A, p_{j'}^B] + M[p_i^A, p_{i'}^B] + S_{seq}(L(p_j^A, p_i^A), L(p_{j'}^B, p_{i'}^B)), \\ M_c[p_i^A, p_{j'}^B] + G(|L(p_{j'}^B, p_{i'}^B)| + |L(p_{i'}^B)|), \\ M_c[p_j^A, p_{i'}^B] + G(|L(p_j^A, p_i^A)| + |L(p_i^A)|). \end{cases} \tag{6.5}$$

In these recursive functions, S_{str} denotes the structural similarity between two base pairs p^A and p^B, S_{seq} denotes the sequence similarity between two unpaired regions, and G indicates the gap penalty for completely deleting the corresponding unpaired region. Note that $G(|L|) = g + |L| * e$ if $|L| > 0$, and $G(|L|) = 0$ otherwise. The base pair set $\mathcal{F}(p_i^A)$ contains all base pairs that are *directly before* and juxtaposed to p_i^A. In other words, if $p_j^A \in \mathcal{F}(p_i^A)$, then there is no such base pair p_k^A, such that $p_j^A <_J p_k^A <_J p_i^A$. In most real scenarios, $|\mathcal{F}|$ is considered as a constant [10]. This chaining technique based on the $\mathcal{F}$ set enables us to handle the multi-loop case efficiently, by only considering $|\mathcal{F}|$ cases when computing M_c.

Recall that the input RNA sequences have an average length of l and form an average of n base pairs. This algorithm can be computed with an expected time complexity of $O(n^4 + n^2l^2)$. To see the time complexity, first note that all sequence similarity scores that are referred in the recursive functions can be computed within $O(n^2l^2)$ time. Because all loop regions are segregated by base pairs, the number of loop regions is clearly bounded by $O(n)$. Therefore, there are $O(n^2)$ combinations of loop matchings, and computing each matching requires $O(l^2)$ time using a standard sequence alignment algorithm [96]. To this point, we assume all sequence similarities are computed using $O(n^2l^2)$ time, and are stored in a matrix for constant-time lookup. Now, observe that this algorithm computes the optimal alignment by filling up the DP table M, which contains $O(n^2)$ values. Computing each value in the matrix M depends on the corresponding values of M_h, M_l, and M_c. The computation of values in matrix M_h can be finished in a constant time due to the pre-computed sequence similarities. The computation of M_l requires $O(n^2)$ time, as determined by the necessity of traversing all possible combinations i and i' (see Equation 6.4). Finally, M_c can also be expected to be computed in a constant time, as $|\mathcal{F}|$ is assumed to be a constant. In this case, the computation of matrix M requires $O(n^4)$ time. Adding up the time required to pre-compute all sequence similarities of the loops, the overall time complexity for this algorithm thus becomes $O(n^4 + n^2l^2)$.

6.2.3 Triangular Inequality and Optimal Pair Matchings

The triangular inequality property servers as the theoretical foundation for the sparse DP technique, which saves search space while maintaining the global optimality. For computational RNA studies, this technique has been used in RNA folding [141], RNA consensus

folding (SAF) [8, 152], as well as RNA-RNA interaction prediction [109] applications. In this work, our aim is to bring this technique into the RNA structure alignment application, where fixed RNA structures are considered instead of RNA structure ensembles.

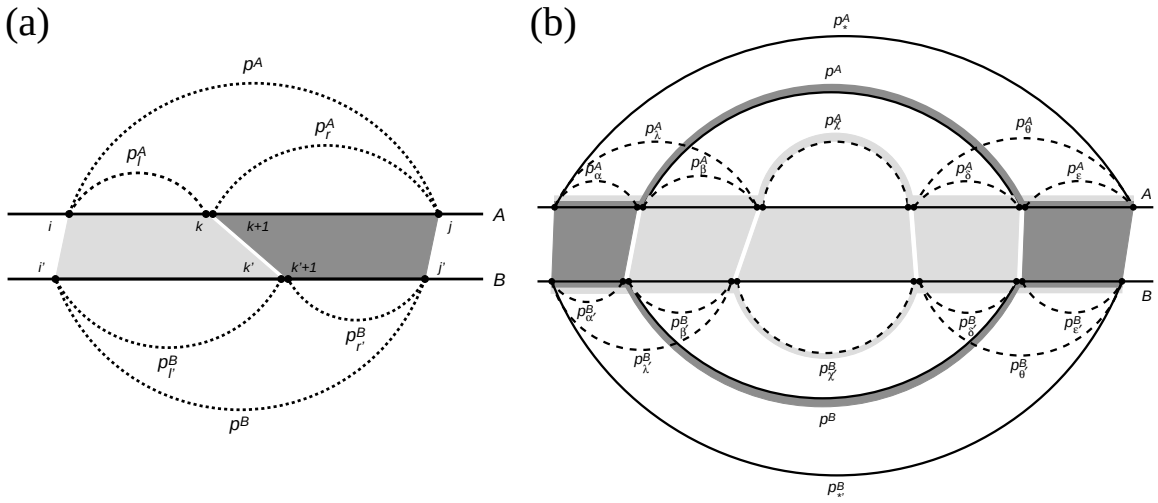

Figure 6.2: Illustration of the triangular inequality property. (a) Triangular inequality property of RNA secondary structure alignment. The horizontal lines indicate RNA sequences A and B. The dashed arcs are the pseudo base pairs added to the specific nucleotides, while the shaded areas define the correspondence between regions that are being aligned. (b) Alternative paths that go through either p^A and p^B, or p^A_χ and $p^B_{\chi'}$. The two shadings (dark and light gray) along the arcs represent the two alternative paths.

Consider the alignment between the RNA secondary structures within the two regions $A[i...j]$ and $B[i'...j']$ (see Figure 6.2 (a)). Denote $M[i, j; i', j']$ as the optimal alignment score for such alignment. The triangular inequality can be summarized using the following inequality:

$$M[i, j; i', j'] \geq M[i, k; i', k'] + M[k + 1, j; k' + 1, j'],$$

where $i \leq k < j$ and $i' \leq k' < j'$. This is because the partitions of the regions $A[i...j]$ and $B[i'...j']$ at positions k and k', respectively, do not necessarily compatible with the optimal alignment.

To simplify the expression of the triangular inequality property, we define a number of pseudo base pairs to indicate specific regions of interest. A pseudo base pair is a void interaction, such that the structural similarity between any two pseudo base pairs is defined to be 0. For instance, let p and p' be two arbitrary pseudo base pairs, we will have $S_{str}(p, p') = 0$. The pseudo base pairs are only used for the sake of representational simplicity, and are not required for the implementation of the algorithm. Define a pseudo base pair $p^A = (i, j)$ and a pseudo base pair $p^B = (i', j')$. In this case, the optimal alignment score between the regions $A[i...j]$ and $B[i'...j']$, i.e. $M[i, j; i', j']$, can be rewritten as $M[p^A, p^B]$. Similarly, define pseudo base pairs $p_l^A = (i, k)$, $p_r^A = (k + 1, j)$, $p_{l'}^B = (i', k')$, and $p_{r'}^B = (k' + 1, j')$ (see Figure 6.2 (a)). The triangular inequality can be simplified using the following observation:

Observation 1: $M[p^A, p^B] \geq M[p_l^A, p_{l'}^B] + M[p_r^A, p_{r'}^B]$.

Using Observation 1, we can detect potential redundant computations in the original algorithm. Consider the structural configurations shown in Figure 6.2 (b), and assume that the base pairs p^A and p^B are being aligned at the current stage. Let p_*^A and p_χ^A be arbitrary base pairs such that $p_\chi^A <_I p^A <_I p_*^A$. Note that p_χ^A may also represent a pseudo base pair in order to consider an arbitrary subregion enclosed by p^A. Define pseudo base pairs $p_\alpha^A = (l(p_*^A), l(p^A)-1)$, $p_\beta^A = (l(p^A), l(p_\chi^A)-1)$, $p_\delta^A = (r(p_\chi^A)+1, r(p^A))$, $p_\epsilon^A = (r(p^A)+1, r(p_*^A))$, $p_\lambda^A = (l(p_*^A), l(p_\chi^A) - 1)$, and $p_\theta^A = (r(p_\chi^A) + 1, r(p_*^A))$. Pseudo base pairs are also added to B symmetrically (see Figure 6.2 (b)). We can then prove Lemma 1 using Observation 1:

Lemma 1: If $\exists\ p_\chi^A$ and $p_{\chi'}^B$, such that $M[p_\beta^A, p_{\beta'}^B] + M[p_\chi^A, p_{\chi'}^B] + M[p_\delta^A, p_{\delta'}^B] \geq M[p^A, p^B]$, then $M[p_\lambda^A, p_{\lambda'}^B] + M[p_\chi^A, p_{\chi'}^B] + M[p_\theta^A, p_{\theta'}^B] \geq M[p_\alpha^A, p_{\alpha'}^B] + M[p^A, p^B] + M[p_\epsilon^A, p_{\epsilon'}^B]$.

Proof:

$$
\begin{aligned}
&M[p_{\lambda}^{A}, p_{\lambda'}^{B}] + M[p_{\chi}^{A}, p_{\chi'}^{B}] + M[p_{\theta}^{A}, p_{\theta'}^{B}] \\
&\geq M[p_{\alpha}^{A}, p_{\alpha'}^{B}] + M[p_{\beta}^{A}, p_{\beta'}^{B}] + M[p_{\chi}^{A}, p_{\chi'}^{B}] + M[p_{\delta}^{A}, p_{\delta'}^{B}] + M[p_{\epsilon}^{A}, p_{\epsilon'}^{B}] \\
&\geq M[p_{\alpha}^{A}, p_{\alpha'}^{B}] + M[p^{A}, p^{B}] + M[p_{\epsilon}^{A}, p_{\epsilon'}^{B}].
\end{aligned}
$$

■

The first inequality is a direct application of Observation 1, and the second inequality is specified in the condition of Lemma 1.

Because p_{*}^{A} and $p_{*'}^{B}$ are arbitrary base pairs, Lemma 1 implies that the matching between p^A and p^B is guaranteed to be suboptimal. That is, the overall alignment score, given that p^A matches with p^B, is always lower than when assuming they do not match (as the matching of p^A and p^B is conflicted with the matching of p_{λ}^{A} and $p_{\lambda'}^{B}$, as well as the matching of p_{θ}^{A} and $p_{\theta'}^{B}$). In this case, we can devise the DP algorithm to bypass the redundant references to the scenarios where p^A matches p^B. Conversely, for the implementation of this idea, the DP algorithm will refer to the scenarios of matching p^A and p^B only when the condition specified in Lemma 1 is NOT satisfied. These necessary base-pair matchings are called the *Optimal Pair Matchings* (OPMs). If the matching of p^A and p^B is an OPM, we denote this OPM as $o^{A,B}$. Similarly, we represent the OPM formed by base pairs p_i^A and $p_{i'}^B$ as $o_{i,i'}^{A,B}$. The new RNA alignment algorithm will maintain an OPM list $\mathcal{O}$, which is modified online as the DP proceeds, so as to included newly identified OPMs and remove obsolete OPMs (which will be discussed in Section 2.6). If we assume that the RNA molecules have the *polymer-zeta* property [141], restricting the search space of the DP using the OPM list $\mathcal{O}$ will reduce the time complexity of the RNA alignment algorithm to $O(zl^2)$ (as will be discussed in Section 2.5).

6.2.4 Detection of Optimal Pair Matchings

In the previous section, we have proved that Lemma 1 can be used to detect the OPMs and save redundant computations. In this section, we will briefly discuss how it will be implemented. Lemma 1 states that if the alignment score assuming p^A matches p^B ($M[p^A, p^B]$) is higher than the alignment score assuming p^A does not match p^B, the matching between p^A and p^B is an OPM. Therefore, to detect the OPMs, we need to compute two alignment scores, i.e. the one when assuming p^A matches p^B and the one when assuming p^A does not match p^B.

Based on previous definition, the first alignment score is computed as $M[p^A, p^B]$. In this case, we only need to compute the second alignment score. However, computing the second alignment score (assuming p^A does not match p^B) is difficult. Instead, we can compute the overall alignment score without assuming any restrictions. Apparently, the overall alignment score includes both cases disregarding whether p^A matches with p^B. Therefore, if $M[p^A, p^B]$ is greater than or equal to such an overall optimal alignment, it is guaranteed to be greater than the alignment score when assuming p^A does not match p^B, and ipso facto the matching of p^A and p^B is an OPM.

Recall that the alignment score $M[p^A, p^B]$ corresponds to the case where p^A matches with p^B, and therefore it can be decomposed as the sum of two parts: the structure similarity between the two base pairs themselves $S_{str}(p^A, p^B)$, and the optimal alignment score between the regions $A[l(p^A)+1...r(p^A)-1]$ and $B[l(p^B)+1...r(p^B)-1]$ without any restrictions. In this case, define two pseudo base pairs $\bar{p}^A = (l(p^A)-1, r(p^A)+1)$ and $\bar{p}^B = (l(p^B)-1, r(p^B)+1)$, then $M[\bar{p}^A, \bar{p}^B]$ can also be decomposed as the sum of two parts: $S_{str}(\bar{p}^A, \bar{p}^B)$, and the

optimal alignment score between the regions $A[l(p^A)...r(p^A)]$ and $B[l(p^B)...r(p^B)]$ without any restrictions. Note that $\bar{p}^A$ and $\bar{p}^B$ are both pseudo base pairs, and thus based on the definition, we have $S_{str}(\bar{p}^A, \bar{p}^B) = 0$. Therefore, $M[\bar{p}^A, \bar{p}^B]$ is exactly the overall alignment score we need to detect the OPMs.

In this case, based on Lemma 1, if $M[p^A, p^B] \geq M[\bar{p}^A, \bar{p}^B]$, we will consider the matching of p^A and p^B as an OPM, and add the OPM $o^{A,B}$ to the OPM list $\mathcal{O}$. The overhead for detecting the OPM is that we need to double the computation for each combination of p^A and p^B. However, such overhead will not raise the time complexity, and it is worthy as it will lead to a more significant speedup of the algorithm. In the following section, we will devise a new algorithm by assuming that the OPM list $\mathcal{O}$ is available.

6.2.5 A New Algorithm with Cubic Time Complexity

In this section, we introduce a new SAF-style RNA structure alignment algorithm, which improves the original `RNAscf` algorithm based on Lemma 1 and has a time complexity of $O(z(n^2 + l^2))$. Here, z is the size of the OPM list $\mathcal{O}$, and we expect that $n < z \ll n^2$ when assuming *polymer-zeta* property [141]. If we also assume $O(n) = O(l)$ (with fixed input RNA structures or efficiently pruned RNA structure ensembles), the overall time complexity of the new algorithm becomes $O(zl^2)$.

The new algorithm is developed based on the `RNAscf` algorithm [10]. Therefore, we adopt the same definition and notation as introduced in Section 2.1, as well as the similar recursive functions style used in Section 2.2. Because the computations of $M[p^A, p^B]$ and $M_h[p^A, p^B]$

are boundary cases for the algorithm and are directly computed without referring to previous alignment results, the recursive functions for computing them are exactly the same as in the original algorithm:

$$M[p^A, p^B] = \max \begin{cases} M_h[p^A, p^B], \\ M_l[p^A, p^B]. \end{cases} \tag{6.6}$$

$$M_h[p^A, p^B] = S_{str}(p^A, p^B) + S_{seq}(L(p^A), L(p^B)). \tag{6.7}$$

The computation of $M_l[p^A, p^B]$, on the other hand, refers to the previous alignment results that assumes p_i^A matches $p_{i'}^B$ (see Equation 6.4). Using Lemma 1, it is clear to see that instead of traversing all combinations of p_i^A and $p_{i'}^B$, we only need to consider the cases when the matching of p_i^A and $p_{i'}^B$ is an OPM:

$$M_l[p^A, p^B] = S_{str}(p^A, p^B) + \max_{o_{i,i'}^{A,B} \in \mathcal{O}} \left\{ M_c[p_i^A, p_{i'}^B] + S_{seq}(L_r(p_i^A, p^A), L_r(p_{i'}^B, p^B)) \right\}. \tag{6.8}$$

Similarly, for the computation of $M_c[p_i^A, p_{i'}^B]$, we need to refer to the scenarios where p_i^A matches $p_{i'}^B$ and p_j^A matches $p_{j'}^B$. The matching of p_i^A and $p_{i'}^B$ is guaranteed to be an OPM, as ensured by Equation 6.8. Therefore, we only need to modify Equation 6.5 to ensure that the matching of p_j^A and $p_{j'}^B$ is an OPM:

$$M_c[p_i^A, p_{i'}^B] = \max_{o_{j,j'}^{A,B} \in \mathcal{F}(o_{i,i'}^{A,B})} \begin{cases} M[p_i^A, p_{i'}^B] + S_{seq}(L_l(p_i^A, p^A), L_l(p_{i'}^B, p^B)), \\ M_c[p_j^A, p_{j'}^B] + M[p_i^A, p_{i'}^B] + S_{seq}(L(p_j^A, p_i^A), L(p_{j'}^B, p_{i'}^B)), \\ M_c[p_j^A, p_{j'}^B] + S_{seq}(L(p_j^A, p_i^A), L(p_{j'}^B, p_{i'}^B)) + S_{seq}(L(p_i^A), L(p_{i'}^B)). \end{cases} \tag{6.9}$$

Here, the set $\mathcal{F}(o_{i,i'}^{A,B})$ contains all OPMs that are directly before the OPM $o_{i,i'}^{A,B}$. The $\mathcal{F}$ set regarding the OPMs is defined as the follows. If an OPM $o_{j,j'}^{A,B} \in \mathcal{F}(o_{i,i'}^{A,B})$, then either $p_j^A \in \mathcal{F}(p_i^A)$ or $p_{j'}^B \in \mathcal{F}(p_{i'}^B)$.

Recall that the time complexity of the original algorithm is $O(n^4 + n^2l^2)$. The first term $O(n^4)$ results from $O(n^2)$ computations by traversing all combinations of p^A and p^B (see Equation 6.2) and $O(n^2)$ time for computing M_l (see Equation 6.4). In the new algorithm, we introduce the OPM constraint to Equation 6.8 and Equation 6.9, and thus reduce the time complexity for computing M_l from $O(n^2)$ to $O(z)$. In this case, the first term $O(n^4)$ of the original time complexity can be reduced to $O(zn^2)$.

The second term $O(n^2l^2)$ in the original time complexity results from computing the sequence similarities between all loop regions. Note that all loop similarities required for computing M_l (Equation 6.8) and M_c (Equation 6.9) are associated with OPMs. For example, in Equation 6.8, all the loops are defined according to p_i^A and $p_{i'}^A$, whose matching is expected to be an OPM. And in Equation 6.9, all the loops are defined according to p_i^A and $p_{i'}^A$, as well as p_j^A and $p_{j'}^B$, where both of these matchings are assumed to be OPMs. In this case, we do not need to compute loop similarities for all $O(n^2)$ base-pair combinations, instead we only need to compute the loop similarities that are associated with the OPMs. In this case, the time complexity for computing the sequence similarities between all loops that are required by the computation of M_l and M_c can be finished in $O(zl^2)$.

The only exception for the sequence similarity computation is the hairpin loop similarity $S_{seq}(L(p^A), L(p^B))$, which is required for computing M_h (Equation 6.7). The computation of M_h is not constrained by the OPM list, and therefore $O(n^2l^2)$ time is still required. To resolve this issue, we observe that most RNA structure alignment algorithms empha-

size the structure similarity other than sequence similarity ($w_1 > w_2$ in Equation 6.1). In this case, if there exist some base pairs within the regions enclosed by p^A and p^B to be matched, we can expect that $M_l[p^A, p^B] > M_h[p^A, p^B]$ in Equation 6.6. In this case, to avoid the unnecessary computation of $M_h[p^A, p^B]$, we can derive an upper bound $\hat{M}_h[p^A, p^B]$, which satisfies $\hat{M}_h[p^A, p^B] > M_h[p^A, p^B]$ and can be estimated in unit time. Note that if $M_l[p^A, p^B] > \hat{M}_h[p^A, p^B]$, we are sure that $M_l[p^A, p^B] > M_h[p^A, p^B]$ by transition, and thus can save the computation of $M_h[p^A, p^B]$. The upper bound $\hat{M}_h[p^A, p^B]$ can be easily derived by assuming maximum number of nucleotide matchings and minimum number of gaps:

$$\hat{M}_h[p^A, p^B] = S_{str}(p^A, p^B) + \min(|L(p^A)|, |L(p^B)|) * d_{max} + I * g + (||L(p^A)| - |L(p^B)||) * e, \tag{6.10}$$

where d_{max} is the highest score in the 4-by-4 nucleotide substitution matrix D, and I is a boolean variable that is set to 1 if $|L(p^A)| \neq |L(p^B)|$ and set to 0 otherwise. For the computation of each $M[p^A, p^B]$, we first estimate the upper bound $\hat{M}_h[p^A, p^B]$ in a unit time, and then compute $M_l[p^A, p^B]$ in $O(z)$ time. By comparing these two values, we will determine whether the computation of $M_h[p^A, p^B]$ is necessary. The computation of $M_h[p^A, p^B]$ is only necessary when there are only a few base pair enclosed by p^A and p^B to be matched. Such condition implies the scenarios that either p^A or p^B is a real hairpin loop in the RNA structures, whose number is bounded by $O(n)$. Overall, the hairpin loop similarity matrix M_h can be computed in $O(nl^2)$ time, and the overall time complexity of this algorithm is thus $O(z(n^2 + l^2))$.

6.2.6 Online Pruning of Optimal Pair Matchings

In the previous sections, we have presented our approaches for detecting OPMs and building an OPM list $\mathcal{O}$, as well as a more efficient algorithm that is developed based on $\mathcal{O}$. Time complexity analysis of the algorithm claims that $O(z(n^2 + l^2))$ time is sufficient for this new algorithm. The size of the OPM list $\mathcal{O}$, i.e. z, thus becomes an important factor that determines the efficiency of the novel algorithm. Under the current algorithmic setup, as well as other similar works that implement a candidate list [8, 141], z continuously grows as the algorithm proceeds. In this case, it is desirable to devise an online pruning technique, which can remove the obsolete OPMs from $\mathcal{O}$, and thus achieve further speedup of the algorithm.

In this section, we will present such an online pruning technique to reduce the size of the OPM list $\mathcal{O}$. The intuition of this online pruning technique comes from the following observation. The RNA structures are primarily stabilized by a number of helices, or *perfectly stacked* base pairs. If p_j^A is perfectly stacked on p_i^A, then $l(p_j^A) = l(p_i^A) - 1$, and $r(p_j^A) = r(p_i^A) + 1$. Consider the alignment between two helices, where each one of them contains m perfectly stacked base pairs. Assume that the first helix contains base pairs $p_i^A, p_{i+1}^A, ..., p_{i+m}^A$, and the second helix contains base pairs $p_{i'}^B, p_{i'+1}^B, ..., p_{i'+m}^B$. Based on Lemma 1, there will be at least m OPMs detected from such alignment, i.e. $o_{i,i'}^{A,B}, o_{i+1,i'+1}^{A,B}, ..., o_{i+m,i'+m}^{A,B}$. Apparently, maintaining all these m OPMs is unnecessary, as these base pairs should be aligned together as two complete helices, rather than be aligned separately as two sets of individual base pairs. In this case, maintaining only one OPM, i.e. $o_{i+m,i'+m}^{A,B}$, is sufficient to represent such an alignment. The other $m - 1$ OPMs become obsolete as soon as the OPM $o_{i+m,i'+m}^{A,B}$ is detected, and can be removed from the OPM list $\mathcal{O}$ to improve computational efficiency. In the following paragraphs, we will extend this idea to consider all situations in addition to the

perfectly stacked scenario, as well as give formal description of this technique and related proofs.

We will demonstrate the major idea of our novel online OPM pruning technique using Figure 6.2 (b). Imagine that at the current stage, $M[p^A, p^B]$ has just been computed and $o^{A,B}$ has been identified as an OPM, where $o^{A,B}_{\chi,\chi'}$ is an arbitrary OPM that has been previously identified and is enclosed by $o^{A,B}$ ($p^A_\chi <_I p^A$ and $p^B_{\chi'} <_I p^B$). Our aim is to estimate whether the detection of the OPM $o^{A,B}$ will make $o^{A,B}_{\chi,\chi'}$ obsolete. Let p^A_* and $p^B_{*'}$ be arbitrary base pairs such that $p^A <_I p^A_*$ and $p^B <_I p^B_{*'}$. The regions enclosed by p^A_* and $p^B_{*'}$ can be partitioned using at least one of the following ways: $M[p^A_\alpha, p^B_{\alpha'}] + M[p^A, p^B] + M[p^A_\epsilon, p^B_{\epsilon'}]$ (which is indicated by dark gray in Figure 6.2 (b)) and $M[p^A_\lambda, p^B_{\lambda'}] + M[p^A_\chi, p^B_{\chi'}] + M[p^A_\theta, p^B_{\theta'}]$ (which is indicated by light gray in Figure 6.2 (b)). If the corresponding score for the first path is higher than the second, $M[p^A_\chi, p^B_{\chi'}]$ will not be referred to by any future matching between arbitrary base pairs p^A_* and $p^B_{*'}$, and thus making the OPM $o^{A,B}_{\chi,\chi'}$obsolete. In this case, the OPM $o^{A,B}_{\chi,\chi'}$ can be removed from $\mathcal{O}$.

We can summarize the criterion for removing $o^{A,B}_{\chi,\chi'}$ as an obsolete OPM using the following inequality:

$$M[p^A_\alpha, p^B_{\alpha'}] + M[p^A, p^B] + M[p^A_\epsilon, p^B_{\epsilon'}] \geq M[p^A_\lambda, p^B_{\lambda'}] + M[p^A_\chi, p^B_{\chi'}] + M[p^A_\theta, p^B_{\theta'}],$$

which can be rewritten as:

$$M[p^A, p^B] - M[p^A_\chi, p^B_{\chi'}] \geq (M[p^A_\lambda, p^B_{\lambda'}] - M[p^A_\alpha, p^B_{\alpha'}]) + (M[p^A_\theta, p^B_{\theta'}] - M[p^A_\epsilon, p^B_{\epsilon'}]).$$

To utilize such criterion, we need to have access to all values included in the above inequality. However, we only know the values at the left hand side of the inequality ($M[p^A, p^B]$ and $M[p^A_\chi, p^B_{\chi'}]$), while the other values at the right hand side are unknown. This is because the definitions of these pseudo base pairs are determined by p^A_* and $p^B_{*'}$, which are arbitrary base pairs that have not yet been computed by the DP algorithm. To solve this issue, observe that the score $M[p^A_\lambda, p^B_{\lambda'}] - M[p^A_\alpha, p^B_{\alpha'}]$ is strongly related to the regions $A[l(p^A_\beta)...r(p^A_\beta)]$ and $B[l(p^B_{\beta'})...r(p^B_{\beta'})]$, and $M[p^A_\theta, p^B_{\theta'}] - M[p^A_\epsilon, p^B_{\epsilon'}]$ is strongly related to the regions $A[l(p^A_\delta)...r(p^A_\delta)]$ and $B[l(p^B_{\delta'})...r(p^B_{\delta'})]$. Note that the regions $A[l(p^A_\beta)...r(p^A_\beta)]$ and $A[l(p^A_\delta)...r(p^A_\delta)]$ can be determined when p^A and p^A_χ are known, which makes the estimation of their impact on future alignments possible (similarly for the regions $B[l(p^B_{\beta'})...r(p^B_{\beta'})]$ and $B[l(p^B_{\delta'})...r(p^B_{\delta'})]$). In this case, we can develop two upper bounds $\hat{U}_\beta$ and $\hat{U}_\delta$, such that:

$$\hat{U}_\beta \geq M[p^A_\lambda, p^B_{\lambda'}] - M[p^A_\alpha, p^B_{\alpha'}],$$
$$\hat{U}_\delta \geq M[p^A_\theta, p^B_{\theta'}] - M[p^A_\epsilon, p^B_{\epsilon'}].$$

In this case, if $M[p^A, p^B] - M[p^A_\chi, p^B_{\chi'}] \geq \hat{U}_\beta + \hat{U}_\delta$, we are sure that the criterion for characterizing $o^{A,B}_{\chi,\chi'}$ as an obsolete OPM will be satisfied, and we will be able to remove $o^{A,B}_{\chi,\chi'}$ from $\mathcal{O}$ immediately.

Now, we can discuss the details for setting up the upper bounds $\hat{U}_\beta$ and $\hat{U}_\delta$. Because $\hat{U}_\beta$ and $\hat{U}_\delta$ are defined symmetrically, we only discuss the computation of $\hat{U}_\beta$. Note that the upper bound $\hat{U}_\beta$ needs to satisfy the condition $\hat{U}_\beta \geq M[p^A_\lambda, p^B_{\lambda'}] - M[p^A_\alpha, p^B_{\alpha'}]$. Clearly, the difference between $M[p^A_\lambda, p^B_{\lambda'}] - M[p^A_\alpha, p^B_{\alpha'}]$ directly comes from concatenating the region $A[l(p^A_\beta)...r(p^A_\beta)]$ to the region $A[l(p^A_\alpha)...r(p^A_\alpha)]$, as well as concatenating the region $B[l(p^B_{\beta'})...r(p^B_{\beta'})]$ to the region $B[l(p^B_{\alpha'})...r(p^B_{\alpha'})]$. The best case scenario for such an operation, is to assume that the

concatenation of the regions $A[l(p_\beta^A)...r(p_\beta^A)]$ and $B[l(p_{\beta'}^B)...r(p_{\beta'}^B)]$ will result in as many new base-pair and nucleotide matches as possible.

Assume that there are m_β^A base pairs that are annotated in the region $A[l(p_\beta^A)...r(p_\beta^A)]$, and $m_{\beta'}^B$ base pairs that are annotated in the region $B[l(p_{\beta'}^B)...r(p_{\beta'}^B)]$. Also assume the maximum base-pair substitution score in the RIBOSUM matrix R is r_{max}. By concatenating the regions $A[l(p_\beta^A)...r(p_\beta^A)]$ and $B[l(p_{\beta'}^B)...r(p_{\beta'}^B)]$, we introduce at most $\max(m_\beta^A, m_{\beta'}^B)$ more base-pair matchings to the alignment indicated by $M[p_\alpha^A, p_{\alpha'}^B]$. This implies the maximum structure alignment score increment of $\max(m_\beta^A, m_{\beta'}^B) * r_{max}$. Similarly, at most $\max(|L(p_\beta^A)|, |L(p_{\beta'}^B|))$ more nucleotide matches, or gap fill-ups, are possible, compared to the existing alignment indicated by the score $M[p_\alpha^A, p_{\alpha'}^B]$. The corresponding alignment score for such case is thus: $\max(|L(p_\beta^A)|, |L(p_{\beta'}^B|)) * (d_{max} - g - e)$. To explicitly represent the upper bound using only the identified OPMs, we rename $\hat{U}_\beta$ as $\hat{U}_l[o_{\chi,\chi'}^{A,B}, o^{A,B}]$ (similarly, we rename $\hat{U}_\delta$ as $\hat{U}_r[o_{\chi,\chi'}^{A,B}, o^{A,B}]$). Therefore, $\hat{U}_l[o_{\chi,\chi'}^{A,B}, o^{A,B}]$ and $\hat{U}_r[o_{\chi,\chi'}^{A,B}, o^{A,B}]$ can be computed using the following equations:

$$\begin{aligned}
\hat{U}_l[o_{\chi,\chi'}^{A,B}, o^{A,B}] &= \max(m_\beta^A, m_{\beta'}^B) * r_{max} + \max(|L(p_\beta^A)|, |L(p_{\beta'}^B)|) * (d_{max} - g - e), \\
\hat{U}_r[o_{\chi,\chi'}^{A,B}, o^{A,B}] &= \max(m_\delta^A, m_{\delta'}^B) * r_{max} + \max(|L(p_\delta^A)|, |L(p_{\delta'}^B)|) * (d_{max} - g - e).
\end{aligned} \tag{6.11}$$

With the upper bounds $\hat{U}_l[o_{\chi,\chi'}^{A,B}, o^{A,B}]$ and $\hat{U}_r[o_{\chi,\chi'}^{A,B}, o^{A,B}]$, we are able to formally prove the correctness of the online OPM pruning technique:

Lemma 2: If $M[p^A, p^B] - M[p_\chi^A, p_{\chi'}^B] \geq \hat{U}_l[o_{\chi,\chi'}^{A,B}, o^{A,B}] + \hat{U}_r[o_{\chi,\chi'}^{A,B}, o^{A,B}]$, where $\hat{U}_l[o_{\chi,\chi'}^{A,B}, o^{A,B}] \geq M[p_\lambda^A, p_{\lambda'}^B] - M[p_\alpha^A, p_{\alpha'}^B]$ and $\hat{U}_r[o_{\chi,\chi'}^{A,B}, o^{A,B}] \geq M[p_\theta^A, p_{\theta'}^B] - M[p_\epsilon^A, p_{\epsilon'}^B]$, then $M[p^A, p^B] + M[p_\alpha^A, p_{\alpha'}^B] + M[p_\epsilon^A, p_{\epsilon'}^B] \geq M[p_\chi^A, p_{\chi'}^B] + M[p_\lambda^A, p_{\lambda'}^B] + M[p_\theta^A, p_{\theta'}^B]$.

Proof:

$$
\begin{aligned}
&M[p^A, p^B] \geq M[p^A_{\chi}, p^B_{\chi'}] + \hat{U}_l[o^{A,B}_{\chi,\chi'}, o^{A,B}] + \hat{U}_r[o^{A,B}_{\chi,\chi'}, o^{A,B}] \\
&\Rightarrow M[p^A, p^B] + M[p^A_{\alpha}, p^B_{\alpha'}] + M[p^A_{\epsilon}, p^B_{\epsilon'}] \\
&\geq M[p^A_{\chi}, p^B_{\chi'}] + \hat{U}_l[o^{A,B}_{\chi,\chi'}, o^{A,B}] + \hat{U}_r[o^{A,B}_{\chi,\chi'}, o^{A,B}] + M[p^A_{\alpha}, p^B_{\alpha'}] + M[p^A_{\epsilon}, p^B_{\epsilon'}] \\
&\Rightarrow M[p^A, p^B] + M[p^A_{\alpha}, p^B_{\alpha'}] + M[p^A_{\epsilon}, p^B_{\epsilon'}] \geq M[p^A_{\chi}, p^B_{\chi'}] + M[p^A_{\lambda}, p^B_{\lambda'}] + M[p^A_{\theta}, p^B_{\theta'}].
\end{aligned}
$$

■

As a result, when the condition given in Lemma 2 is satisfied, the enclosed OPM $o^{A,B}_{\chi,\chi'}$ can be readily removed.

6.2.7 Pseudo-code

The pseudo-code for the new RNA secondary structure alignment algorithm that implements both speedup techniques is summarized in Figure 6.3.

Algorithm 1 Pseudo-code for the new $O(zl^2)$ algorithm

Order base pairs in A by their ending nucleotides; Order base pairs in B by their enc
cleotides; Initialize the OPM list $\mathcal{O} \leftarrow \emptyset$;
for $i = 1$ to $|\mathcal{P}^A|$ **do**
  $p^A \leftarrow i$th base pair in A
  for $j = 1$ to $|\mathcal{P}^B|$ **do**
    $p^B \leftarrow j$th base pair in B; Compute $M_l[p^A, p^B]$;
    Estimate $\hat{M}_h[p^A, p^B]$ with Equation 10, $M_h[p^A, p^B] \leftarrow \hat{M}_h[p^A, p^B]$;
    if $\hat{M}_h[p^A, p^B] \geq M_l[p^A, p^B]$ **then**
      Compute $M_h[p^A, p^B]$;
    end if
    $M[p^A, p^B] \leftarrow max(M_l[p^A, p^B], M_h[p^A, p^B])$; Compute $M[\bar{p}^A, \bar{p}^B]$;
    if $M[p^A, p^B] \geq M[\bar{p}^A, \bar{p}^B]$ **then**
      Identify the matching of p^A and p^B as an OPM $o^{A,B}$;
      for each OPM $o^{A,B}_{k,k'} \in \mathcal{O}$ **do**
        Estimate $\hat{U}_l[o^{A,B}_{k,k'}, o^{A,B}]$ and $\hat{U}_r[o^{A,B}_{k,k'}, o^{A,B}]$ with Equation 11;
        if $M[p^A, p^B] \geq \hat{U}_l[o^{A,B}_{k,k'}, o^{A,B}] + \hat{U}_r[o^{A,B}_{k,k'}, o^{A,B}] + M[p^A_k, p^B_{k'}]$ **and** There exists
        pair between $o^{A,B}_{k,k'}$ and $o^{A,B}$ **then**
          Remove $o^{A,B}_{k,k'}$ from the OPM list $\mathcal{O}$;
        end if
      end for
      Add $o^{A,B}$ to the OPM list $\mathcal{O}$;
    end if
  end for
end for

Figure 6.3: Pseudo-code for the implementation of the speedup techniques.

6.3 Results

We implemented the proposed SAF-style RNA structural alignment algorithm into a program called `ERA` (Efficient RNA Alignment) using GNU C++. In this section, we will show that (1) `ERA` has the expected $O(zl^2)$ time complexity; (2) `ERA` is as accurate as the other state-of-the-art RNA alignment tools; and (3) `ERA` runs much faster than the other RNA alignment

tools. In addition to these goals, we have also benchmarked `ERA` to demonstrate its $O(l^2)$ space complexity.

We benchmarked the `ERA` with two other state-of-the-art RNA alignment tools: `LocARNA` as a representative of the SAF-style RNA structure alignment algorithms and `RNAforester` as a representative of the tree-based RNA structure alignment algorithms. Note that although `LocARNA` is developed to compare RNA structure ensembles, its flexible parameter setup makes it easy to prune its input RNA ensembles (see Section 3.1 for more details). We do not compare `ERA` with its predecessor `RNAscf`, because `RNAscf` is implemented to find consensus helical configurations that do not include individual base pairs [10]. Both `LocARNA` and `RNAforester` were invoked using their default parameters.

6.3.1 Running `LocARNA`

Note that `LocARNA` was originally developed to compare two RNA structure ensembles [142]. Due to the recent technical advances in experimental RNA structure probing, we anticipate that RNA structures can be predicted with much higher accuracy. Therefore, we develop `ERA` to compare two fixed RNA structures. In this case, we need to prune the original inputs of `LocARNA`, so as to ensure that they only represent the fixed structures rather than any additional information.

The input RNA ensembles for `LocARNA` are represented using the base-pairing probability matrices, which can be computed using the McCaskill's algorithm [66, 93]. In a base-pairing probability matrix, each base pair (possibly crossing) is assigned with a probability to indi-

cate its thermodynamic stability. Our goal is to prune such a base-pair probability matrix, such that it only contains information regarding the fixed RNA structure (in our experiment, we take the Rfam [57] annotation or the BraliBase II [52] annotation as the fixed structure for an RNA sequence). For each base pair in the matrix, if it is not presented in the annotated structure, its corresponding probability is reset to 0. On the other hand, if it is included in the annotated structure, its probability is reset to 1. In this case, the pruned base-pairing probability matrix contains only the information regarding the fixed RNA structure. All `LocARNA` inputs for experiments mentioned in this chapter are preprocessed using this strategy.

6.3.2 Time Complexity

In this section, we expect to show that the proposed sparsification is successfully implemented, and `ERA` has the expected $O(zl^2)$ time complexity. To show the $O(zl^2)$ time complexity, we chose a number of RNA families from Rfam that have a wide range of sequence lengths. We then randomly selected two individual RNA structures from each family to run `ERA` alignment. The running time for their alignments, versus n^3 (note that $n < l$ for annotated structures and $O(n) = O(l)$), is plotted in Figure 6.4 (a). We can clearly observe the expected $O(zl^2)$ time complexity from the figure. In addition, we are also able to show that the speedup ratio, when comparing to the $O(l^4 + n^2l^2)$ `LocARNA` algorithm, is strongly correlated with the efficiency of pair matching reduction due to the sparse DP technique (the ratio n^2/z, see Figure 6.4 (b)). The relatively large deviations are observed for biocoid_3UTR and snR86 RNA structures. This is because they contain a large number of base pairs and have a high base pair to sequence length ratio. In this case, the overhead for maintaining the

OPM list becomes apparent and makes the speedup less significant. In summary, we have shown that the sparse DP technique is successfully implemented, `ERA` has an expected time complexity of $O(zl^2)$.

6.3.3 Alignment Quality

In addition to time complexity improvement, we also expect to show that `ERA` is as accurate as the other state-of-the-art SAF-style RNA structure alignment tools. We used BraliBase II [52] as the reference data set, and used its corresponding structure annotations as the fixed input structures. We adopted two measures to indicate the alignment quality, i.e., the Sum-of-Pair Score (SPS) [52] and the Structure Conservation Index (SCI) [137]. The benchmark results are shown in Figure 6.5. The alignment qualities of `ERA` and `LocARNA` are nearly identical, since incorporating the sparse DP technique will not compromise global optimality. The benchmark results also show that `ERA` and `LocARNA` can produce more accurate alignments when compared to `RNAforester`. This is because `ERA` and `LocARNA` are both SAF-style RNA alignment algorithms that are capable of flexibly handling incorrectly predicted base-pairs, while `RNAforester` is a tree-based RNA alignment algorithm that is sensitive to such errors.

Table 6.1: Comparison on running time of ERA, LocARNA, and RNAforester

RNA family	length (bp)	num. pairs	ERA (sec)	LocARNA (sec)	ERA vs. LocARNA (fold)	RNAforester (sec)	ERA vs. RNAforester (fold)
tRNA	78	21	0.017	0.100	5.882	0.047	2.765
Gly riboswitch	105	22	0.015	0.277	18.46	0.162	10.80
U12 spliceosome	160	42	0.035	0.311	8.886	0.657	18.77
Phage_pRNA	244	43	0.124	0.647	5.218	6.935	55.93
tmRNA	367	64	0.929	22.45	24.16	225.4	242.6
biocoid_3UTR	549	155	4.898	170.3	34.77	13.99	2.856
snR86	1004	333	53.15	4862	91.48	5.579	-9.527*
Sacc_telomerase	1162	181	23.93	522.3	21.82	3697	154.5

ERA is slower than RNAforester when aligning snR86 RNA structures.

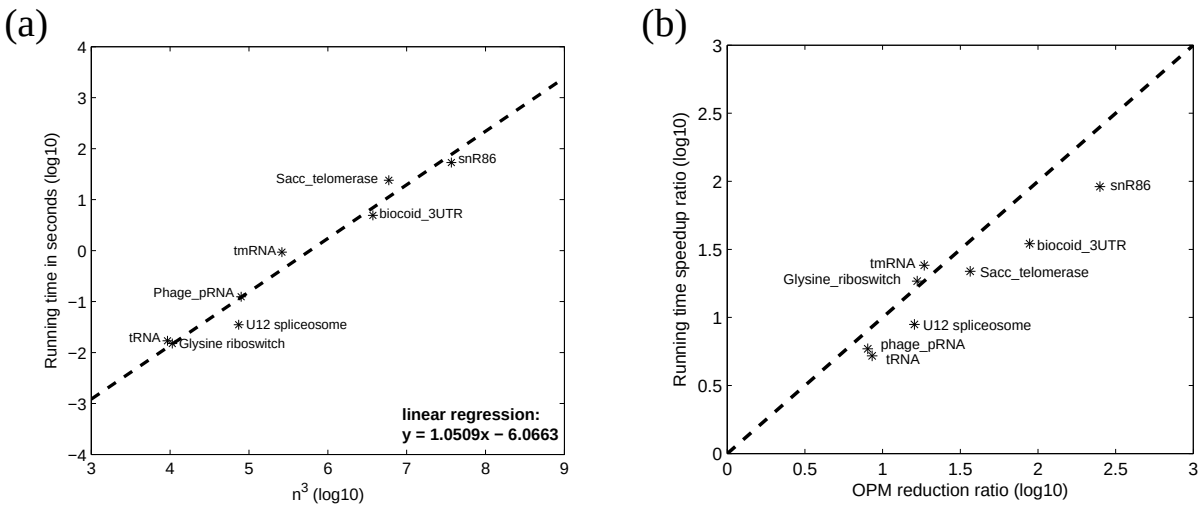

Figure 6.4: Time complexity and OPM reduction of `ERA`. (a) Running time versus n^3, where n is the average number of base pairs in the RNA structures. (b) OPM reduction ratio versus running time speedup ratio. The OPM reduction ratio is computed by n^2/z, where z is the number of OPMs.

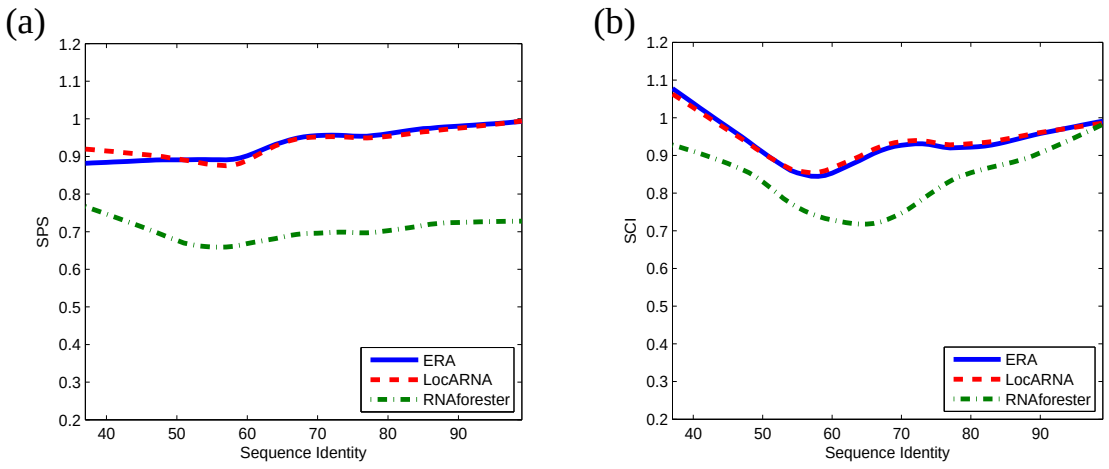

Figure 6.5: Alignment quality comparison of `ERA`, `LocARNA` and `RNAforester`. The comparison of (a) Sum-of-Pair Score and (b) Structure Conservation Index between `ERA`, `LocARNA` and `RNAforester` on BraliBase II data set. The sequence identity range is between 0.37 to 0.99. The curves are generated using LOWESS smoothing with a smoothing factor of 0.3.

6.3.4 Running Time Speedup

Finally, after benchmarking the time complexity and alignment accuracy of ERA, we also expect to show that ERA is an efficient implementation and can run faster than other state-of-the-art RNA alignment tools. We compared the real running time of ERA, LocARNA, and RNAforester on the selected RNA structures from Rfam. The benchmark results are summarized in Table 6.1. We can observe that ERA is capable of speeding up LocARNA by a minimum of 5.2 fold and a maximum of 91.5 fold. ERA can also speedup RNAforester by a minimum of 2.8 fold and a maximum of 242.6 fold, with only one exception in which RNAforester is 9.6 times faster than ERA. This is because the RNA structures being aligned (snR86) contain only one stem-loop structure; and in such a special case, the time complexity of RNAforester becomes $O(l^2)$ [63].

To further investigate the real running time speedup of ERA on randomly selected RNA structures, we compiled a much larger data set that contains 1,000 pairs of randomly selected RNA structures from Rfam. The benchmark results on this large data set are summarized in Figure 6.6. In Figure 6.6, we can see that ERA (blue triangle) runs much faster than LocARNA (red cross) and RNAforester (green star). In addition, we can also observe that the running time of ERA grows slower than those of LocARNA and RNAforester, which further confirms our previous time complexity analysis (see Figure 6.4 (a)). This speedup is significant, and renders ERA with the power of aligning long ncRNAs that are revealed by recent research advances. In summary, ERA is an efficient and accurate RNA structure alignment tool as compared to its state-of-the-art counterparts LocARNA and RNAforester.

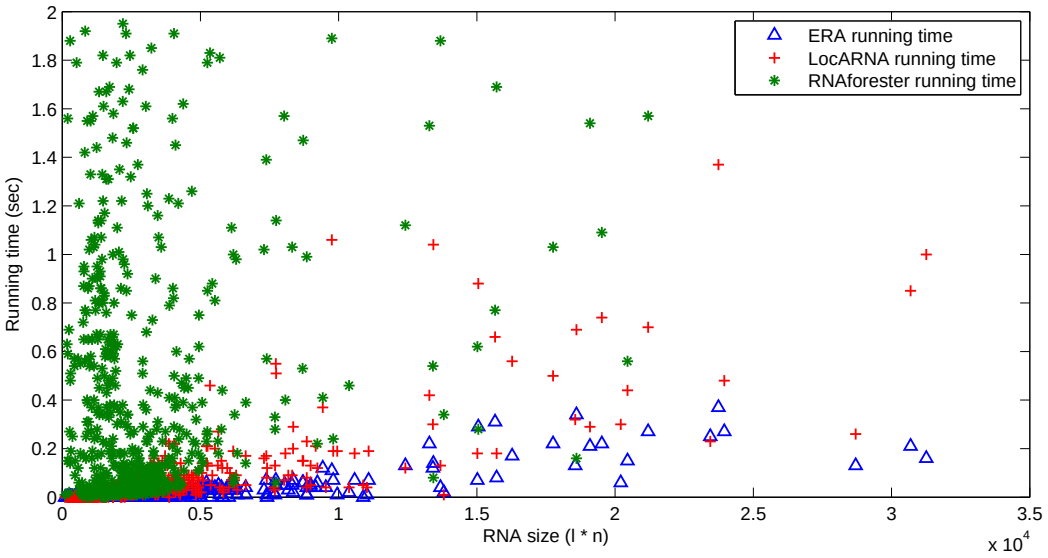

Figure 6.6: Computational efficiency comparison between `ERA`, `LocARNA` and `RNAforester` on aligning randomly selected RNA structures from Rfam. The running time for `ERA` (blue triangles), `LocARNA` (red crosses) and `RNAforester` (green stars) on aligning 1,000 pairs of randomly selected RNA structures from the Rfam database. The x-axis corresponds to the average sizes of the RNA structures being aligned, which is computed as the product of their average length (l) and their average number of base pairs (n). The y-axis corresponds to the actual running time in the unit of second. We can see that `ERA` is significantly faster than the other two tools.

6.4 Conclusions and Discussion

In this chapter, we have presented a novel algorithm for efficient alignment of RNA secondary structures by incorporating the sparse DP technique. The major theoretical contribution of this work lies in two parts. First, to our knowledge, this is the first application of the sparse DP technique to RNA structure-structure alignment. Second, the novel online OPM pruning technique can provide insights for future algorithm designs that need to maintain a candidate list. The implementation of this novel algorithm is a tool called `ERA`, which can run in $O(zl^2)$ time and $O(l^2)$. Such time and space complexity make `ERA` one of the most efficient RNA structure alignment tools that are currently available.

The online OPM pruning technique is newly developed from this work, which aims at deleting obsolete candidates as the DP proceeds. Although this technique cannot improve the computational complexity, it is efficient in reducing the real running time. We observed that by incorporating this technique, the running time of `ERA` was reduced by an average of 2.3 fold. Meanwhile, the speedup ratio is highly uniform (with 1.7 fold as the lowest and 3.1 fold as the highest) across RNA structures with different sizes, meaning that it reduces running time by a constant factor. The online OPM pruning technique can also be modified and incorporated into other related algorithms that implement the candidate list, such as the sparse DP algorithms for RNA folding [141], RNA consensus folding [8, 152], and RNA-RNA interaction [109].

With the completion of the ENCODE [18] and modENCODE [25] projects, more and more RNA transcripts will be experimentally revealed. At the same time, with the advance of high-throughput RNA structure probing techniques [72, 85, 134], the secondary structures

of these RNA transcripts will also be predicted with a much higher accuracy. In this case, `ERA`, which can compare fixed RNA structure efficiently and accurately, becomes an ideal computational tool to evaluate the structural similarities of these RNA transcripts. `ERA` can be used to perform all-against-all alignments on these RNA transcripts, which will then be subsequently summarized as the distance matrix for clustering purposes. Various clustering algorithms [142] can then be applied to identify ncRNA families with similar secondary structures and infer their amazing cellular and molecular functionalities.

CHAPTER 7: CONCLUSIONS AND DISCUSSION

7.1 RNA Structural Motif Identification

The importance of ncRNAs has recently been highly appreciated due to the discovery of their amazing cellular functions. The molecular functions of the ncRNAs are usually determined by their specific structures. Therefore, analyzing their structures will provide invaluable insight in understanding their functions. In this dissertation, we have developed a suite of computational methods for the comparative analysis of their secondary structures, with an aim to facilitate the corresponding functional annotations. We begin with the comparison of RNA structural motifs, and end with the genome-wide clustering survey of general ncRNA secondary structures. Our computational methods span a wide-range of scales in RNA structures, and will likely promote and improve the research in related areas.

In Chapter 2 and Chapter 3, we present two computational methods for comparative analysis of RNA structural motifs. The resulting tools, `RNAMotifScan` and `RNAMotifScanX`, have been implemented using C and C++ and benchmarked with the other state-of-the-art motif search tools. Our benchmark results for `RNAMotifScan` indicate that modeling RNA structural motifs using base-pair isostericity is superior to the existing abstraction methods that are based on RNA 3D structure geometries. By incorporating the base-stacking information into the non-canonical base-pairing patterns, `RNAMotifScanX` has made further improvements

over RNAMotifScan in terms of identification accuracy. In addition, a universal p-value cutoff is predicted using RNAMotifScanX to facilitate fully automated motif identification.

RNAMotifScan implements the polynomial-time algorithm framework of RNAscf [10] by incorporating specific features for non-canonical base pairs and structural motifs (as we have discussed in Chapter 2). Since the size of RNA structural motif is usually small, the polynomial-time complexity algorithm is very efficient in aligning RNA structural motif instances. For example, scanning the entire PDB with the largest motif as query (kink-turn motif) using RNAMotifScan takes less than two hours (using single-core configuration). In this case, we design a new algorithm which is more computationally demanding, but can produce much more accurate identification results. For this new algorithm, we model RNA structural motif as a graph, and develop a branch-and-bound algorithm, called RNAMotifScanX, to optimally align the motif graphs with the consideration of both base-pairing and base-stacking information. Benchmarking RNAMotifScanX against RNAMotifScan clearly shows the improvement on accuracy, with RNAMotifScanX producing nearly perfect search results on five (kink-turn, C-loop, sarcin-ricin, reverse kink-turn, and E-loop) important RNA structural motif families. The computational overhead for RNAMotifScanX is moderate and can be handled using current personal computers. For example, searching the largest motif kink-turn against a 50S ribosomal RNA using RNAMotifScanX takes less than 50 mins (using a single-core configuration).

Our experiment of scanning the PDB using RNAMotifScan led to the discovery of many novel motif instances. The RNA structural motif instances identified in this study, even under a very stringent p-value cutoff, significantly outnumber what we have previously known (for example, compared to the motif instances that have been registered in the SCOR [127] database). The prevalence of RNA structural motif instances in RNA structures motivates

us to develop a corresponding database for archiving the discoveries and disseminating the annotations for the registered motif instances. The RNA structural motif database will contain comprehensive annotation of the motif instances like the kink-turn database [115]. Instead of focusing on their 3D structures, we will emphasize their base-pairing and base-stacking interaction patterns. The database, upon its release, will download the new structures deposited to the PDB automatically and update its archive periodically.

By incorporating non-canonical base pairs and the base-stacking interactions, we have proposed a new modeling for RNA structural motifs. We have also developed two computational methods, `RNAMotifScan` and `RNAMotifScanX`, to search for the RNA structural motif instances based on such modeling. While both tools produce highly accurate search results on RNA structures with high resolution, their performance is limited when it comes to low-quality RNA structures. Both of the computational methods assume that the base-pair and base-stacking annotation for the RNA 3D structures are accurate (annotated using `MC-Annotate` [53] of `RNAVIEW` [147]). However, such an assumption is not always true when RNA structures are resolved with limited resolution. The potential annotation errors of `MC-Annotate` and `RNAVIEW` on low-quality RNA structures will be inherited by the motif alignment tools, where incorrect alignments may be produced. In this case, one of our future directions is to incorporate base-pairing probabilities into the representation of RNA structural motif, which is similar to the base-pairing matrix that has been used to represent the RNA secondary structures [65].

Another type of the RNA structural motif identification problem we have revisited in this dissertation is *de novo* RNA structural motif identification. The problem is an important complement to the model-based RNA structural motif identification problem (as we have covered in Chapter 2 and 3 with the search tools `RNAMotifScan` and `RNAMotifScanX`). The *de*

novo motif identification does not presume an explicitly defined query model, but classifies the candidate motif instances based on their mutual structural similarity using a clustering approach. Therefore, it is able to discover novel RNA structural motif families. It may also discover motif instances that do not resemble the query consensus but resemble an individual instance. We have demonstrated the importance of this problem by clustering RNA structural motif elements in ribosomal RNAs (details covered in Chapter 4). In this study, we have discovered two novel RNA structural motif families and many novel instances. These findings have significantly enriched our understanding of the RNA structural motifs, and also suggest that there are more novel instances to be discovered. We conjecture that the novel motif families can be important for protein-binding, and we are seeking collaborators to experimentally verify their specific molecular functions.

The initial candidate motif instances selection step is of great importance to the final performance of the *de novo* identification analysis. It has been observed that RNA structural motifs are usually found in the junctions between the regular A-form helices (internal or bulge loops, and multi-branch loops) or the hairpin loop regions. It is a common practice to fetch candidate RNA structural motifs from these regions [37]. However, incorrect predictions of the base-pairing pattern due to the inadequate RNA 3D structure resolution will obscure the definition of the junctions. To solve this issue, one may use a scanning-window approach to exhaustively generate all possible candidates. The drawback of this approach is that it will generate many overlapping segments and complicates the post-processing step. It is highly recommended that the initial candidate motif instance should be selected with enough attention to ensure well-balanced specificity and sensitivity (or depending on the specific purpose of the analysis).

7.2 Genome-wide Non-coding RNA Classification

In this dissertation, we have also systematically improved a standard clustering pipeline for structural classification of ncRNAs in the genome. In Chapter 5, we present an optimization for the pipeline itself by normalizing the alignment scores and designing an accurate and a fully automatic clustering algorithm. We have normalized the length-biased alignment score using a simulation-based method. After the normalization of the alignment scores, we further apply the CLCL algorithm to extract individual ncRNA clusters from the resulting p-values. Benchmark results of this new pipeline against the traditional hierarchical clustering method clearly show significant improvements for both sensitivity and specificity. This clustering pipeline is also highly automated, which makes the pipeline more robust as compared to the hierarchical clustering approach.

In Chapter 5, we demonstrate the utility of this novel clustering pipeline by clustering the post-transcriptional control elements in the fly 3'-UTR. We have discovered two important clusters of ncRNA elements, where one is responsible for the preferential expression of a cluster of genes in male flies, and regulates the expressions of several genes at the fly septate junction. These discoveries lead to new insights in the functionalities of these ncRNA elements, and have significantly enriched our knowledge about their regulation mechanisms. This genome-wide analysis of ncRNA elements in the fly 3'-UTR points to two conclusions: First, there exist more potential interesting ncRNA families to be discovered in the genome. Second, the functional annotation of these ncRNA families remains difficult, not to mention the prediction of their interactions with other biological molecules. This is because the majority of the genomes are not fully annotated. Experimental verification, in most of the cases, is still the only approach that we can use to confirm the biological discoveries.

As important biological discoveries are made by applying this clustering pipeline to fly 3'-UTR, we also expect to apply this pipeline to other regions of the genome, such as 5'-UTR or even the entire genome. We also propose cross-species clustering, which can provide evolutionary insights for the ncRNAs. However, direct application of this pipeline to large data sets is infeasible, because the all-against-all alignment step is extremely slow. To solve this problem, in Chapter 6, we describe an algorithmic improvement for the RNA secondary structure alignment algorithm. The new alignment algorithm is called `ERA`, which is developed by incorporating the sparse dynamic programming technique. An average of 10 fold speedup in terms of alignment running time is observed when comparing `ERA` with other alignment tools such as `LocARNA` and `RNAforester`. More over, this improvement is made without sacrificing the global optimality of the dynamic programming, and high-quality alignment results are still guaranteed. These advantages make `ERA` an ideal tool to be incorporated in the new clustering pipeline. We expect that with `ERA`, the new clustering pipeline will be more applicable to larger data sets and the clustering of long ncRNAs.

LIST OF REFERENCES

[1] P. L. Adams, M. R. Stahley, M. L. Gill, A. B. Kosek, J. Wang, and S. A. Strobel. Crystal structure of a group I intron splicing intermediate. *RNA*, 10:1867–1887, 2004.

[2] P. L. Adams, M. R. Stahley, A. B. Kosek, J. Wang, and S. A. Strobel. Crystal structure of a self-splicing group I intron with both exons. *Nature*, 430:45–50, 2004.

[3] V. Alesker, R. Nussinov, and H.J. Wolfson. Detection of non-topological motifs in protein structures. *Protein Eng.*, 9:1103–1119, 1996.

[4] S. F. Altschul and B. W. Erickson. Significance of nucleotide sequence alignments: a method for random sequence permutation that preserves dinucleotide and codon usage. *Mol. Biol. Evol.*, 2:526–538, 1985.

[5] A. H. Antonioli, J. C. Cochrane, S. V. Lipchock, and S. A. Strobel. Plasticity of the RNA kink turn structural motif. *RNA*, 16(4):762–768, 2010.

[6] A. Apostolico, G. Ciriello, C. Guerra, C. E. Heitsch, C. Hsiao, and L. D. Williams. Finding 3D motifs in ribosomal RNA structures. *Nucleic Acids Res.*, 37:e29, 2009.

[7] V. J. Auld, R. D. Fetter, K. Broadie, and C. S. Goodman. Gliotactin, a novel transmembrane protein on peripheral glia, is required to form the blood-nerve barrier in Drosophila. *Cell*, 81:757–767, 1995.

[8] R. Backofen, D. Tsur, S. Zakov, and M. Ziv-Ukelson. Sparse RNA folding: Time and space efficient algorithms. *J. of Discrete Algorithms*, 9:12–31, 2011.

[9] V. Bafna, S. Muthukrishnan, and R. Ravi. Computing similarity between RNA strings. In *Proceedings of the 6th Annual Symposium on Combinatorial Pattern Matching*, pages 1–16, Espoo, Finland, 1995. Springer-Verlag, Berlin.

[10] V. Bafna, H. Tang, and S. Zhang. Consensus folding of unaligned RNA sequences revisited. *J. Comput. Biol.*, 13:283–295, 2006.

[11] N. Ban, P. Nissen, J. Hansen, P. B. Moore, and T. A. Steitz. The complete atomic structure of the large ribosomal subunit at 2.4 A resolution. *Science*, 289:905–920, 2000.

[12] A. L. Barabasi and R. Albert. Emergence of scaling in random networks. *Science*, 286:509–512, 1999.

[13] C. Barreau, E. Benson, E. Gudmannsdottir, F. Newton, and H. White-Cooper. Post-meiotic transcription in Drosophila testes. *Development*, 135:1897–1902, 2008.

[14] D. P. Bartel. MicroRNAs: target recognition and regulatory functions. *Cell*, 136:215–233, 2009.

[15] S. Bauer, J. Gagneur, and P. N. Robinson. GOing Bayesian: model-based gene set analysis of genome-scale data. *Nucleic Acids Res.*, 38:3523–3532, 2010.

[16] A. Ben-Dor, R. Shamir, and Z. Yakhini. Clustering gene expression patterns. *J. Comput. Biol.*, 6:281–297, 1999.

[17] H. M. Berman, J. Westbrook, Z. Feng, G. Gilliland, T. N. Bhat, H. Weissig, I. N. Shindyalov, and P. E. Bourne. The Protein Data Bank. *Nucleic Acids Res.*, 28:235–242, 2000.

[18] B. E. Bernstein, E. Birney, and I. Dunham. An integrated encyclopedia of DNA elements in the human genome. *Nature*, 489(7414):57–74, 2012.

[19] F. Besse and A. Ephrussi. Translational control of localized mRNAs: restricting protein synthesis in space and time. *Nat. Rev. Mol. Cell Biol.*, 9:971–980, 2008.

[20] P. N. Borer, B. Dengler, I. Tinoco, and O. C. Uhlenbeck. Stability of ribonucleic acid double-stranded helices. *J. Mol. Biol.*, 86:843–853, 1974.

[21] D. Bouthinon and H. Soldano. A new method to predict the consensus secondary structure of a set of unaligned RNA sequences. *Bioinformatics*, 15(10):785–798, 1999.

[22] J. Šponer, J. Leszczyński, and P. Hobza. Nature of nucleic acid base stacking: Nonempirical ab initio and empirical potential characterization of 10 stacked base dimers. comparison of stacked and h-bonded base pairs. *The Journal of Physical Chemistry*, 100(13):5590–5596, 1996.

[23] C. Bron and J Kerbosch. Finding All Cliques of an Undirected Graph. *Communications of the ACM*, 16:575–579, 1973.

[24] J. J. Cannone, S. Subramanian, M. N. Schnare, J. R. Collett, L. M. D'Souza, Y. Du, B. Feng, N. Lin, L. V. Madabusi, K. M. Muller, N. Pande, Z. Shang, N. Yu, and R. R. Gutell. The comparative RNA web (CRW) site: an online database of comparative sequence and structure information for ribosomal, intron, and other RNAs. *BMC Bioinformatics*, 3:2, 2002.

[25] S. E. Celniker, L. A. Dillon, M. B. Gerstein, K. C. Gunsalus, S. Henikoff, G. H. Karpen, M. Kellis, E. C. Lai, J. D. Lieb, D. M. MacAlpine, G. Micklem, F. Piano, M. Snyder, L. Stein, K. P. White, and R. H. Waterston. Unlocking the secrets of the genome. *Nature*, 459(7249):927–930, 2009.

[26] V. R. Chintapalli, J. Wang, and J. A. Dow. Using FlyAtlas to identify better Drosophila melanogaster models of human disease. *Nat. Genet.*, 39:715–720, 2007.

[27] William M. Clemons, Ditlev E. Brodersen, John P. McCutcheon, Joanna L. C. May, Andrew P. Carter, Robert J. Morgan-Warren, Brian T. Wimberly, and V. Ramakrishnan. Crystal structure of the 30 s ribosomal subunit from thermus thermophilus: purification, crystallization and structure determination. *J. Mol. Biol.*, 310(4):827 – 843, 2001.

[28] C. C. Correll, B. Freeborn, P. B. Moore, and T. A. Steitz. Metals, motifs, and recognition in the crystal structure of a 5S rRNA domain. *Cell*, 91:705–712, 1997.

[29] A. Coventry, D. J. Kleitman, and B. Berger. MSARI: Multiple sequence alignments for statistical detection of RNA secondary structure. *PNAS*, 101(33):12102–12107, 2004.

[30] S. Crucs, S. Chatterjee, and E. R. Gavis. Overlapping but distinct RNA elements control repression and activation of nanos translation. *Mol. Cell*, 5:457–467, 2000.

[31] J. A. Cruz and E. Westhof. Sequence-based identification of 3D structural modules in RNA with RMDetect. *Nat. Methods*, 8:513–519, 2011.

[32] P. Daldrop and D. M. Lilley. The plasticity of a structural motif in RNA: structural polymorphism of a kink turn as a function of its environment. *RNA*, 19(3):357–364, 2013.

[33] A. Dallas and P. B. Moore. The loop E-loop D region of Escherichia coli 5S rRNA: the solution structure reveals an unusual loop that may be important for binding ribosomal proteins. *Structure*, 5:1639–1653, 1997.

[34] E. Davydov and S. Batzoglou. A computational model for RNA multiple structural alignment. *Theoret. Comput. Sci.*, 368:205–216, 2006.

[35] E. D. Demaine, S. Mozes, B. Rossman, and O. Weimann. An optimal decomposition algorithm for tree edit distance. *ACM Trans. Algorithms*, 6(1):2:1–2:19, 2009.

[36] D. di Bernardo, T. Down, and T. Hubbard. ddbRNA: detection of conserved secondary structures in multiple alignments. *Bioinformatics*, 19:1606–1611, 2003.

[37] M. Djelloul and A. Denise. Automated motif extraction and classification in RNA tertiary structures. *RNA*, 14:2489–2497, 2008.

[38] C. B. Do, C. S. Foo, and S. Batzoglou. A max-margin model for efficient simultaneous alignment and folding of RNA sequences. *Bioinformatics*, 24(13):68–76, 2008.

[39] E. A. Doherty, R. T. Batey, B. Masquida, and J. A. Doudna. A universal mode of helix packing in RNA. *Nat. Struct. Biol.*, 8:339–343, 2001.

[40] E. A. Doherty and J. A. Doudna. Ribozyme structures and mechanisms. *Annu Rev Biophys Biomol Struct*, 30:457–475, 2001.

[41] Z. Dominski and W. F. Marzluff. Formation of the 3' end of histone mRNA. *Gene*, 239:1–14, 1999.

[42] O. Dror, R. Nussinov, and H. Wolfson. ARTS: alignment of RNA tertiary structures. *Bioinformatics*, 21 Suppl 2:47–53, 2005.

[43] R. Drysdale. FlyBase : a database for the Drosophila research community. *Methods Mol. Biol.*, 420:45–59, 2008.

[44] C. M. Duarte, L. M. Wadley, and A. M. Pyle. RNA structure comparison, motif search and discovery using a reduced representation of RNA conformational space. *Nucleic Acids Res.*, 31:4755–4761, 2003.

[45] S. R. Eddy. Non-coding RNA genes and the modern RNA world. *Nat. Rev. Genet.*, 2:919–929, 2001.

[46] S. R. Eddy and R. Durbin. RNA sequence analysis using covariance models. 22:2079–2088, 1994.

[47] E. Ennifar, A. Nikulin, S. Tishchenko, A. Serganov, N. Nevskaya, M. Garber, B. Ehresmann, C. Ehresmann, S. Nikonov, and P. Dumas. The crystal structure of UUCG tetraloop. *J. Mol. Biol.*, 304:35–42, 2000.

[48] A. Fasano. Regulation of intercellular tight junctions by zonula occludens toxin and its eukaryotic analogue zonulin. *Ann. N. Y. Acad. Sci.*, 915:214–222, 2000.

[49] F. Ferre, Y. Ponty, W. A. Lorenz, and P. Clote. DIAL: a web server for the pairwise alignment of two RNA three-dimensional structures using nucleotide, dihedral angle and base-pairing similarities. *Nucleic Acids Res.*, 35:W659–668, 2007.

[50] J. A. Firth. Endothelial barriers: from hypothetical pores to membrane proteins. *J. Anat.*, 200(6):541–548, 2002.

[51] D. M. Freymann, R. J. Keenan, R. M. Stroud, and P. Walter. Structure of the conserved GTPase domain of the signal recognition particle. *Nature*, 385(6614):361–364, 1997.

[52] P. P. Gardner, A. Wilm, and S. Washietl. A benchmark of multiple sequence alignment programs upon structural RNAs. *Nucleic Acids Res.*, 33:2433–2439, 2005.

[53] P. Gendron, S. Lemieux, and F. Major. Quantitative analysis of nucleic acid three-dimensional structures. *J. Mol. Biol.*, 308:919–936, 2001.

[54] J. L. Genova and R. G. Fehon. Neuroglian, Gliotactin, and the Na+/K+ ATPase are essential for septate junction function in Drosophila. *J. Cell Biol.*, 161(5):979–989, 2003.

[55] J. Gorodkin, L.J. Heyer, and G.D. Stormo. Finding the most significant common sequence and structure motifs in a set of RNA sequences. *Nucleic Acids Res.*, 25(18):3724–32, 1997.

[56] B. R. Graveley, A. N. Brooks, J. W. Carlson, M. O. Duff, J. M. Landolin, L. Yang, C. G. Artieri, M. J. van Baren, N. Boley, B. W. Booth, J. B. Brown, L. Cherbas, C. A. Davis, A. Dobin, R. Li, W. Lin, J. H. Malone, N. R. Mattiuzzo, D. Miller, D. Sturgill, B. B. Tuch, C. Zaleski, D. Zhang, M. Blanchette, S. Dudoit, B. Eads, R. E. Green, A. Hammonds, L. Jiang, P. Kapranov, L. Langton, N. Perrimon, J. E. Sandler, K. H. Wan, A. Willingham, Y. Zhang, Y. Zou, J. Andrews, P. J. Bickel, S. E. Brenner, M. R. Brent, P. Cherbas, T. R. Gingeras, R. A. Hoskins, T. C. Kaufman, B. Oliver, and S. E. Celniker. The developmental transcriptome of Drosophila melanogaster. *Nature*, 471:473–479, 2011.

[57] S. Griffiths-Jones, A. Bateman, M. Marshall, A. Khanna, and S. R. Eddy. Rfam: an RNA family database. *Nucleic Acids Res.*, 31:439–441, 2003.

[58] A. R. Gruber, S. Findeiss, S. Washietl, I. L. Hofacker, and P. F. Stadler. RNAZ 2.0: improved noncoding RNA detection. *Pac Symp Biocomput*, 15:69–79, 2010.

[59] A. M. Harrison, D. R. South, P. Willett, and P. J. Artymiuk. Representation, searching and discovery of patterns of bases in complex RNA structures. *J. Comput.-Aided Mol. Des.*, 17:537–549, 2003.

[60] T. P. Hausner, J. Atmadja, and K. H. Nierhaus. Evidence that the G2661 region of 23S rRNA is located at the ribosomal binding sites of both elongation factors. *Biochimie*, 69:911–923, 1987.

[61] J. H. Havgaard, R. B. Lyngsø, and J. Gorodkin. The FOLDALIGN web server for pairwise structural RNA alignment and mutual motif search. *Nucleic Acids Res.*, 33:W650–653, 2005.

[62] D. K. Hendrix, S. E. Brenner, and S. R. Holbrook. RNA structural motifs: building blocks of a modular biomolecule. *Q. Rev. Biophys.*, 38:221–243, 2005.

[63] M. Höchsmann, T. Töller, R. Giegerich, and S. Kurtz. Local similarity in RNA secondary structures. *Proc. IEEE Comput. Soc. Bioinform. Conf.*, pages 159–168, 2003.

[64] I. L. Hofacker. Vienna RNA secondary structure server. *Nucleic Acids Res.*, 31(13):3429–3431, 2003.

[65] I. L. Hofacker, S. H. Bernhart, and P. F. Stadler. Alignment of RNA base pairing probability matrices. *Bioinformatics*, 20:2222–2227, 2004.

[66] I. L. Hofacker, W. Fontana, P. F. Stadler, L. S. Bonhoeffer, M. Tacker, and P. Schuster. Fast folding and comparison of RNA secondary structures. *Monatsh. Chem.*, 125:167–188, 1994.

[67] R. P. Jansen. mRNA localization: message on the move. *Nat. Rev. Mol. Cell Biol.*, 2:247–256, 2001.

[68] T. Jiang, G. Lin, B. Ma, and K. Zhang. A general edit distance between RNA structures. *J. Mol. Biol.*, 9:371–388, 2002.

[69] B. Kaczkowski, E. Torarinsson, K. Reiche, J. H. Havgaard, P. F. Stadler, and J. Gorodkin. Structural profiles of human miRNA families from pairwise clustering. *Bioinformatics*, 25:291–294, 2009.

[70] S. Karlin and S. F. Altschul. Methods for assessing the statistical significance of molecular sequence features by using general scoring schemes. *Proc. Natl. Acad. Sci. U.S.A.*, 87:2264–2268, 1990.

[71] S. Karunanithi, J. W. Barclay, I. R. Brown, R. M. Robertson, and H. L. Atwood. Enhancement of presynaptic performance in transgenic Drosophila overexpressing heat shock protein Hsp70. *Synapse*, 44(1):8–14, 2002.

[72] M. Kertesz, Y. Wan, E. Mazor, J. L. Rinn, R. C. Nutter, H. Y. Chang, and E. Segal. Genome-wide measurement of RNA secondary structure in yeast. *Nature*, 467:103–107, 2010.

[73] H. Kiryu, Y. Tabei, T. Kin, and K. Asai. Murlet: a practical multiple alignment tool for structural RNA sequences. *Bioinformatics*, 23(13):1588–1598, 2007.

[74] D. J. Klein, P. B. Moore, and T. A. Steitz. The roles of ribosomal proteins in the structure assembly, and evolution of the large ribosomal subunit. *J. Mol. Biol.*, 340:141–177, 2004.

[75] D. J. Klein, T. M. Schmeing, P. B. Moore, and T. A. Steitz. The kink-turn: a new RNA secondary structure motif. *EMBO J.*, 20:4214–4221, 2001.

[76] R. J. Klein and S. R. Eddy. Rsearch: Finding homologs of single structured rna sequences. *BMC Bioinf.*, 4(1):44, 2003.

[77] E. Lecuyer, H. Yoshida, N. Parthasarathy, C. Alm, T. Babak, T. Cerovina, T. R. Hughes, P. Tomancak, and H. M. Krause. Global analysis of mRNA localization reveals a prominent role in organizing cellular architecture and function. *Cell*, 131:174–187, 2007.

[78] N. B. Leontis, A. Lescoute, and E. Westhof. The building blocks and motifs of RNA architecture. *Curr. Opin. Struct. Biol.*, 16:279–287, 2006.

[79] N. B. Leontis, J. Stombaugh, and E. Westhof. Motif prediction in ribosomal RNAs Lessons and prospects for automated motif prediction in homologous RNA molecules. *Biochimie*, 84:961–973, 2002.

[80] N. B. Leontis, J. Stombaugh, and E. Westhof. The non-Watson-Crick base pairs and their associated isostericity matrices. *Nucleic Acids Res.*, 30:3497–3531, 2002.

[81] N. B. Leontis and E. Westhof. The 5S rRNA loop E: chemical probing and phylogenetic data versus crystal structure. *RNA*, 4:1134–1153, 1998.

[82] N. B. Leontis and E. Westhof. Geometric nomenclature and classification of RNA base pairs. *RNA*, 7:499–512, 2001.

[83] N. B. Leontis and E. Westhof. Analysis of RNA motifs. *Curr. Opin. Struct. Biol.*, 13:300–308, 2003.

[84] A. Lescoute, N. B. Leontis, C. Massire, and E. Westhof. Recurrent structural RNA motifs, Isostericity Matrices and sequence alignments. *Nucleic Acids Res.*, 33:2395–2409, 2005.

[85] J. B. Lucks, S. A. Mortimer, C. Trapnell, S. Luo, S. Aviran, G. P. Schroth, L. Pachter, J. A. Doudna, and A. P. Arkin. Multiplexed RNA structure characterization with selective 2'-hydroxyl acylation analyzed by primer extension sequencing (SHAPE-Seq). *Proc. Natl. Acad. Sci. U.S.A.*, 108:11063–11068, 2011.

[86] R. Lyne, R. Smith, K. Rutherford, M. Wakeling, A. Varley, F. Guillier, H. Janssens, W. Ji, P. Mclaren, P. North, D. Rana, T. Riley, J. Sullivan, X. Watkins, M. Woodbridge, K. Lilley, S. Russell, M. Ashburner, K. Mizuguchi, and G. Micklem. FlyMine: an integrated database for Drosophila and Anopheles genomics. *Genome Biol.*, 8(7):R129, 2007.

[87] T. J. Macke, D. J. Ecker, R. R. Gutell, D. Gautheret, D. A. Case, and R. Sampath. RNAMotif, an RNA secondary structure definition and search algorithm. *Nucleic Acids Res.*, 29:4724–4735, 2001.

[88] B. E. Maden and J. M. Hughes. Eukaryotic ribosomal RNA: the recent excitement in the nucleotide modification problem. *Chromosoma*, 105(7-8):391–400, 1997.

[89] F. Major and P. Thibault. *RNA tertiary structure prediction. In Bioinformatics: From genomes to therepies (ed. T. Lengauer)*, volume I, pages 491–539. Wiley-VCH, Weinheim, Germany, 2007.

[90] K. C. Martin and A. Ephrussi. mRNA localization: gene expression in the spatial dimension. *Cell*, 136:719–730, 2009.

[91] D. H. Mathews and D. H. Turner. Dynalign: an algorithm for finding the secondary structure common to two RNA sequences. *J. Mol. Biol.*, 317(2):191–203, 2002.

[92] B. Mazumder, V. Seshadri, and P. L. Fox. Translational control by the 3'-UTR: the ends specify the means. *Trends Biochem. Sci.*, 28:91–98, 2003.

[93] J. S. McCaskill. The equilibrium partition function and base pair binding probabilities for RNA secondary structure. *Biopolymers*, 29:1105–1119, 1990.

[94] D. Moazed, J. M. Robertson, and H. F. Noller. Interaction of elongation factors EF-G and EF-Tu with a conserved loop in 23S RNA. *Nature*, 334:362–364, 1988.

[95] P. B. Moore. Structural motifs in RNA. *Annu. Rev. Biochem.*, 68:287–300, 1999.

[96] E. W. Myers and W. Miller. Optimal alignment in linear space. 4(1):11–17, 1988.

[97] E. P. Nawrocki, D. L. Kolbe, and S. R. Eddy. Infernal 1.0: inference of RNA alignments. *Bioinformatics*, 25:1335–1337, 2009.

[98] S. B. Needleman and C. D. Wunsch. A general method applicable to the search for similarities in the amino acid sequence of two proteins. *J. Mol. Biol.*, 48(3):443–453, 1970.

[99] P. Nissen, J. A. Ippolito, N. Ban, P. B. Moore, and T. A. Steitz. RNA tertiary interactions in the large ribosomal subunit: the A-minor motif. *Proc. Natl. Acad. Sci. U.S.A.*, 98:4899–4903, 2001.

[100] H. Pang, J. Tang, S. S. Chen, and S. Tao. Statistical distributions of optimal global alignment scores of random protein sequences. *BMC Bioinformatics*, 6:257, 2005.

[101] M. Parisien, J. A. Cruz, E. Westhof, and F. Major. New metrics for comparing and assessing discrepancies between RNA 3D structures and models. *RNA*, 15:1875–1885, 2009.

[102] A. G. Pedersen and H. Nielsen. Neural network prediction of translation initiation sites in eukaryotes: perspectives for EST and genome analysis. *Proc. Int. Conf. Intell. Syst. Mol. Biol.*, 5:226–233, 1997.

[103] J. S. Pedersen, G. Bejerano, A. Siepel, K. Rosenbloom, K. Lindblad-Toh, E. S. Lander, J. Kent, W. Miller, and D. Haussler. Identification and classification of conserved RNA secondary structures in the human genome. *PLoS Comput Biol*, 2(4):e33, 2006.

[104] M. Rabani, M. Kertesz, and E. Segal. Computational prediction of RNA structural motifs involved in posttranscriptional regulatory processes. *Proc. Natl. Acad. Sci. U.S.A.*, 105:14885–14890, 2008.

[105] R. R. Rahrig, N. B. Leontis, and C. L. Zirbel. R3D Align: global pairwise alignment of RNA 3D structures using local superpositions. *Bioinformatics*, 26(21):2689–2697, 2010.

[106] J. Reimand, M. Kull, H. Peterson, J. Hansen, and J. Vilo. g:Profiler–a web-based toolset for functional profiling of gene lists from large-scale experiments. *Nucleic Acids Res.*, 35:193–200, 2007.

[107] W. Ritchie, M. Legendre, and D. Gautheret. RNA stem-loops: to be or not to be cleaved by RNAse III. *RNA*, 13:457–462, 2007.

[108] E. Rivas and S. R. Eddy. Noncoding RNA gene detection using comparative sequence analysis. *BMC Bioinformatics*, 2:8, 2001.

[109] R. Salari, M. Möhl, S. Will, S. Cenk Sahinalp, and R Backofen. Time and space efficient rna-rna interaction prediction via sparse folding. In *RECOMB'10*, pages 473–490, 2010.

[110] D. Sambandan, M. A. Carbone, R. R. Anholt, and T. F. Mackay. Phenotypic plasticity and genotype by environment interaction for olfactory behavior in Drosophila melanogaster. *Genetics*, 179(2):1079–1088, 2008.

[111] D. Sankoff. Simulations solution of the RNA folding, alignment and protosequence problems. *SIAM J. Appl. Math.*, 45(5):810–825, 1985.

[112] K. Sargsyan and C. Lim. Arrangement of 3D structural motifs in ribosomal RNA. *Nucleic Acids Res.*, 38:3512–3522, 2010.

[113] M. Sarver, C. L. Zirbel, J. Stombaugh, A. Mokdad, and N. B. Leontis. FR3D: finding local and composite recurrent structural motifs in RNA 3D structures. *J. Math. Biol.*, 56:215–252, 2008.

[114] D. Schmucker, H. Jackle, and U. Gaul. Genetic analysis of the larval optic nerve projection in Drosophila. *Development*, 124:937–948, 1997.

[115] K. T. Schroeder, S. A. McPhee, J. Ouellet, and D. M. Lilley. A structural database for k-turn motifs in RNA. *RNA*, 16(8):1463–1468, 2010.

[116] K. Seggerson and P. B. Moore. Structure and stability of variants of the sarcin-ricin loop of 28S rRNA: NMR studies of the prokaryotic SRL and a functional mutant. *RNA*, 4:1203–1215, 1998.

[117] S. Smit, K. Rother, J. Heringa, and R. Knight. From knotted to nested RNA structures: a variety of computational methods for pseudoknot removal. *RNA*, 14:410–416, 2008.

[118] N. Spackova and J. Sponer. Molecular dynamics simulations of sarcin-ricin rRNA motif. *Nucleic Acids Res.*, 34:697–708, 2006.

[119] K. St-Onge, P. Thibault, S. Hamel, and F. Major. Modeling RNA tertiary structure motifs by graph-grammars. *Nucleic Acids Res.*, 35:1726–1736, 2007.

[120] J. Stombaugh, C. L. Zirbel, E. Westhof, and N. B. Leontis. Frequency and isostericity of RNA base pairs. *Nucleic Acids Res.*, 37:2294–2312, 2009.

[121] G. Storz. An expanding universe of noncoding RNAs. *Science*, 296(5571):1260–1263, 2002.

[122] S. A. Strobel, P. L. Adams, M. R. Stahley, and J. Wang. RNA kink turns to the left and to the right. *RNA*, 10:1852–1854, 2004.

[123] T. C. Sudhof. Neuroligins and neurexins link synaptic function to cognitive disease. *Nature*, 455(7215):903–911, 2008.

[124] S. Szep, J. Wang, and P. B. Moore. The crystal structure of a 26-nucleotide RNA containing a hook-turn. *RNA*, 9:44–51, 2003.

[125] A. A. Szewczak, P. B. Moore, Y. L. Chang, and I. G. Wool. The conformation of the sarcin/ricin loop from 28S ribosomal RNA. *Proc. Natl. Acad. Sci. U.S.A.*, 90:9581–9585, 1993.

[126] K. Tai. The Tree-to-Tree Correction Problem. *J. ACM*, 26(3):422–433, 1979.

[127] M. Tamura, D. K. Hendrix, P. S. Klosterman, N. R. Schimmelman, S. E. Brenner, and S. R. Holbrook. SCOR: Structural Classification of RNA, version 2.0. *Nucleic Acids Res.*, 32:D182–184, 2004.

[128] I. Tinoco, O. C. Uhlenbeck, and M. D. Levine. Estimation of secondary structure in ribonucleic acids. *Nature*, 230(5293):362–367, 1971.

[129] E. Torarinsson, J. H. Havgaard, and J. Gorodkin. Multiple structural alignment and clustering of RNA sequences. *Bioinformatics*, 23:926–932, 2007.

[130] A. Torres-Larios, A. C. Dock-Bregeon, P. Romby, B. Rees, R. Sankaranarayanan, J. Caillet, M. Springer, C. Ehresmann, B. Ehresmann, and D. Moras. Structural basis of translational control by Escherichia coli threonyl tRNA synthetase. *Nat. Struct. Biol.*, 9:343–347, 2002.

[131] H. H. Tseng, Z. Weinberg, J. Gore, R. R. Breaker, and W. L. Ruzzo. Finding noncoding RNAs through genome-scale clustering. *J Bioinform Comput Biol*, 7:373–388, Apr 2009.

[132] B. J. Tucker and R. R. Breaker. Riboswitches as versatile gene control elements. *Curr. Opin. Struct. Biol.*, 15:342–348, 2005.

[133] D. H. Turner, N. Sugimoto, and S. M. Freier. RNA structure prediction. *Annu Rev Biophys Biophys Chem*, 17:167–192, 1988.

[134] J. G. Underwood, A. V. Uzilov, S. Katzman, C. S. Onodera, J. E. Mainzer, D. H. Mathews, T. M. Lowe, S. R. Salama, and D. Haussler. FragSeq: transcriptome-wide RNA structure probing using high-throughput sequencing. *Nat. Methods*, 7:995–1001, 2010.

[135] I. Vidovic, S. Nottrott, K. Hartmuth, R. Luhrmann, and R. Ficner. Crystal structure of the spliceosomal 15.5kD protein bound to a U4 snRNA fragment. *Mol. Cell*, 6:1331–1342, 2000.

[136] L. M. Wadley and A. M. Pyle. The identification of novel RNA structural motifs using COMPADRES: an automated approach to structural discovery. *Nucleic Acids Res.*, 32:6650–6659, 2004.

[137] S. Washietl, I. L. Hofacker, and P.F. Stadler. Fast and reliable prediction of noncoding RNAs. *Proc. Natl. Acad. Sci. U.S.A.*, 102:2454–2459, 2005.

[138] R. M. Waterhouse, E. M. Zdobnov, F. Tegenfeldt, J. Li, and E. V. Kriventseva. OrthoDB: the hierarchical catalog of eukaryotic orthologs in 2011. *Nucleic Acids Res.*, 39(Database issue):D283–288, 2011.

[139] M. S. Waterman. Secondary structure of single stranded nucleic acids. *Adv. Math. Suppl. Stud.*, I:167–212, 1978.

[140] Z. Weinberg, J. Perreault, M. M. Meyer, and R. R. Breaker. Exceptional structured noncoding RNAs revealed by bacterial metagenome analysis. *Nature*, 462:656–659, 2009.

[141] Y. Wexler, C. Zilberstein, and M. Ziv-Ukelson. A study of accessible motifs and RNA folding complexity. *J. Comput. Biol.*, 14:856–872, 2007.

[142] S. Will, K. Reiche, I. L. Hofacker, P. F. Stadler, and R. Backofen. Inferring noncoding RNA families and classes by means of genome-scale structure-based clustering. *PLoS Comput. Biol.*, 3:e65, 2007.

[143] A. S. Williams and W. F. Marzluff. The sequence of the stem and flanking sequences at the 3' end of histone mRNA are critical determinants for the binding of the stem-loop binding protein. *Nucleic Acids Res.*, 23:654–662, 1995.

[144] B. T. Wimberly, D. E. Brodersen, W. M. Clemons, R. J. Morgan-Warren, A. P. Carter, C. Vonrhein, T. Hartsch, and V. Ramakrishnan. Structure of the 30S ribosomal subunit. *Nature*, 407:327–339, 2000.

[145] C. R. Woese, S. Winker, and R. R. Gutell. Architecture of ribosomal RNA: constraints on the sequence of "tetra-loops". *Proc. Natl. Acad. Sci. U.S.A.*, 87:8467–8471, 1990.

[146] I. G. Wool, A. Gluck, and Y. Endo. Ribotoxin recognition of ribosomal RNA and a proposal for the mechanism of translocation. *Trends Biochem. Sci.*, 17:266–269, 1992.

[147] H. Yang, F. Jossinet, N. Leontis, L. Chen, J. Westbrook, H. Berman, and E. Westhof. Tools for the automatic identification and classification of RNA base pairs. *Nucleic Acids Res.*, 31:3450–3460, 2003.

[148] K. Zhang and D. Shasha. Simple fast algorithms for the editing distance between trees and related problems. *SIAM J. Comput.*, 18(6):1245–1262, 1989.

[149] S. Zhang, I. Borovok, Y. Aharonowitz, R. Sharan, and V. Bafna. A sequence-based filtering method for ncRNA identification and its application to searching for riboswitch elements. *Bioinformatics*, 22:e557–565, 2006.

[150] S. Zhang, B. Hass, E. Eskin, and V. Bafna. Searching genomes for non-coding RNA using FastR. *IEEE/ACM Trans. Comput. Biol. Bioinf.*, 2(4):366–379, 2005.

[151] J. Zhao, G. Klyne, E. Benson, E. Gudmannsdottir, H. White-Cooper, and D. Shotton. FlyTED: the Drosophila Testis Gene Expression Database. *Nucleic Acids Res.*, 38:D710–715, 2010.

[152] M. Ziv-Ukelson, I. Gat-Viks, Y. Wexler, and R. Shamir. A Faster Algorithm for RNA co-folding. In: Workshop on Algorithms in Bioinformatics, pages 174–185, Berlin, Heidelberg, 2008. Springer-Verlag.

[153] M. Zuker. On finding all suboptimal foldings of an RNA molecule. *Science*, 244:48–52, 1989.

[154] M. Zuker. Mfold web server for nucleic acid folding and hybridization prediction. *Nucleic Acids Res.*, 31(13):3406–3415, 2003.

[155] M. Zuker and D. Sankoff. RNA secondary structures and their prediction. *Bull. Math. Biol.*, 46:591–621, 1984.