uOttawa

L'Université canadienne
Canada's university

FACULTÉ DES ÉTUDES SUPÉRIEURES ET POSTOCTORALES

uOttawa

L'Université canadienne
Canada's university

FACULTY OF GRADUATE AND POSDOCTORAL STUDIES

Beeta Masoumi

AUTEUR DE LA THÈSE / AUTHOR OF THESIS

Master of Computer Science

GRADE / DEGREE

School of Information Technology and Engineering

FACULTÉ, ÉCOLE, DÉPARTEMENT / FACULTY, SCHOOL, DEPARTMENT

Simulation Alignment and Structure Prediction for Three Ribonucleic Acid Sequences

TITRE DE LA THÈSE / TITLE OF THESIS

Marcel Turcotte

DIRECTEUR (DIRECTRICE) DE LA THÈSE / THESIS SUPERVISOR

CO-DIRECTEUR (CO-DIRECTRICE) DE LA THÈSE / THESIS CO-SUPERVISOR

EXAMINATEURS (EXAMINATRICES) DE LA THÈSE / THESIS EXAMINERS

Lucia Moura

Evangelos Kranakis

Gary W. Slater

LE DOYEN DE LA FACULTÉ DES ÉTUDES SUPÉRIEURES ET POSTDOCTORALES /
DEAN OF THE FACULTY OF GRADUATE AND POSTDOCORAL STUDIES

SIMULTANEOUS ALIGNMENT AND STRUCTURE PREDICTION FOR THREE RIBONUCLEIC ACID SEQUENCES

Beeta Masoumi

Thesis submitted to the
Faculty of Graduate and Postdoctoral Studies
in partial fulfillment of the requirements for the
M.Sc. degree in Computer Science

School of Information Technology and Engineering

Faculty of Engineering

University of Ottawa

Your file Votre référence
ISBN: 0-494-11344-8
Our file Notre référence
ISBN: 0-494-11344-8

Abstract

The first step in modeling the structure of an RNA molecule is the prediction of its secondary structure. Many computational techniques have been developed for this task. One approach, comparative sequence analysis, stands out for its remarkably accurate predictions. However, the technique has proven to be notoriously difficult to automate. Consequently, comparative sequence analysis still involves considerable human intervention.

Sankoff proposed a dynamic programming algorithm that can simultaneously solve the sequence alignment and folding problems for multiple sequences. It combines free energy minimization with comparative sequence analysis to improve the quality of secondary structure prediction. Dynalign has been the first implementation of this algorithm for two RNA sequences. Using more input sequences should improve the accuracy, reduce the likelihood that bad predictions are made, but also lower the sensitivity. To investigate these claims, we have extended the software system Dynalign to use three input sequences, rather than two, and tested our algorithm with 10 tRNAs and 13 5S rRNAs. Specifically, the following hypotheses were tested: 1) the use of three input sequences improves the average accuracy compared to predictions based on two input sequences. Since it should be less likely that all three input sequences simultaneously fold into a bad free energy minimum compared to predictions based on two sequences, 2) the worst prediction for any sequence should be more accurate when three input sequences are used rather than two. Finally, the consensus structure of three sequences is probably less representative of the individual sequences. Therefore, 3) the average coverage should be less compared to Dynalign.

List of Figures

Figure 1 Levels of organisation of RNA. 17
Figure 2 Elements of RNA secondary structure 18
Figure 3 Types of loops that are commonly found in RNAs 37
Figure 4 Traceback flowchart part 1 60
Figure 5 Traceback flowchart part 2 61
Figure 6 Effect of various gap penalty scores for tRNA dataset 70
Figure 7 Effect of various gap penalty scores for 5S rRNA dataset 71
Figure 8 Reference structure for RS0380 76
Figure 9 Dynalign prediction for RS0380 76
Figure 10 X-Dynalign prediction for RS0380 76
Figure 11 Reference structure for RD0500 86
Figure 12 Dynalign prediction for RD0500 87
Figure 13 X-Dynalign prediction for RD0500 87

List of Tables

Table 1 Thermodynamic parameters for Watson-Crick and GU base pairs 33
Table 2 Thermodynamic parameters for unpaired dangling nucleotides 35
Table 3 Thermodynamic parameters for terminal mismatches 36
Table 4 Free energy changes for the initiation of the three types of loops 38
Table 5 tRNA dataset 67
Table 6 5 S rRNA dataset 67
Table 7 X-Dynalign lower sensitivity vs Dynalign predictions 72
Table 8 X-Dynalign does not degrade already good predictions of Dynalign 73
Table 9 Mean sensitivity, PPV, and MCC for tRNA dataset 75
Table 10 Sensitivity for tRNA dataset 77
Table 11 PPV for tRNA dataset 78
Table 12 MCC for tRNA dataset 78
Table 13 X-Dynalign improves accuracy vs. Dynalign for tRNA 79
Table 14 Mean sensitivity, PPV, and MCC for 5 S rRNA dataset 80
Table 15 Sensitivity for 5 S rRNA dataset 81
Table 16 PPV for 5 S rRNA dataset 81
Table 17 MCC for 5 S rRNA 81
Table 18 X-Dynalign improves accuracy vs. Dynalign for 5 S rRNA 82
Table 19 MFOLD results for tRNA sequences 85
Table 20 MFOLD results for 5 S rRNA sequences 85

I would like to take this opportunity to thank those who contributed in one way or another to this work.

Dedication

This thesis is dedicated to *Arash,* my supportive friend, my caring husband who patiently, months and months, has been waiting for me to finish this thesis. Without his encouragements this would not be possible. Thank you for your love and your support!

Acknowledgements

My deepest gratitude goes to my supervisor, *Professor Marcel Turcotte.* I have been blessed by his exceptional knowledge, wonderful attitude, amazing understating and patience, continuous support and his generosity of his time. I am greatly indebted to him.

To my lovely *mom & dad* who have always been there for me: your love and your presence has nurtured me in so many wonderful ways. Thank you for everything you do for me. You are simply the best mom & dad in the world.

To my smart brother, *Pooyaun*: your enthusiasm gives me so much energy, your sweet sense of humour makes me laugh, and your big apple-shaped heart always inspires me. Having you as a brother is the best gift in the world. Thank you for being you!

I would like to express my gratitude to *David H. Mathews* for providing invaluable comments on the Dynalign paper. I would also like to thank *Mohammad Anwar* for his help with the final editing of this thesis.

Glossary

A: Adenine, a constituent of DNA and RNA.

Base pair: the hydrogen-bounded structure formed by two complementary nucleotides.

Calorimetry: the science of measuring the heat of chemical reactions or physical changes.

Complementary: refers to Watson-Crick base pairs, C:G and A:T (or A:U in RNA).

C: Cytosine, a constituent of DNA and RNA.

Entropy: measures the energy of a physical system that cannot be used for work. It is often viewed as a measure of the disorder of a system.

G: Guanine, a constituent of DNA and RNA.

Hydrophobic: water-hating chemical group.

Metabolites: metabolites are the intermediates and products of metabolism, usually small molecules.

Nuclease: an enzyme that degrades a nucleic acid molecule.

Nucleotide: a purine or pyrimidine base attached to a five-carbon sugar, to which a mono-, di-, or tri-phosphate is also attached. The monomeric unit of DNA and RNA. Also known as "base".

Purine: one of the two types of nitrogenous base making up the nucleotides. It comprises Adenine and Guanine.

Pyrimidine: one of the two types of nitrogenous base making up the nucleotides. It comprises Cytosine and Uracil (or Thymine).

Ribonuclease: an enzyme that degrades RNA.

Ribonuclease P: an enzyme involved in processing pre-tRNA in bacteria.

Ribozyme: an RNA molecule that has catalytic activity.

Secondary structure: the secondary structure consists of hydrogen-bonded pairings between complementary bases (G and C or A and U) and the loops formed by unpaired bases.

Tandem repeat: direct repeats that are adjacent to each other.

Tertiary structure: the tertiary structure is made up of interactions between secondary structures, generally through the formation of additional hydrogen bonds.

Transfer RNA (tRNA): a small RNA molecule that acts as an adaptor during translation and is responsible for decoding the genetic code.

T: Thymine, one of the pyrimidine nucleotides found in DNA.

U: Uracil, one of the pyrimidine nucleotides found in RNA.

Watson-Crick base pair: two complementary nucleotides on opposite strands that are connected via hydrogen bonds.

Wobble base pair: a GU pair in RNA secondary structure.

X-ray crystallography: a technique for determining the three-dimensional structure of a large molecule.

X-ray diffraction: the pattern obtained after diffraction of X-rays through a crystal.

Table of Contents

CHAPTER 1 ..... **10**
1-1 Essential Cell Biology **11**
1-2 RNAology **12**
1-2-1 Central Dogma 12
1-2-2 Main Classes of RNA 13
1-2-3 RNA Therapeutics 15
1-2-4 RNA World Hypothesis 15
1-3 RNA Folding Problem: 2D and 3D **15**
1-3-1 Assumptions 20
1-3-1-1 Constraints on Secondary Structure 20
1-3-1-2 Cycles 20
1-4 Organization of the thesis **22**
1-5 Contributions **23**

CHAPTER 2 ..... **24**
2-1 Non-Computational Approaches **25**
2-1-1 X-ray Diffraction Methods for Structural Analysis 26
2-1-2 NMR Methods for Studying Nucleic Acid 26
2-1-3 Molecular Modeling and Simulation of Nucleic Acids 27
2-1-4 Chemical and Enzymatic Probes of Structure 28

CHAPTER 3 ..... **29**
3-1 Thermodynamics of RNA Motifs **30**
3-1-1 Watson-Crick Helical Regions 31
3-1-2 GU Pairs 34
3-1-3 Dangling Ends and Terminal Mismatches 34
3-1-4 Loops 37
3-1-4-1 Hairpin Loops 37
3-1-4-2 Bulge Loops 39
3-1-4-3 Internal Loops 39
3-1-4-4 Multibranch Loops 41
3-2 Computational Approaches **41**
3-2-1 Free Energy Minimization 42
3-2-2 Combinatorial and Recursive Algorithms 43
3-2-3 Comparative Sequence Analysis Algorithms 44
3-2-4 A Genetic Algorithm with Energy Minimization 45
3-2-5 Structure Optimization by Energy Minimization: The MFOLD Algorithm 45
3-2-6 Free energy Minimization and Comparative Sequence Analysis in One: 46
The Dynalign Algorithm 46

CHAPTER 4 ..... **49**
4-1 X-Dynalign: RNA Secondary Structure Prediction **50**
4-1-1 Algorithm 51

4-1-2 Detailed Algorithm 51
4-1-3 Complexity Analysis 62

CHAPTER 5 64
5-1 Performance Measures 65
5-2 Experiments 66
5-2-1 Calibrating Gap Penalties 68
5-3 Results 72
5-3-1 Comparative Analysis 75
5-3-1-1 tRNA Dataset 75
5-3-1-2 5S rRNA Dataset 80

CHAPTER 6 88
6-1 Conclusion 89
6-2 Future work 90
APPENDIX A 93
REFERENCES 106

CHAPTER 1

INTRODUCTION

IN THIS CHAPTER

- Essential cell biology
- RNAology
- RNA folding problem: 2D and 3D
- Organization of the thesis
- Contribution

CHAPTER ONE

The appreciation of RNA is evolving both as a tool for increased understanding of biological systems, as well as the study of molecular evolution, and perhaps more importantly, as a target for therapeutic intervention [8]. Among all the molecules of living organisms, RNA is the one that can play exceptionally different roles in different living cells. In this chapter, we introduce RNA and its features at first. Then, we will explain the RNA folding problem and the challenges of RNA structure prediction.

1-1 Essential Cell Biology

Cells are the smallest basic units of life. One way to classify cells is to divide them based on their internal structure: Prokaryotic cells and Eukaryotic cells. Prokaryotic cells are structurally simple and are only found in single-celled and colonial organisms. Archaea and eubacteria are members of this group. Eukaryotic cells are more complex and also exist in multicellular forms, such as animalia and plantae.

All cells, whether Eukaryotic or Prokaryotic, have a protective coat, which is called the membrane. Inside the membrane there is a large fluid-filled space called the cytoplasm. In Eukaryotes, the cytoplasm contains many organelles-like nuclei, such as the mitochondria.

The nucleus is the place where the DNA is located and where RNA is transcribed. The ribosomes are the sites of protein synthesis (where RNA is translated into protein). They

are made up of many proteins and several RNAs. The main focus of this thesis is on RNA molecules.

1-2 RNAology

RNA or ribonucleic acid is an organic acid composed of repeating nucleotide units: adenine, guanine, cytosine, and uracil. It plays an important role in the protein synthesis where the genetic information from the DNA is first copied to RNA (transcription) and then translated into proteins (translation). RNA is not just a passive structural element. It is also an active component in many reactions. For instance, RNA molecule acting alone is able to catalyze RNA processing [5]. In a protein-RNA complex, the RNA component of Ribonuclease P is an active component of tRNA processing [5].

Structurally, RNA is similar to DNA. RNA molecules often fold into more complex structures involving complementary internal sequences. That is, one part of a single RNA molecule is the nucleic acid complement of another part of the same molecule, so that the two strands bind together. This allows the formation of hairpin loops, coils, etc., which then direct the formation of higher-order structures.

The activity of RNA is determined by its structure, the way it is folded back on itself. The function of RNA can only be understood in terms of its secondary or tertiary structure [25]. For the understanding of catalytic activity, knowledge of secondary structure alone is insufficient [25]. Although few large structures have been determined by crystallography, still the need for modeling is great.

1-2-1 Central Dogma

The central dogma of molecular biology states that the flow of genetic information travels from DNA to RNA and finally to proteins.

DNA → RNA → Protein

DNA is replicated in the nucleous of the cell using one strand of the double helix as a template. The process by which the information contained in a section of DNA is

transferred to a newly assembled messenger RNA (mRNA) is called transcription. The transcription is facilitated by the RNA polymerase and transcription factors. The mRNA is then transported out of the nucleus, into the cytoplasm of Eukaryotes, and it eventually finds its way to the ribosomes where proteins are manufactured through a process called translation. A series of 3 adjacent nucleotides, or codon, is responsible for coding one amino acid. The amino acids are carried to the site of translation by transfer RNA molecules. Long chains of these 20 different amino acids form proteins. The following is a simple example of the protein synthesis process:

```
5′ ATGGCC 3′ (sense strand)
3′ TACCGG 5′ (antisense strand)
5′ AUGGCC 3′ mRNA (transcription of antisense)
  Met-Ala   peptide (translation)
```

The start and end positions of a nucleic acid chain are labelled 5′ and 3′ respectively. The notation is derived from the labels of the carbon atoms on the sugar ring.

1-2-2 Main Classes of RNA

There are many different kinds of RNA, each serving a different purpose: messenger RNA, non-coding RNAs or RNA genes including transfer RNA and ribosomal RNA, and double stranded RNA.

Messenger RNAs carry information from the DNA to the ribosome sites. Once an mRNA has been transcribed, it is exported from the nucleus into the cytoplasm (in Eukaryotes the mRNA is "processed" before being exported), where it is bound to ribosomes and translated into protein. After a certain amount of time the mRNA is degraded into its component nucleotides, usually with the assistance of RNases. The mRNA molecules are heterogenous in size and sequence. Besides serving as a template for the protein synthesis, mRNAs are also playing an important role in the regulation of gene expression. Messenger RNAs consists of three parts: 5′ UTR (untranslated region), coding region and 3′ UTR. Furthermore, in Eukaryotes the coding region is also interrupted by one or several non-coding regions called introns. Several discoveries in the last few years have helped us

understand the many roles of the UTRs. In the 5′ UTRs of bacteria, structural motifs, termed riboswitches, have been found that can sense the level of specific metabolites and block the translation of the transcript accordingly[22]. Other motifs, particularly in the 3′ UTRs, have been linked to the routing of the transcript in the cell, as well as their degradation.

RNA genes are genes that encode functional RNA molecules; in contrast to mRNA, these RNAs do not code for proteins. The best-known examples are transfer RNA and ribosomal RNA. Both classes of RNA genes participate in the process of translation. Ribosomal RNA is a crucial part of the ribosome, which is found in all living cells [5]. Transfer RNA acts as an adaptor during translation and is responsible for decoding the genetic code [5]. There are more than 20 different tRNA molecules and all have between 75-95 nucleotides. The rRNA molecules make up at least 80% of the RNA molecules found in a typical eukaryotic cell.

Small nuclear RNA (snRNA) refers to a number of small RNA molecules found in the nucleus. These RNA molecules are important in a number of processes including RNA splicing (converting pre-mRNA into mRNA by removing the introns and splicing the exons).

Double-stranded RNA (dsRNA) is RNA with two complementary strands, similar to the DNA found in all "higher" cells. dsRNA forms the genetic material of some viruses. In Eukaryotes, it may play a role in the process of RNA interference. Typically, they are more than 200 nucleotides in a dsRNA molecule.

Small interfering RNA (siRNA) is another class of 20-25 nucleotide-long RNA molecules that interfere with the expression of genes. These are produced as part of the RNA interference (RNAi) process. In this process, long double-stranded RNAs can be used to silence the expression of target genes. Thus, RNA interference is a gene-silencing technique used in studying the absence of normal gene action by disrupting its activity *in vivo* (i.e. turning it off).

1-2-3 RNA Therapeutics

RNA is a molecule of many faces, with amazing capabilities. It is expected that exciting developments in RNA-based drug discovery and RNA therapeutics research will be made in a near future. To design new agents, it is critical to better understand the sequence-structure-activity relationship.

The dsRNA that triggers RNAi may be usable as drugs. For example, dsRNA could repress essential genes in human pathogens that are dissimilar from any human genes; this would be analogous to action mechanism of existing drugs [20].

Nonsense mutations are the causes of many inherited diseases such as cystic fibrosis and haemophilia. Nonsense suppressor tRNAs have been suggested as potential agents for human somatic gene therapy for diseases caused by nonsense codons [3].

1-2-4 RNA World Hypothesis

The RNA World hypothesis suggests that the first form of life on earth may have been RNA-based. It states that RNA was dominant, before the emergence of the first cell and probably was the only form of life. This hypothesis is supported by RNA's ability to participate in the storage, transmission, and duplication of genetic information [10]. From the point of view of reproduction, molecules exist for two basic purposes: self-replication and catalysis assisting self-replication [5]. DNA is capable of self-replication, but only assisted by proteins. Proteins are excellent catalysts, but fail to catalyze processes complex enough to recreate themselves, individually [5]. RNA is capable of both catalysis and self-replication [5]. The phrase "The RNA World" was first used by Walter Gilbert in 1986 [10].

1-3 RNA Folding Problem: 2D and 3D

Biological molecules must fold into the correct three-dimensional shape to acquire their active, functional form [27]. The folding of biological molecules is directed by the physical and chemical properties inherent to their molecular makeup [17]. The blueprint for the

construction of biological molecules is contained in the genetic material of each organism [17]. It is possible to sequence the genetic information and determine the message present in the genes that code for the covalent structure (primary structure) of biological molecules [17]. Since the advent of rapid methods for sequencing DNA, there has been an exponential growth in nucleic acid sequence information. This progress will continue into the foreseeable future. To make full use of this information, it will be necessary to determine the secondary and tertiary structures of the biological molecules encoded by this information [23].

The primary structure is determined by the sequence of G, A, C, and U bases in a strand. The secondary structure consists of hydrogen-bonded pairings between complementary bases (G and C or A and U) and the loops formed by unpaired bases. RNA secondary structure is generally divided into helices (contiguous base pairs), and various kinds of loops (unpaired nucleotides surrounded by helices).

The tertiary structure is made up of interactions between secondary structures, generally through the formation of additional hydrogen bonds or hydrophobic interactions (see Figure 1). The determination of the tertiary structure is more complex than the secondary structure [5]. This is mainly because for secondary structures there are no continuously varying parameters such as bond lengths, angles, or inter-atomic distances, which must be accounted for in tertiary structure [5]. It is generally assumed that the influence of the tertiary structure on the secondary structure is negligible; consequentially, secondary structures can be determined independently of tertiary structures [12].

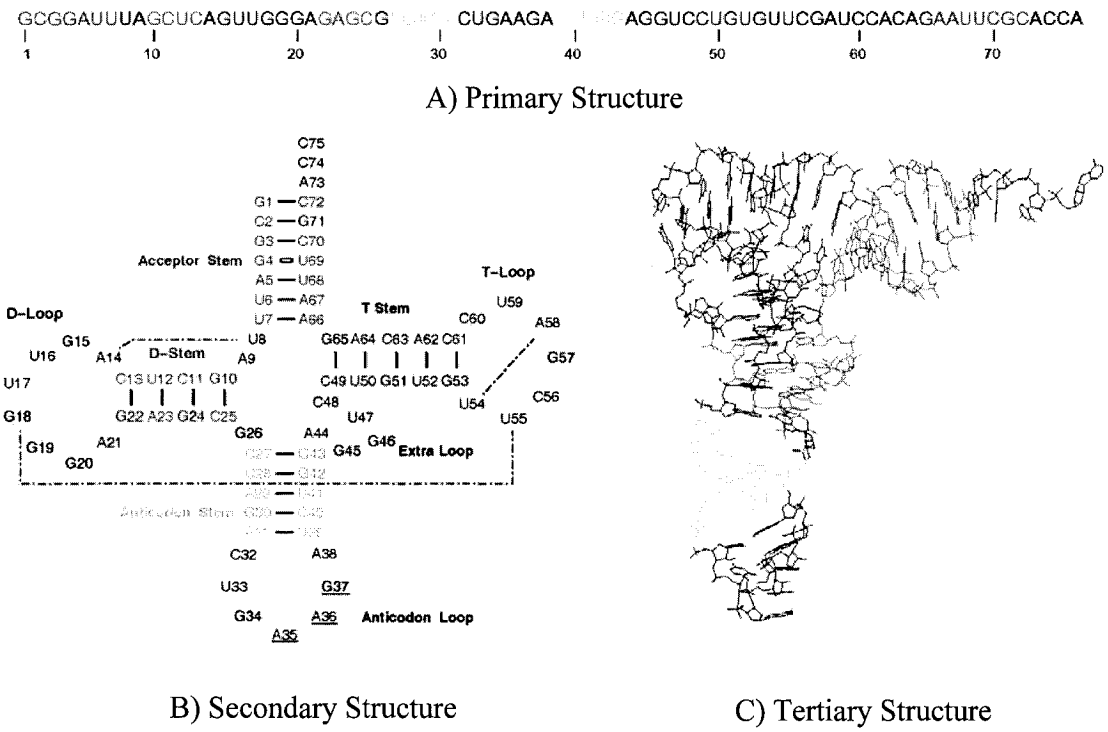


Figure 1 Levels of organisation of RNA. The figure shows the primary, secondary and tertiary structure of the yeast tRNA ^{Phe}.

Under natural conditions, a ribonucleic chain will twist and bend, and the bases will form bonds with one another in a complicated pattern [21]. Both the conformation and the pattern of bonding are called the secondary structure [21], but this thesis deals only with the pattern of bonding. Many interesting RNAs conserve a secondary structure of base-pairing interactions more than they conserve their sequences [25]. This makes RNA sequence analysis more complicated and difficult than protein or DNA sequence analysis.

One of the major problems in computational molecular biology is the development of folding algorithms capable of predicting the three-dimensional (tertiary) structure of RNA molecules based on sequence information. NMR (nuclear magnetic resonance) and X-ray crystallography are two of the methods for determining RNA structure. However, these

methods cannot keep pace with the rate of sequencing of new RNA molecules. Thus, there is need for other reliable methods of determining RNA structure. To be able to determine the three-dimensional structure, one approach is to solve a simpler problem first. That is the prediction of secondary structure, since it provides many constraints [25].

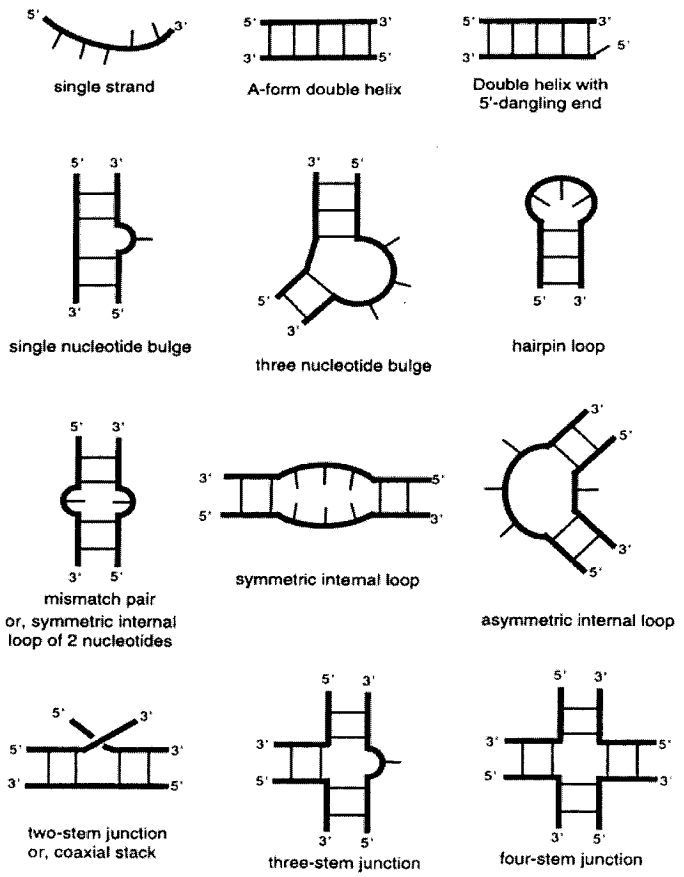


Figure 2 Elements of RNA secondary structure. A secondary structure can be divided into single-strand regions, helices, bulges, hairpin loops, internal loops, and junctions. The distinction between a single-strand region and a bulge, loop, or junction is that in a single-strand the ends are not constrained. In contrast, the ends of bulges, loops, or junctions must be in a tightly limited volume. Single-strand regions next to helices are dangling ends; the dangling nucleotide may be on a 5′ end, or a 3′ end. A dangling mismatch is produced by apposing 5′ and 3′ dangling nucleotides. (reproduced from [10])

Qualitatively, the way single-stranded RNAs organize themselves was understood almost 40 years ago. They fold in a way that the short subsequences they contain –that are "accidentally" complementary– form short double helices to (approximately) the maximum extent possible. The dominant structural element that results is the *hairpin loop,* or *stem-*

loop, which is produced when an RNA chain folds back on itself so that complementary sequences close to each other in its sequence can pair. Thus, most RNAs have secondary structures that consist of a series of stem-loops separated by sequences of less certain conformation that are usually represented as single-stranded. Figure 2 shows elements of RNA secondary structure.

In addition to occasional non-canonical base pairs, helical stems are often interrupted by *bulged bases* and by *internal loops*. Bulged bases are bases on one strand that have no partner (to pair with) on the other strand. *Internal loops*, are the ones in which longer sequences on both strands are juxtaposed that cannot obviously be paired. Some internal loops have sequences long enough to include stem-loops of their own; they are called *junctions.* Whether the stem of a stem-loop contains irregularities or not, it must have a *terminal loop,* i.e. a sequence that links the 5′- to the 3′-strand of its stem. The terminal loops of some stem-loops are big enough to contain stem-loops of their own.

Unfortunately, a huge number of secondary structures are possible for any given sequence. For example, when A, C, G, and U occur randomly with equal probability, the number of valid secondary structures is greater than 1.8^N, where N is the number of nucleotides [28]. To make the connection between structure and function, it is necessary to determine the one or few native foldings [27]. One method that can be used to restrict the number of secondary structures is free energy minimization [27]. In this method, all the folds are considered. However, only the relevant ones are presented to the user. In principle, this method can predict the equilibrium secondary structure. In practice, only limited experimental data are available for the parameterization, and small changes in energy parameters often result in large predicted folding changes [28]. Thus, the problem is ill-conditioned in a mathematical sense. Moreover, a cell is not at equilibrium, so there are no fundamental reasons for the lowest free energy structures and the biologically important structures to be the same [28].

1-3-1 Assumptions

The following assumptions are commonly made for making the secondary structure problem computationally tractable. These assumptions were also considered for development of X-Dynalign.

1-3-1-1 Constraints on Secondary Structure

A secondary structure S for a sequence p is a set of pairs *i:j* with *i<j* (*i, j* are positions in the sequence *p*). The set S is required to satisfy the following constraints [24]:

1. Watson-Crick or Wobble base-pairs
2. No overlap of pairs: If S contains *i:k*, then it cannot contain *i:j* with $j \neq k$ or *j:k* with $j \neq i$.
3. No knots: If $h < i < j < k$, then S cannot contain both *h:j* and *i:k*.
4. No sharp turns: If S contains *i:j*, then $|j-i|>=4$.

It follows from 2 that each *i* occurs either in exactly one pair or in no pairs, and *i* is described as paired or unpaired accordingly. Constraint 4 is local, while 2 and 3 constraint the pairing in a global way. All these constraints derive from a variety of stereo chemical, mechanical, and thermodynamic considerations.

Constraints 2 and 3 are occasionally violated in nature, but the resulting triplets and knots are considered features of a higher level of organisation –tertiary structure– that can sometimes be determined *a posteriori*. The majority of the secondary structure prediction algorithms also make these assumptions.

1-3-1-2 Cycles

The following are different types of substructures (cycles) described formally. If *i:j* is a pair and $i < r < j$, we say that *i:j* surrounds *r*. Similarly; *i:j* surrounds a pair *p:q* if it surrounds both *p* and *q*; due to constraint 3, forbidden knots, if *i:j* surrounds either *p* or *q*, it must surrounds both *p* and *q*. If $i < p < q < j$, then *p:q* is called the interior base pair of the loop and *i:j* is called the exterior base pair, and is said to close the loop.

1. if S contains *i:j* but none of the surrounded elements *i+1, ..., j-1* are paired, the loop formed is called a *hairpin.*
2. if S contains *i:j*, *(i+1):(j-1), .., (i+h):(j-h)*, each of these pairs may be referred to as a *stacked pair.*
3. if $i+1 < p < q < j-1$ and S contains *i:j* and *p:q*, but the elements between *i* and *p* are unpaired and the elements between *q* and *j* are unpaired as well, then the two regions are said to constitute an *interior loop.*
4. if S contains *i:j* and *(i+1):q*, and there are some unpaired elements between *q* and *j*, these unpaired elements form a *bulge.* Symmetrically, a bulge also occurs if S contains *i:j*, and *p:(j-1)* and some unpaired elements between *i* and *p.*
5. if S contains *i:j* and *i:j* surrounds two or more pairs *p:q*, *r:s*, ... which do not surround one another, then a *multiple loop* is formed.
6. if *r* is unpaired and there is no pair in S surrounding *r* then we say that *r* is in an *external single-stranded region.*

Note that any region containing unpaired base(s) is a loop, and that the only kind of region that is not a loop is a stacked-pair region [24].

Terminal Loops

Terminal loops are concise structures that stabilize 180° changes in backbone direction.

U-turns

The U-turn is a four-base terminal loop motif. Its consensus sequence is UNRN. (U stands for Uracil, N stands for any nucleotides, and R means any purine). They were first characterized in the mid-1970s by crystallographers working on transfer RNAs, and their existence in tRNAs in solution has been confirmed. All of the U-turns characterized so far are components of larger terminal loops.

Tetraloops

In the late 1980s, it was noticed that helical stems terminated by 4-nucleotide loops, or *tetraloops,* having the sequence UNCG are unusually abundant in rRNAs. It was demonstrated that they are unusually stable. Further analysis revealed the existence of two other "special" tetraloop sequences: GNRA and CUNG.

Internal Loops

Internal loops are flanked by two helices with canonical pairs, and contain nucleotides on both strands that are not in canonical pairs. RNA internal loops may play important roles in tertiary interactions and in protein recognition. Stabilities of internal loops are very dependent on the identity and orientation of closing base pairs, on the sequence in the loop, and on the size and symmetry of the loop.

Pseudoknots

Many RNAs contain pseudoknots, which are structures in which the loop of some hairpin forms a double helix by pairing with nucleotides from another part of the same molecule. In order to keep the complexity of the algorithm at a reasonable order, we chose to use the "no knot" constraint, which means the algorithm does not predict pseudoknots. Rivas and Eddy have included pseudoknots in the recurrence equations describing the RNA secondary structure prediction problem [22]. However, the huge runtime severely limits its application.

1-4 Organization of the thesis

There are 6 chapters in this thesis, including this chapter. We had a brief introduction to RNA and its structure prediction problem in chapter one. In the next two chapters, we will explain non-computational and computational approaches for determining and predicting RNA secondary structure. Then, in Chapter 4 X-Dynalign is presented. The results and future work are presented in Chapter 6 and 7 respectively.

1-5 Contributions

We have extended the software system Dynalign to use three input sequences, rather than two. The resulting system is called eXtended-Dynalign (X-Dynalign for short). We incorporate a third sequence. Consequently, the program is transformed into a six dimensional dynamic programming algorithm. Accordingly, all the recurrence equations are also modified. For instance, there are 16 different equations for a multibranch loop matrix in Dynalign while in X-Dynalign this increases to 64 equations. We ran ΔG_{gap} calibration experiments as well. The results show that in X-Dynalign the gap penalty is less dependent (in our limited experiments not dependent) on the type of RNA.. Our experiments have shown that:

1. Using three input sequences -instead of two- improves the average accuracy. The more the input sequences, the less likely they all fold into an unfavourable minimum free energy.
2. The worst prediction should be more accurate when three input sequences are used rather than two.
3. On the other hand, the common structure of three sequences should be less representative of the structure of each individual sequence, as a result the average coverage should be less than that of for Dynalign.

The results of this work have been published in [16] and [17].

CHAPTER 2

NON-COMPUTATIONAL APPROACHES

IN THIS CHAPTER

- X-ray Diffraction
- NMR
- Molecular modeling
- Chemical and Enzymatic probes

CHAPTER TWO

Our knowledge of DNA and RNA three-dimensional structure has advanced considerably since the elucidation of the DNA double helix. DNA structure itself continues to surprise with its ability to exist in a wide variety of forms, such as left-handed (most DNAs twist as a right-hand screw) and multiple-stranded helices. The study of RNA structure reveals that RNA can fold in a variety of complex ways, as well as double-helical form. This presentation of experimental approaches to structure determination is based on [21].

2-1 Non-Computational Approaches

The determination of the double helix provided new insights into the nature of genetic events. Now we have extensive knowledge of both the detail and the variety of DNA and RNA structures as well as the way they interact with other molecules. This is giving us a more profound understanding of processes such as gene regulation, transcription and translation, mutation, and drug action at the molecular level [12].

Advances in nucleic acid structure have been largely due to the increased power and sophistication of X-ray crystallography [9]. NMR (Nuclear Magnetic Resonance) spectroscopy, molecular modeling/simulation and chemical/biochemical probe techniques also play important roles in providing complementary information on nucleic acid structure [21]. Traditional spectroscopic-based biophysical methods can provide important

complementary information, mostly at the macroscopic level. More recently developed techniques and atomic force microscopy are extending their power so that the gap is diminishing between macroscopic data on nucleic acids which these methods provide, and that at the atomic level from X-ray crystallography and NMR [21].

2-1-1 X-ray Diffraction Methods for Structural Analysis

This method is based on the reconstruction of the internal molecular arrangement by analysis of the scattered X-rays. It is very similar to a lens focusing scattered light from a microscope sample. X-ray diffraction method provides a picture of the electron density distribution in the molecule. Due to the loss of information during the diffraction process this reconstruction is complex [21].

The accuracy and reliability of the resulting structure depends in part on the quantity and resolution of the diffraction data, as well as the quality of its measurement. Of key importance is the actual correctness of the structural model itself.

One of the limitations of this method is that experiments are done in solid state, while in reality, these molecules are floating in liquid. Furthermore, finding adequate experimental conditions to grow the necessary crystals has proven to be difficult for nucleic acids. This is exemplified by the fact that PBD, the public repository of nucleic acid and protein structures, contains only 747 entries for RNA/DNA compared to 21,563 proteins, as of September 21, 2004 [1].

2-1-2 NMR Methods for Studying Nucleic Acid

The underlying principle of nuclear magnetic resonance is the detection of atomic nuclei in a molecule via magnetic field. Protons are abundant in nucleic acids and oligo-nucleotides, and fortunately have readily detectable spin signals. These signals, termed chemical shifts, are dependent on the shielding effect of neighbouring protons, and thus can be used to determine the chemical environment of a proton once they can be assigned as arising from particular atoms. NMR studies of oligo-nucleotides have been extensively used to examine

interactions, by monitoring characteristic changes in particular chemical shifts.

Solution-phase studies have the obvious advantage that molecules do not have to be crystallized, which is often the major limitation to the analysis of a macromolecule by X-ray crystallography. There is also the apparent advantage that a structure determined in solution is more relevant to physiological processes than an X-ray crystallographic study in the solid state. However, the two techniques should not be considered as alternatives. Rather they are complementary, providing distinct information.

There are a number of limitations to the accuracy and reliability of NMR methods as applied to nucleic acids. By contrast with crystallography, there is a limitation on the size of the problem that can be analyzed in detail.

2-1-3 Molecular Modeling and Simulation of Nucleic Acids

Molecular modeling techniques enable dynamic changes in structure and conformation to be calculated and visualized. The theoretical methods thus provide information complementary to the experimental techniques.

It is not feasible at present time to compute the conformational or energetic properties for lengths of nucleic acid sequence by quantum mechanics. Instead, empirical force-field methods are widely used. These have been derived from experimental data that describe the energetics of a DNA or RNA molecule in terms of the sum of a number of factors.

The use of molecular mechanics and dynamics methods has greatly increased in recent years, due to the ready availability of high-performance computing facilities. There are a number of modeling programs in common use which have their force fields parameterized specifically for nucleic acids.

The most widely used simulation programs are:

1. AMBER (Assisted Model Building and Energy Refinement), from the laboratory of P. A. Kollman, University of California, San Francisco.

2. CHARMM (Chemistry at Harvard Macromolecular Mechanics), from the laboratory of M. Karplus, Harvard University.
3. GROMOS (Groningen Molecular Simulation), from the laboratory of W. F. van Gunsteren and H. I. C. Berendsen, University of Groningen, The Netherlands.
4. JUMNA (Junction Minimization of Nucleic Acids) from the laboratory of R. Lavery, *Institut de Biologie Physico-Chimique*, France. This program enables the torsion angles and helicoidal parameters in a structure to be varied, rather than the Cartesian coordinates. It enables large regions of conformational space to be rapidly explored.

2-1-4 Chemical and Enzymatic Probes of Structure

The enzyme DNase cleaves the phosphodiester bonds of a DNA duplex at every nucleotide position. However, the cutting efficiency is markedly dependent on sequence, and by implication, on sequence-related structural features. Cleavage may be blocked by protein or drug binding. Hence, DNase can be used to determine sites of binding along a DNA sequence as well as to assess possible effects of particular sequences on DNA structure. Chemical cleaving agents, such as hydroxyl radicals, can give similar information. Since these are much smaller molecules than cleavage enzymes, their effects on DNA structure are less perturbing and sequence-dependent. Other types of chemical probe can attack specific base sites. These can be useful in defining the precise sites of protection resulting from drug or protein binding to a DNA sequence.

These methods have the important advantage, over the fine-structure techniques of crystallography and NMR, of being applicable to long (up to several thousand base pair) DNA sequences, and thus of being more directly relevant to DNA in the cell. Hence, the use of chemical and enzymatic probes for DNA provides a way of obtaining at least some molecular-level data on otherwise inaccessible structural problems in DNA-protein and drug recognition. Meanwhile chemical and enzymatic probes are often used in conjunction with computational approaches. Those experiments provide constraints that can be fed to computational tools.

CHAPTER 3

COMPUTATIONAL APPROACHES

IN THIS CHAPTER

- Thermodynamics of RNA Secondary structure
- Free energy minimization
- Combinatorial and recursive algorithms
- Comparative sequence analysis
- Genetic algorithm with free energy minimization
- Comparative sequence analysis with free energy minimization

CHAPTER THREE

Chapter 1 introduced the basic concepts about RNA structure. It was observed that biological molecules must fold into the correct three-dimensional shape to acquire their active, functional form. In chapter 2, the main non-computational methods for determining RNA structure were introduced. It was shown that the critical issues with those non-computational methods are time and cost. Therefore, predicting the structure of RNA molecules, based on sequence information, is a major accomplishment. To be able to determine the tertiary structure, one approach could be solving a simpler problem and that is the determination of secondary structure. In this chapter, we introduce some well-known algorithms for RNA secondary structure prediction.

3-1 Thermodynamics of RNA Motifs

Thermodynamics of RNA motifs are fundamental concepts for many of the computational techniques introduced here. The nearest neighbour model breaks down the total free energy into the sum of independent contributions. This section presents the motifs that are taken into account by this model.

Calorimetry is one of the main methods for determining thermodynamic parameters of nucleic acids. In this method, the heat lost or gained during the structural changes are measured. The information obtained by calorimetry for a particular RNA can be later used

for prediction of thermodynamic values of other RNA molecules with similar sequences [28].

3-1-1 Watson-Crick Helical Regions

Watson-Crick base pairs involve about 50% of the nucleotides in an RNA sequence [28]. They have been studied through different experiments. Results from experiments show that the base pair composition alone doesn't explain the thermodynamic properties of this motif. Consider the duplexes (5'GUUCGAAC3') and (5'UUGGCCAA3'), both have four AU and four GC base pairs, but have different free energy changes of duplex formation (-8.8 and -11.0 kcal/mol, respectively) [28]. Experiments like these show that there are sequences with the same base composition but different order that have different free energies [28]. Therefore, the stability depends not only on the number of hydrogen bonds formed, but also on vertical stacking interactions between neighbouring base pairs [28].

The nearest-neighbour model is an established model for the approximation of the base pair interactions. This model assumes that the stability of a given base pair depends on the identity of its adjacent base pair (nearest neighbour) [26]. X-Dynalign uses the parameters from nearest neighbour model of Xia *et al* [29] as implemented in the computer program Dynalign. This model assumes that the stability of a duplex depends on several factors including an initiation term for the formation of the first pair, 10 helix propagation terms for the 10 possible nearest neighbour interactions (these are AA/UU; AU/UA; UA/AU; CA/GU; GU/CA; CU/GA; GA/CU; CG/GC; GC/CG and GG/CC. Note that for example GA/CU is equivalent to UC/AG, which you can see by Turning GA 180°) and other factors as well [28].

The following duplexes have the same nearest neighbours,

```
5'ACGCA3'          5'GCACG3'
3'UGCGU5'    and   3'CGUGC5'
```

However, they have different base compositions. The measured free energies of duplex formation are -4.97 and -6.17 kcal/mol, respectively [28]. Duplexes with terminal GC pairs are more stable than duplexes with the same nearest neighbours but with terminal AU pairs. Therefore, the free energy term for terminal AU pairs is unfavourable [28].

The free energy change for the formation of a duplex with only Watson-Crick pairs (in the nearest neighbour model of Xia et *al*) is calculated by [28]:

$$\Delta G\ (duplex) = \Delta G_{init} + \Sigma\ n_j\ \Delta G_j(NN) + m_{term\text{-}AU}\ \Delta G_{term\text{-}AU} + \Delta G_{sym}. \qquad (1)$$

The ΔG_{init} term is the free energy for the formation of the first pair. It is assumed that this free energy of initiation is independent of length, although this assumption is not completely correct [28]. Each $\Delta G_j(NN)$ term is the free energy contribution of the jth nearest neighbour with n_j occurrences in the sequence. $m_{\text{term-AU}}$ is the number of terminal AU pairs and $\Delta G_{\text{term-AU}}$ is the free energy parameter per terminal AU pair. The ΔG_{sym} is a symmetry term that is zero except when the duplex is self-complementary [28].

Parameters for the nearest neighbour model are obtained by applying multiple linear regression analyses of experimental data to the regression function of Equation (1). Parameters based on a set of 90 RNA duplexes with only Watson-Crick base pairs are listed in Table 1. On average, these parameters predict ΔG_{37} (total free energy at 37°C) of duplex formation within 3.2% [28].

Propagation of Watson-Crick nearset neighbors	△G
5'AA3' 3'UU5'	-0.93
5'AU3' 3'UA5'	-1.1
5'UA3' 3'AU5'	-1.33
5'CU3' 3'GA5'	-2.08
5'CA3' 3'GU5'	-2.11
5'GU3' 3'CA5'	-2.24
5'GA3' 3'CU5'	-2.35
5'CG3' 3'GC5'	-2.36
5'GG3' 3'CC5'	-3.26
5'GC3' 3'CG5'	-3.42
Propagation of GU-containing nearest neighbors	
5'GU3' 3'UG5'	1.29
5'GG3' 3'UU5'	0.47
5'UG3' 3'GU5'	0.3
5'AG3' 3'UU5'	-0.55
5'UG3' 3'AU5'	-1
5'GA3' 3'UU5'	-1.27
5'GU3' 3'UA5'	-1.36
5'CG3' 3'GU5'	-1.41
5'GG3' 3'CU5'	-1.53
5'GG3' 3'UC5'	-2.11
5'GC3' 3'UG5'	-2.51
Duplex parameters	
Initiation	4.09
Each terminal-AU or GU	0.45
Symmetry correction(only for self-complementary duplexes)	0.43

Table 1 **Thermodynamic parameters for Watson-Crick and GU base pairs (reproduced from [28])**

A simple comparison of the ΔG_{37} values (Table 1) of AU-only, one-AU/one-GC, and GC-only nearest-neighbours (1.1, 2.2, and 3.0 kcal/mol respectively) shows that base composition is an important factor [28].

In another instance, it has been shown that the ΔG_{37} for $\begin{smallmatrix}5'GC3'\\3'CG5'\end{smallmatrix}$ combination (-3.42 kcal/mol) is more favourable than for $\begin{smallmatrix}5'CG3'\\3'GC5'\end{smallmatrix}$ combination (-2.36 kcal/mol) by more than 1 kcal/mol. This observation suggests that orientations of base pairs and therefore stacking patterns are also important [28].

3-1-2 GU Pairs

The second most abundant motifs after Watson-Crick base pairs are GU pairs. The thermodynamics of GU pairs have been studied extensively. The results show that GU pairs are thermodynamically similar to AU pairs [26]. They have similar stabilities to AU base pairs. It has been observed that GU base pairs are conserved in RNA secondary structure. This observation leads us to the fact that GU pairs either have a structural role or functional role. For instance, in group I introns there is a helix (P1) that contains conserved GU pairs and it allows the formation of tertiary interactions with the intron's catalytic core [28].

To calculate the parameters of GU pairs in the nearest neighbour model, a database of duplexes containing single or tandem GU pairs is essential. These parameters are listed in Table 1. Similarly to terminal AU pairs, terminal GU pairs should be penalized (0.45 kcal/mol per terminal GU pair) for it has only two hydrogen bonds [28].

The parameters in Table 1 allow reasonable predictions of the thermodynamic properties of RNA duplexes with Watson-Crick and GU pairs.

3-1-3 Dangling Ends and Terminal Mismatches

Most of the helical regions in RNA have the average length of seven base pairs [26]. Unpaired nucleotides adjacent to helical regions have effects on stability of a helix. A 5′ **or**

3′ single unpaired nucleotide is called a dangling end, while a pair of 5′ **and** 3′ unpaired nucleotides at the same helix end is called a terminal mismatch [28].

One way to study the effects of dangling ends and terminal mismatches is to compare the stabilities of helices with and without them in model systems. Many of these parameters have been measured and the values are summarized in Table 2 and Table 3.

	X = A	*X = C*	*X = G*	*X = U*
3'-Dangling nucleotides				
CX G	-1.7	-0.8	-1.7	-1.2
GX C	-1.1	-0.4	-1.3	-0.6
RX U	-0.8	-0.5	-0.8	-0.6
UX R	-0.7	-0.1	-0.7	-0.1
5'-Dangling nucleotides				
XC G	-0.5	-0.3	-0.2	-0.1
XG C	-0.2	-0.3	0	0
XR U	-0.3	-0.3	-0.4	-0.2
XU R	-0.3	-0.1	-0.2	-0.2

Table 2 Thermodynamic parameters for unpaired dangling nucleotides (reproduced from [28])

Two sets of values are shown in Table 2 (R is A or G). Comparison of values for 3′ and 5′ dangling ends in this table indicates that 3′ terminal nucleotides effects are sequence dependent.

The difference between stabilizing effects of 5′ and 3′ dangling ends can be justified by structural considerations. In an A-form helix, a 5′ dangling end is not close to a base on the opposite strand, while a 3’ dangling end can stack directly on the base of the opposite strand of the terminal pair, which will help keeping the duplex together [28].

Some of the 3′ terminal nucleotides can stabilize a helix as much as a base pair while the 5′ terminal nucleotides have little effects on the stability of a duplex. Studies have proven the fact that 5′ dangling ends do not interact with its adjacent helix.

Base pair	X↓	Y→	A	C	G	U
5'AX3' 3'UY5'	A		-0.8	-1	-0.8	*
	C		-0.6	-0.7	*	-0.7
	G		-0.8	*	-0.8	*
	U		*	-0.8	*	-0.8
5'CX3' 3'GY5'	A		-1.5	-1.5	-1.4	*
	C		-1	-1.1	*	-0.8
	G		-1.4	*	-1.6	*
	U		*	-1.4	*	-1.2
5'GX3' 3'CY5'	A		-1.1	-1.5	-1.3	*
	C		-1.1	-0.7	*	-0.5
	G		-1.6	*	-1.4	*
	U		*	-1	*	-0.7
5'GX3' 3'UY5'	A		0.3	-1	-0.8	*
	C		-0.6	-0.7	*	-0.7
	G		0.6	*	-0.8	*
	U		*	-0.8	*	-0.8
5'UX3' 3'AY5'	A		-1	-0.8	-1.1	*
	C		-0.7	-0.6	*	-0.5
	G		-1.1	*	-1.2	*
	U		*	-0.6	*	-0.5
5'UX3' 3'GY5'	A		-1	-0.8	-1.1	*
	C		-0.7	-0.6	*	-0.5
	G		0.5	*	0.8	*
	U		*	-0.6	*	-0.5

Table 3 Thermodynamic parameters for terminal mismatches (reproduced from [28])

By adding the values for dangling ends or terminal mismatches from Table 3 or Table 3 to the duplex parameters predicted by the nearest neighbour model using Table 1, we can calculate the thermodynamic properties of duplexes with dangling ends or terminal mismatches [28].

3-1-4 Loops

RNA Loops are the regions of the sequence that are not involved in canonical pairs but are flanked by one or more canonical paired regions. Here a canonical pair is defined as a Watson-Crick or wobble GU pair. The types of loops commonly found in RNAs are shown in Figure 3. The thermodynamics of RNA loops have not been fully investigated due to their enormous sequence diversity. However, with more and more experimental data available, our understanding of their contributions increases and the models for approximating their stabilities are becoming more and more realistic.

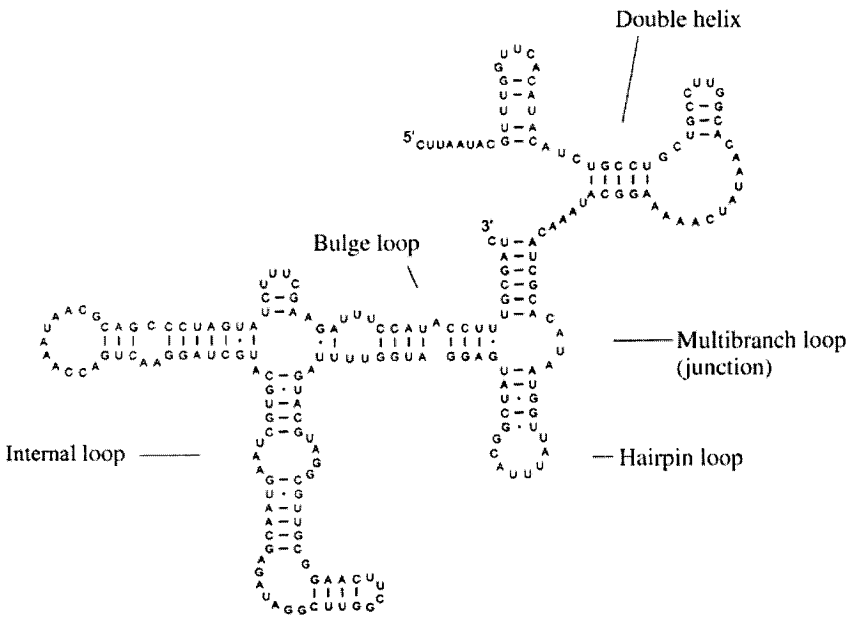


Figure 3 **Types of loops that are commonly found in RNAs (reproduced from [28])**

3-1-4-1 Hairpin Loops

Hairpin loops occur when nucleic acid strands fold back onto themselves to form base pairs. Hairpin loops occur frequently in RNA secondary structure. Nearly 70% of the small subunit rRNA of *Escherichia coli* consists of small hairpin loops [28]. Hairpins can provide nucleation sites for the overall three-dimensional folding, and be involved in tertiary interactions.

Hairpin loops in RNA can be very large. Alternatively, in at least one case, they can be as small as two nucleotides [28]. Tetraloops (hairpins with four nucleotides) are the predominant hairpin loops in the ribosomal RNAs, with GNRA and UNCG being the most common sequences, where N is any nucleotide and R is purine [28]. In general, the stabilities of hairpins depend on the stem, the first mismatch on the closing base pair, and the size of the loop. These factors have been studied extensively. The free energy of loop formation is obtained by measuring the free energy of forming the whole hairpin with the stem, and subtracting out the contribution of the stem. These are approximated by the parameters of the INN-HB model, including GU parameters (Table 1). The duplex initiation or symmetry term is not included [28]. Note that hairpin stems with an AU or GU base pair at either end of the stem are penalized by the terminal AU term. The free energy calculated for the hairpin loop ΔG_{HL} is assumed to be the sum of the following interactions [28]:

$$\Delta G_{HL} = \Delta G_{hairpin} - \Delta G_{stem} = \Delta G_{init}(n) + \Delta G \text{ (first mismatch)} + \Delta G \text{ (bonus/penalty)}. \quad (2)$$

Note that the duplex initiation is not included in determining the stem value because the initiation term is already included into the calculation of the free energy of the loop. The contributions of the first mismatches are approximated by the values for terminal mismatches (Table 3), except for loops smaller than four nucleotides, which are too constrained to allow the same stacking that is possible for terminal mismatches at the end of a duplex. The free energy for the hairpin loop initiation is unfavourable, due to the unfavourable entropy associated with constraining the nucleotides in the loop. The initiation values depend on the loop length ***n*** and are listed in Table 4.

Types of loop	*Number of nucleotides in loop*								
	1	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*
Hairpin loops	*	*	5.7	5.6	5.6	5.4	5.9	5.6	6.4
Bulge loops	3.8	2.8	3.2	3.6	4	4.4	*	*	*
Internal loops	*	*	*	1.7	1.8	2	*	*	*

Table 4 Free energy changes for the initiation of the three types of loops (reproduced from [28])

Several effects can be included in the ΔG*(bonus/penalty)* term. For example, Giese *et al* found that hairpins closed by GU have an enhanced stability of 2.1 kcal/mol if the G is directly preceded by two Gs [28].

Interestingly, hairpins closed by GU pairs are often preceded by two Gs in known secondary structures. Some tetraloops also have enhanced stability that can be added as part of the ΔG*(bonus/penalty)* term. The UNCG tetraloops are the most stable, but GNRA loops are also somewhat more stable than random sequence tetraloops [28].

Very small loops (fewer than 4 nucleotides) or large loops (more than eight nucleotides) are not predicted well by this model, and additional terms will need to be considered when more data become available [18].

3-1-4-2 Bulge Loops

Bulge loops have unpaired nucleotides on only one strand of a double helix (Figure 2 and Figure 3). They can be either extra-helical or intra-helical. In natural RNAs, one-nucleotide bulges are the most common bulge loops, and single purine bulges tend to stack in the helix and bend the helix, whereas single pyrimidine bulges are extra-helical [4]. Single bulged are important for protein binding and perhaps for tertiary folding as well [21]. Bulge loops of two or more nucleotides disrupt the nearest neighbour interactions of the adjacent base pairs and they become more destabilizing as the number of nucleotides in the bulge increases. Single nucleotide bulges do not interrupt nearest-neighbour stacking, while larger bulges interrupt stacking. Due to lack of data, the destabilizing effects of bulge loops are considered sequence independent. Free energy parameters for bulge loops of different sizes are given in Table 4.

3-1-4-3 Internal Loops

Internal loops are flanked by two helices, and contain nucleotides on both strands that are not in canonical pairs (Figure 2 and Figure 3). RNA internal loops may play important roles in tertiary interactions and in protein recognition [18]. Stabilities of internal loops are

dependent on the identity and orientation of the closing base pairs, on the sequence of the loop, and on the size and symmetry of the loop [15]. The contributions of the internal loops can be estimated by comparing the stabilities of duplexes with and without the loop [28]:

$$\Delta G(\text{internal loop}) = \Delta G(\text{duplex w/loop}) - \Delta G(\text{duplex w/o loop}) + \Delta G(NN), \quad (3)$$

where ΔG(NN) is the relevant nearest-neighbour parameter of the INN-HB model (Table 1). It is assumed that the loop does not affect the regions beyond the closing base pairs [28].

Single mismatches (1 x 1 internal loops)

The smallest internal loops in a helix are single mismatches. Only a few of them have been studied. Most of those that have been studied destabilize duplexes, but some are stabilizing as well.

Tandem mismatches (2 x 2 internal loops)

Tandem mismatches are formed when there are two opposing nucleotides on each strand that are not involved in Watson-Crick or GU pairs. These 2x2 internal loops can be symmetric or asymmetric in terms of loop sequence and closing base pairs [28]. Tandem mismatches are the most studied internal loops, and the results show that their stabilities are sequence dependent. Loops closed by GC base pairs are more stable than those closed by AU base pairs [28].

2 x 1 internal loops

The 2x1 internal loops contain two unpaired nucleotides opposing one nucleotide. Schroeder *et al* investigated internal loops closed by GC base pairs, and found considerable sequence dependence to the stabilities [28]. Loops with the potential for forming GA and UU mismatches generally have more favourable free energies of formation than loops without such potential [28].

3-1-4-4 Multibranch Loops

Multibranch loops or junctions are loops where more than two helices intersect. They usually also contain unpaired nucleotides, and are a major determinant of RNA tertiary structure. The stability of a multibranch loop depends on the number of helices involved, the number of unpaired nucleotides in the loop, and interactions like base triples near the loop.

3-2 Computational Approaches

Most folding programs fit into one or more of the four following classes:

1. "Basic" algorithms that predict hairpin and simple loop formation, but don't predict multi-branch loops and perform very basic energy minimization. The first algorithms written were of this type, and most have been updated or are no longer in use.
2. "Combinatorial" methods that generate lists of all possible secondary structure elements and put them together in all possible ways to find those with the lowest free energy.
3. "Recursive" algorithms that build the secondary structure, one nucleotide at a time while computing minimum energies along the way. Recursive algorithms are used in dynamic programming technique as well. It computes folding according to the low energy paths for generating secondary structure.
4. "Comparative sequence analysis" algorithms that find conserved structure for a set of sequences. It uses stochastic optimization on a set of uncertain solutions.

3-2-1 Free Energy Minimization

Most RNA secondary structure prediction algorithms perform thermodynamic optimization on a set of plausible structures in order to obtain the structure or structures with the lowest free energy. Thermodynamic principles suggest that the structure with the lowest free energy should be the most stable, and hence, the active fold. Unfortunately, not all of the factors that determine the energy of a fold are understood, and computational time limitations would make it infeasible to include all of such influences.

The free energy of a secondary structure is determined by summing the energy contributions of all base pairs, loops, hairpins, etc. Melting studies have shown that energy contributions are additive for short oligo-nucleotides. However, the free energies for longer RNA strands (>50-100 nucleotides) have not been determined experimentally.

Parameters for the contributions of individual secondary structure elements were determined by melting studies with short oligo-nucleotides [10]. For simple base-pairing energies, the individual nearest neighbour (INN) method has been used. Since then it has been updated to improve the accuracy of values [28]. The nearest neighbour model assumes that the thermodynamic stability of a base pair only depends on the identity of adjacent bases. Accordingly, the thermodynamic contributions from both base pairing and base stacking are considered. Thermodynamic properties were obtained by plotting melting data from short RNA duplexes (4-10 base pairs).

Thermodynamic parameters for loops were determined from additional melting data. The stability of hairpins, bulges, and other loops is largely dependent on four factors: (1) sequence of the loop, (2) nucleotides adjacent to and closing the loop, (3) nearby sequences not adjacent to the loop, and (4) size and shape of the loop [26].

The accuracy of the melting data when applied to very large loops is unknown. The thermodynamic parameters for hairpins come from melting studies of hairpins of only six

nucleotides [26], while hairpins over 50 nucleotides are sometimes predicted. The thermodynamic values obtained from these studies may provide sufficient approximations; however, better size- and sequence-based parameters would be extremely useful in determining large loops structures. It has been shown that very small changes in energy parameters often result in very large changes in predicted folding [30].

3-2-2 Combinatorial and Recursive Algorithms

Combinatorial and recursive folding algorithms are capable of finding minimum energy secondary structures. The combinatorial method forms structures by combining all possible helices in all possible ways, which means they predict a series of folds [8]. Unfortunately, these programs are extremely time consuming and they require a great amount of memory, since the number of potential folds increases exponentially with the length of the sequence. Most combinatorial programs are limited to folding sequences of length 200 or less. Furthermore, there are often too many predicted folds within a reasonable threshold to make any statistical or biological sense of them.

Recursive algorithms work in two steps. The first part, known as "fill", starts with small fragments (usually penta-nucleotides) and builds up to larger segments in a recursive way by iteratively minimizing the free energy. Ultimately, it computes and stores the minimum folding energies for all the fragments of the sequence. Next, the "traceback" computes a minimum energy structure by searching through the matrix of stored energies and combining compatible fragments. The building blocks or fragments of secondary structures are stacked base pairs, internal loops, bulges, and multi loops, rather than individual base pairs. The optimal/suboptimal free energy traceback must refine partial structures by exactly reversing the filling procedure used to generate structures from smaller fragments. Recursive algorithms can be much faster than combinatorial algorithms; the simplest algorithms determine the secondary structure in time proportional to the cube of the sequence length [31].

In their basic form, recursive methods generate only one optimal structure. There are a number of ways to overcome this limitation. One method is to take a standard recursive algorithm and set a threshold energy level in order to output all the structures with energies below that threshold. However, those suboptimal structures will be from the same region of the energy landscape. Moreover, if the threshold is set too low not much variation is possible, and if it is set too high, too many structures may be generated.

3-2-3 Comparative Sequence Analysis Algorithms

The folds of structural RNAs (e.g. tRNAs and rRNAs) are highly conserved among all kingdoms of life, and have been widely used to determine phylogenic relationships between different species [14]. Phylogenic-comparative analysis is one way to predict the fold of structural RNA. Most of these algorithms rely on the analysis of aligned nucleotide sequences to determine conserved regions of secondary structure. The foundation for these algorithms is the assumption that mutations that disrupt Watson-Crick base pairs have a negative effect. This negative effect may be overcome by a second compensatory mutation in the other half of the stem, which is restoring the base pair.

Briefly, comparative sequence analysis algorithms are seeking to find pairs of columns in a multiple sequence alignment that are statistically correlated. When a high quality alignment is available, this method is considered to be highly accurate. However, the construction of a truthful alignment requires knowledge about the structure. Comparative sequence analysis methods have proven to be notoriously difficult to automate and consequently are requiring extensive human intervention.

There is a trade-off between the number of sequences entered and the accuracy of the results. Inputting more sequences will yield fewer regions of conservation, but these regions will tend to be more accurate; inputting fewer sequences will give a greater number of conserved regions with lower accuracy. In general, the reliability of non-conserved regions is questionable.

3-2-4 A Genetic Algorithm with Energy Minimization

Chen and coworkers have developed a comparative sequence analysis algorithm that uses a different approach to find the common RNA secondary structures for a set of RNA sequences [6]. Genetic algorithms operate on a population of solutions, each of which has an encoded representation similar to the genetic material of an individual in nature. The solutions are modified by mutation (random changes) and crossover (recombination of features), and the modified solutions are selected by predefined criteria, such as the free energy here.

This method, unlike the previously discussed comparative sequence analysis methods, does not require an alignment to determine a common structure for a series of RNA sequences. The structural energy and structural similarity, both are considered among sequences of potential solutions. The free energy is modeled using the nearest neighbour principle, with penalties or bonuses for other secondary structure elements (since the focus is on structural similarity, the free energy rules are not as complex as those for recursive algorithms).

This approach has been very successful for determining conserved structures of tRNAs and small rRNA subunits. It can only be applied to phylogenically related sequences, as it is designed for choosing structures that satisfy conditions of conservation and thermodynamic stability [6]. Unfortunately, there is no proof that an optimal solution has been found.

3-2-5 Structure Optimization by Energy Minimization: The MFOLD Algorithm

As mentioned before, recursive algorithms find optimal structures for a single sequence while comparative sequence analysis finds common structures for a group of sequences. These programs rely on free energy minimization for determining the best structure(s), so the thermodynamic parameters used are very important [30].

Early recursive algorithms had many limitations. The thermodynamic parameters were not very accurate, nor were they always incorporated correctly into the algorithms. However, the slow speed of computers in the early 1980s was perhaps the biggest setback.

MFOLD is one of the recursive algorithms based on free energy minimization. It is capable of producing suboptimal structures, and it assumes that RNA thermodynamics has a linear dependence on the frequency of base pair [30].

Many recursive algorithms have a similar general structure. The accuracy of the predictions is determined by how algorithms handle thermodynamic parameters. Some algorithms incorporate information about the stacking of helices, additional sequence dependent information for bulges and so on. Such systems can potentially be more accurate.

3-2-6 Free energy Minimization and Comparative Sequence Analysis in One: The Dynalign Algorithm

The combination of comparative sequence analysis and free energy minimization is likely to improve secondary structure prediction. Dynalign is one of the recently developed algorithms for determining a common structure for two sequences [19]. It uses the thermodynamic parameters described above as estimates of the free energies [19].

Dynalign is a restricted implementation of an algorithm first introduced by Sankoff in 1985. Sankoff originally proposed a solution for the simultaneous alignment and RNA folding problems for N sequences. RNA molecules preserve their structure more than their sequence. Consequently, we can use information about aligned regions to limit the search for common secondary structure [23].

Dynalign takes two sequences as input and produces a sequence alignment and common structure are output. Base pairs are only permitted in the common structure if both sequences allow a base pair at the position, with one exception: a single inserted base pair may be included in one structure if it is between two conserved base pairs. Besides being

restricted to two input sequences, Dynalign also differs from Sankoff's algorithm in the following ways. First, contrarily to Sankoff's proposal, the objective function in Dynalign has no substitution term. Classical sequence alignment methods are seeking to find an alignment that minimises the weighted edit distance between two input sequences. As such, their objective function includes a contribution for each aligned symbols (nucleotides or amino acid). The magnitude of the contribution depends on the identity the aligned symbols. Some substitutions are favourable others are not. This paradigm is particularly relevant for the alignment of DNA and protein sequences. However, in the case of RNA, the preservation of base pairs, at least in the helical regions, seems to be more important than preserving the identity of the nucleotides. Therefore, Dynalign does not directly take substitutions into account. This means that the alignment of nucleotides in the unpaired regions is not well defined. Secondly, the set of recurrence equations proposed by Sankoff models the insertion/deletion of substructures *in toto*. Perhaps because of the lack of data for accurately modelling these events or simply to reduce the complexity the recurrence equations, no specific cases exist for the insertion/deletion of entire substructures. Thirdly, Dynalign limits the maximum distance between aligned nucleotides to be at most M. This constraint considerably reduces the runtime of the alignment. In fact, M has to be less than 15 or 20 nucleotides otherwise the execution time is not acceptable. This limits the application of the algorithm to sequences that are approximately of the same length and can be aligned with few insertions/deletions. Still there are many interesting families of RNAs that match those criteria. In practice, since M has to be small, the deletion/insertion of substructure *in toto* would be severely restricted. Therefore, this is maybe yet another reason for not including it the model.

The algorithm is a four-dimensional dynamic program divided into fill and traceback steps. The fill step calculates three free energy arrays, *W(i,j,k,l)*, *V(i,j,k,l)*, and *W5(i,k)*.
W(i,j,k,l) is the sum of the minimum free energies for nucleotide fragments i to j from the first sequence and k to l from the second sequence with i aligned to k and j aligned to l.
V(i,j,k,l) is the same as *W(i,j,k,l)*, except that i is base-paired with j, and k is based-paired with l [19]. *W5(i,k)* is the sum of free energies of nucleotide fragments from 1 to i in the

first sequence and 1 to *k* in the second. Consequently, *W5(N1,N2)* (where *N1* and *N2* are the lengths of the sequences 1 and 2, respectively) is the lowest free energy sum for a structure common to both sequences. The traceback step uses the information in the energy arrays to find the structure that has the lowest free energy.

CHAPTER 4

X-DYNALIGN

IN THIS CHAPTER

- X-Dynalign

CHAPTER FOUR

We have introduced RNA molecule and ways to predict its secondary structure in the previous three chapters. We mentioned well-established experimental techniques for determining RNA structure. These techniques play important roles in providing complementary information. As mentioned, the most accurate computational method for determining the common structure for two RNA sequences so far is the Dynalign algorithm. Herein, we validate the claim that using three input sequences, rather than two, will improve the quality of the predictions. In this chapter, we present a newly developed algorithm, called X-Dynalign, and its implementation. The results presented here has been published in [16] and [17].

4-1 X-Dynalign: RNA Secondary Structure Prediction

The thermodynamic parameters are crucial for RNA secondary structure predictions because of the large number of possible pairings [10]. Recursive (dynamic programming) algorithms use predicted free energies from smaller fragments to predict the free energies for larger fragments. This recursion continues until the free energy for the whole structure is calculated. Currently, little is known about the thermodynamics of knotted structures [27], and many other motifs require approximations, as discussed before.

4-1-1 Algorithm

X-Dynalign (eXtended Dynalign) is a direct extension of Dynalign that takes three sequences as input and produces a sequence alignment and a common structure. The total free energy for the common structure, ΔG_{total}, is the sum of the free energy of each sequence (given the common structure), plus gap penalties:

$$\Delta G_{total} = \Delta G_{sequence1} + \Delta G_{sequence2} + \Delta G_{sequence3} + (\#\ of\ gaps)\ \Delta G_{gap}.$$

X-Dynalign seeks to find a structure minimizing ΔG_{total}.

If all three sequences can adopt a canonical pair at the same pair of positions in the alignment, there will be base pair in the common structure at that pair of positions. A gap is inserted into a sequence if that sequence has no analogous nucleotide in other sequence(s). Gaps represent the deletion or insertion of one or more nucleotide into a sequence. The value of ΔG_{gap} is determined empirically, those experiments are presented in Section 5-1-1. $\Delta G_{sequence1}$, $\Delta G_{sequence2}$ and $\Delta G_{sequence3}$ represent the conformational free energy for each input sequence when folded onto the common secondary structure, according to the nearest neighbour model.

The original idea was first suggested by Sankoff in 1985[23]. The proposed algorithm was formulated for N input sequences. Dynalign is an implementation of Sankoff's algorithm for two sequences. Here, we increase the number of inputs to three sequences.

4-1-2 Detailed Algorithm

X-Dynalign is a six-dimensional dynamic programming algorithm. The calculation is divided into two steps: the fill step and the traceback. During the fill step all the values for each array of free energy are calculated. At the end of this step, we have the values of the minimum free energy for the common structure. To obtain the common structure itself and the alignment, we simply traceback the matrices. Three sets of recurrence equations define the objective function: W, V and $W9$.

The maximum distance between aligned nucleotides, M, is a parameter to reduce the number of operations. The value of the parameter is set by the user. It should be at least greater than the length difference of sequences, or in most cases larger than the difference. The following relationships always hold for $W(i,j,k,l,m,n)$ and $V(i,j,k,l,m,n)$:

$i-M \leq k \leq i+M$; $j-M \leq l \leq j+M$; $k-M \leq m \leq k+M$ and $l-M \leq n \leq l+M$.

The fill step calculates three arrays of free energies: $W(i,j,k,l,m,n)$, $V(i,j,k,l,m,n)$, and $W9(i,j,k,l,m,n)$. $W(i,j,k,l,m,n)$ represents the minimum free energies for nucleotide fragments i to j from S_1, k to l from S_2 and m to n from S_3, with i aligned to k, and m; and j aligned to l and n plus any gap penalties for interior nucleotides.

Here is the pseudo code for the initialization of W:

```
for (i=0 to number of bases of first sequence) {
   for (j=0 to number of bases of first sequence) {
      for (k=0 to 2M) {
         for (l=0 to 2M) {
            for (m=0 to 2M) {
               for (n=0 to 2M) {
                  w[i][j][k][l][m][n] = infinity;
               }
            }
         }
      }
}
```

$V(i,j,k,l,m,n)$ is defined similarly to $W(i,j,k,l,m,n)$, except that i and j are base-paired, k and l are base-paired, and m and n are base-paired. Space allocation and initialization for V is exactly the same as for W.

$W9(i,k,m)$ is the lowest free energy sum for the prefix alignment of the nucleotide fragments from 1 to i in the first sequence, 1 to k in the second sequence, and 1 to m in the third sequence.

```
for (i=0 to number of bases of first sequence) {
   for (k=0 to 2M) {
      for (m=0 to 2M) {
         W9[i][k][m] = egap*number of gaps;
      }
   }
}
```

where egap :: ΔG_{gap}

The traceback step uses values stored in *W9* matrix to find the structure common to the three sequences that has the lowest free energy sum.

Free energies are rounded to the nearest tenth of a kcal/mol and are multiplied by ten for storage into the arrays. This allows integer arithmetic and therefore *V(i,j,k,l,m,n)*, *W(i,j,k,l,m,n)*, and *W9(i,k,m)* are arrays of short integers, which also helps saving space.
V and *W* are filled by considering every segment of 5 bases (the minimum sequence length that allows unimolecular secondary structure without base-pair distortion), 6 bases, 7 bases, and so forth up to length of *N*. The pseudo code below illustrates the fill step.

```
for (i=1 to number of bases of first sequence) {
   for (j=i+minloop to number of bases of first sequence) {
      for (k=i+M downto i-M) {
         for (l=j-M to j+M) {
            for (m=k+M downto k-M) {
               for (n=l-M to l+M) {
                  fill V
                  fill W
               }
            }
         }
      }
   }
}
minloop: The minimum substructure with a base pair possible = 3
```

If *i:j*, *k:l* or *m:n* is a non-canonical base-pair, then *V* is set to a large positive free energy value (infinity). If all three pairs *i:j* and *k:l* and *m:n* can form canonical base-pairs, A:U, G:C, or G:U, then *V* is the minimum of three terms, V_1, V_2, and V_3:

$V = min\ (V_1, V_2, V_3)$

V_1 considers hairpin loops closed by base-pairs *i:j*, *k:l* and *m:n*:

$V_1 = \Delta G_{hairpin}\ (i, j) + \Delta G_{hairpin}\ (k, l) + \Delta G_{hairpin}\ (m, n) + (no.\ of gaps)\ \Delta G_{gap}$

```
v[j][i][k][l][m][n] =
ehairpin(i,j)+ ehairpin(k,l)+ ehairpin(m,n)+ egap*number of gaps;
```

V_2 is the lowest sum of the free energies for a helix extension, bulge loop, or internal loop in the common structure. V_2 requires a search through parameters *i', j', k', l', m', n'* so that:

$V_2 = min\ (V(i', j', k', l', m', n') + \Delta G_{motif1} + \Delta G_{motif2} + \Delta G_{motif3})$.

For $i' = i+1$ and $j' = j-1$, this is a continuation of a canonical helix. For either $i' = i+1$ or $j' = j-1$, but not both, this motif is a bulge loop. Otherwise, this is an internal loop. The same applies for k', l' from second sequence and m', n' from third sequence. *S* is a user defined constraint limiting the size of internal loops.

```
for (c=i+1 to i+S) {
   for (d=j-1 downto j-S) {
      for (e=k+1 to k+S) {
         for (f=l-1 downto l-S) {
            for (g=m+1 to m+S) {
               for (h=n-1 downto n-S) {

                  // Cases for Sequence 1

                  // helical stacking
                  if (c==i+1 && d==j-1) {
                        energy = ebasepair(i,j,i+1,j-1);
                  }
                  // base pair insertion
                  else if (single bp insertion is enabled &&
                           i+1 and j-1 are base paired &&
                           i+2 and j-2 are base paired &&
                           (k+1 and l-1 are base paired ||
                            m+1 and n-1 are base paired)) {
                        energy = ebasepair(i,j,i+1,j-1) +
                                 ebasepair(i+1,j-1,i+2,j-2);
                  // internal loop
```

```
                    else {
                    energy = einternal(i,j,c,d);
                    }
                    // cases for sequence 2
                       . . .
                       . . .
                    // cases for sequence 3
                }
            }
        }
    }
}
```

where: $i < i' < j' < j$, $k < k' < l' < l$ and $m < m' < n' < n$. Furthermore, the following constraints must hold: $i'-M \leq k' \leq i'+M$, $j'-M \leq l' \leq j'+M$, $k'-M \leq m' \leq k'+M$ and $l'-M \leq n' \leq l'+M$. The search is restricted to the region of: $i < i' \leq i+S$, $j-S \leq j' < j$, $k < k' \leq k+S$, $l-S \leq l' < l$, $m < m' \leq m+S$, $n-S \leq n' < n$. For our tests S was set to 20. This is a commonly used technique to limit the size of internal loop searching. ΔG_{motif1} is the free energy of the motif closed by pairs $i:j$ and $i':j'$; ΔG_{motif2} and ΔG_{motif3} are similarly the energy for the motif in sequence 2 and 3 respectively.

V_3 is the lowest sum of free energies for a multibranch loop closed by pairs $i:j$, $k:l$ and $m:n$. V_3 requires a search through parameters i', k' and m', with $i < i' < j$, $k < k' < l$, $m < m' < n$ and is the minimum of 64 terms. The 64 cases account for all combinations of whether or not $i+1$ and $j-1$ are dangling ends on the pair $i:j$, whether $k+1$ or $l-1$ are dangling ends on the pair $k:l$ and whether $m+1$ or $n-1$ are dangling ends on the pair $m:n$. See Appendix A for further details.

$V_3 = W(a, i', b, k', c, m') + W(i'+1, d, k'+1, e, m'+1, f) + x\ \Delta G_{unpaired\ nucleotides\ in\ MBL\ loop} + 3\ \Delta G_{MBL\ closure} + 3\ \Delta G_{helix\ terminating\ in\ MBL} + \Delta G(dangling\ ends) + y\ \Delta G_{gap}$.

```
for (c=i+minloop to j-minloop) {
   for (d=k+minloop to l-minloop) {
      e = d-c+M;
      for (f=m+minloop to n-minloop) {
         g = f-d+M;

         //case 1
         energy = min(energy,w[i+1][c][k+1][e][m+1][g]+
                      w[c+1][j-1][e][l-1][g][n-1]+
                      3*eMBLclosure+3*ehelix_termin_MBL);
```

```
          //case 2
            .
            .
            .
          //case 63
          //case 64
          energy = min(energy,w[i+2][c][k+2][e][m+2][g]+
                       w[j-2][c+1][e][b][g][bb]+
                       3*eMBLclosure+3*ehelix_termin_MBL+
                       6*eunpaired_nuc_MBL+
                       edangle5(j,i,j-1)+edangle3(i,j,i+1)
                       edangle5(l,k,l-1)+edangle3(k,l,k+1)+
                       edangle5(n,m,n-1)+edangle3(m,n,m+1));
       }
    }
}
```

W is the minimum of three terms, W_1, W_2, and W_3:

$W = min\ (W_1, W_2, W_3)$

W_1 represents adding unpaired nucleotides to a multibranch loop. Similarly to V_3, there are 64 ways for adding nucleotides as it is listed below (a is either i or $i+1$, b is either j or $j-1$, and so on). The $\Delta G_{unpaired\ nucleotide\ in\ MBL\ loop}$ is multiplied by the number of nucleotides added, x. The gap penalty is multiplied by y, the number of nucleotides that are added in one sequence, but not added in the other two sequences (See Appendix A).

$W_1 = W(a, b, c, d, e, f) + x\ \Delta G_{unpaired\ nucleotides\ in\ MBL\ loop} + y\ \Delta G_{gap}$

$a=i$ or $i+1$

$b=j$ or $j-1$

$c=k$ or $k+1$

$d=l$ or $l-1$

$e=m$ or $m+1$

$f=n$ or $n-1$

```
//case 2
energy = min(energy,w[i][j][k][l][m][n-1]+eunpaired_nuc_MBL+2*egap);
    .
    .
    .
//case 64
energy = min(energy,w[i+1][j-1][k+1][l-1][m+1][n-1]+
             6*eunpaired_nuc_MBL);
```

W_2 accounts for helix termini. The terms a, b, c, d, e, f, x, and y are defined similarly to W_1. In this case, the variation in values for a, b, c, d, e, and f allows for dangling ends at helix termini and the favourable stability for each dangling end is added to the term.

$W_2 = V(a, b, c, d, e, f) + x\,\Delta G_{unpaired\ nucleotides\ in\ MBL\ loop} + y\,\Delta G_{gap} + 3\,\Delta G_{MBL\ closure}$

```
//case 1
energy = v[i][j][k][l][m][n]+3*eMBLclosure

//case 2
energy = min(energy,v[i][j][k][l][m][n-1]+3*eMBLclosure+
             edangle3(n-1,m,n,)+eunpaired_nuc_MBL+2*egap);
    .
    .
    .
//case 64
energy = min(energy,v[i+1][j-1][k+1][l-1][m+1][n-1]+3*eMBLclosure+
             edangle5(i+1,j-1,i)+edangle3(j-1,i+1,j)+
             edangle5(k+1,l-1,k)+edangle3(l-1,k+1,l)+
             edangle5(m+1,n-1,m)+edangle3(n-1,m+1,n)+
             6*eunpaired_nuc_MBL);
```

W_3 accounts for bifurcations in the structure. This term is necessary for considering multibranch loops with more than two branching helices. A search is conducted through $i < i' < j$, $k < k' < l$ and $m < m' < n$. The number of iterations is limited to $i'\text{-}M \le k' \le i'\text{+}M$ and $k'\text{-}M \le m' \le k'\text{+}M$.

$W_3 = min\ (\ W(i, i', k, k', m, m') + W(i'+1, j, k'+1, l, m'+1, n)\ )$

$i < i' < j$

$k < k' < l$

$m < m' < n$

$i'\text{-}M \le k' \le i'\text{+}M$

$k'\text{-}M \le m' \le k'\text{+}M$

```
for (c=i+minloop to j-minloop) {
   for (d=k+minloop to l-minloop) {
      e = d-c+maxsep;
      for (f=m+minloop to n-minloop) {
         g = f-d+maxsep;
         energy = min(energy,w[i][c][k][e][m][g]+
                        w[c+1][j][e+1][l][g+1][n]);
      }
   }
}
```

W9 is the minimum of 8 terms $W9 = min\ (W9_1, W9_2, ..., W9_8)$ where:

$W9_1 = V(i', d, k', e, m', f) + W9(a, b, c) + x\ \Delta G_{unpaired\ nucleotide\ in\ MBL\ loop} + 3\ \Delta G_{MBL\ closure} + y\ \Delta G_{gap} + \Delta G(dangling\ ends)$

$W9_2 = W9(i, k, m-1) + 2\ \Delta Ggap,$

$W9_3 = W9(i, k-1, m) + 2\ \Delta Ggap,$

$W9_4 = W9(i, k-1, m-1) + \Delta Ggap,$

$W9_5 = W9(i-1, k, m) + 2\ \Delta Ggap,$

$W9_6 = W9(i-1, k, m-1) + \Delta Ggap,$

$W9_7 = W9(i-1, k-1, m) + \Delta Ggap,$

$W9_8 = W9(i-1, k-1, m-1) + \Delta Ggap,$

where $i' < i$, $k' < k$, and $m' < m$, $a = i'-1$ or $i'-2$, $b = k'-1$ or $k'-2$, $c = m'-1$ or $m'-2$, and $d = i$ or $i-1$, $e = k$ or $k-1$, and $f = m$ or $m-1$. Similarly to W_2 and V_3, there are 64 possible cases to consider for $W9_1$ allowing for dangling ends on the helices closed by $V(i', d, k', e, m', f)$. The rest of the equations from $W9_1$ to $W9_8$ are used for the global alignment. Each represents an insertion, a deletion or a match state for the global alignment of three input sequences.

With *W(i,j,k,l,m,n)*, *V(i,j,k,l,m,n)*, and *W9(i,k,m)* defined as above, a traceback procedure as shown in Figure 4 and Figure 5 is used to find the structure that corresponds the optimal solution.

The traceback has two parts. The first part consists of finding the exterior base pairs in the lowest free energy structure. Exterior base pairs are those base pairs that close a domain. The second part is the construction of the structure. It means tracing back the base pairs interior to those found in first step of traceback.

```
for (i=1 to number of bases of sequence 1) {
   for (k=0 to 2M) {
      for(m=0 to 2M) {

         find i_lowest, k_lowest, and m_lowest;
      }
   }
}
traceback W9 to find exterior base pairs (mirror of fill W9)
traceback V and W to find interior base pairs (mirror of fill V and W)
```

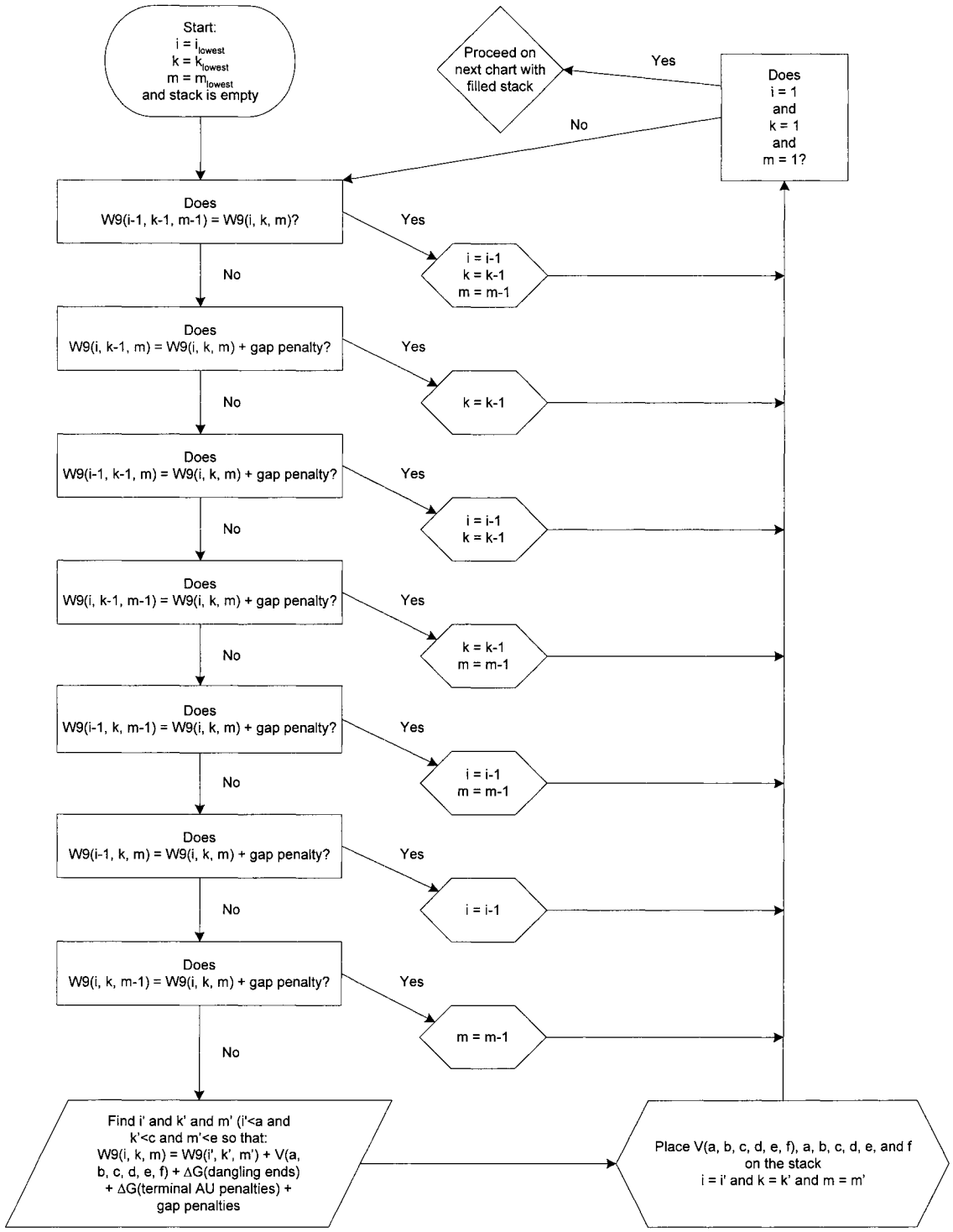


Figure 4 Traceback flowchart part 1, to find the exterior base pairs, those base pairs that close a domain (adapted for three sequences from [19])

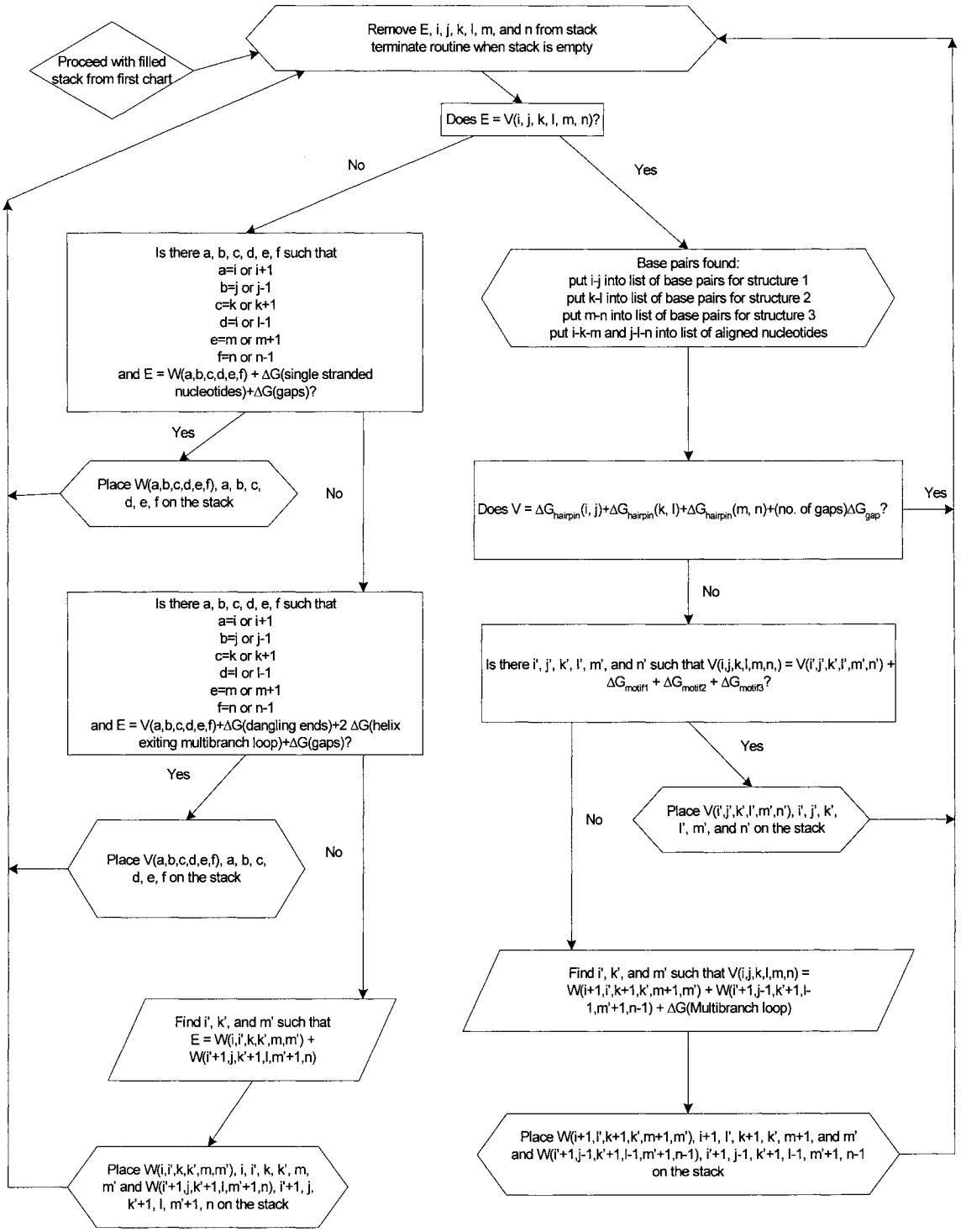


Figure 5 Traceback flowchart part 2, to trace the base pairs interior to the base pairs found in part 1 (adapted for three sequences from [19])

4-1-3 Complexity Analysis

As mentioned before, there are three matrices to be filled: *V*, *W*, and *W9*. *V* and *W* matrices are 6-dimensional:

```
for (i=0 to number of bases of first sequence)
       for (j=0 to number of bases of first sequence)
          for (k=0 to 2M)
             for (l=0 to 2M)
                for (m=0 to 2M)
                   for (n=0 to 2M)
                      w[i][j][k][l][m][n] = infinity;
                      v[i][j][k][l][m][n] = infinity;
```

hence the memory use for each of them is:

$N \times N \times 2M \times 2M \times 2M \times 2M$,

where N is the number of nucleotides in the sequence 1 (the shortest of all three sequences) and M is the maximum separation.

The memory use for *W9*, which is a 3-dimensional matrix, is:

$N \times 2M \times 2M$.

By comparison to *V* and *W*, the memory use of *W9* is very small. Therefore the memory use for the algorithm is:

$16 N^2 \times M^4 + 16 N^2 \times M^4 = 32 N^2 \times M^4 \sim O(N^2 M^4)$.

As for the time complexity, there are six main outer for-loops for filling the matrices *V* and *W*.

```
for (i=1 to number of bases of first sequence)
   for (j=i+minloop to number of bases of first sequence)
      for (k=i+M downto i-M)
         for (l=j-M to j+M)
            for (m=k+M downto k-M)
               for (n=l-M to l+M)
                  fill V
                  fill W
```

For filling V_1 and V_2 there are no more "for"-loops other than those defined above so the time complexity is:

$N^2 \times M^4$.

For V_3, because there are three more "for"-loops inside those six outer for-loops (see pseudo code for filling V_3 in Section 4-1-2), the time complexity is:

$N^2 \times M^4 \times N^1 \times M^2 = N^3 . M^6$.

The same holds true for filling W_1, W_2, and W_3 and therefore the time complexity for the whole algorithm is order of: $O(N^3 M^6)$.

CHAPTER 5

RESULTS

IN THIS CHAPTER

- Performance Measures
- Experiments
- Results

CHAPTER FIVE

To validate the hypothesis that RNA secondary structure prediction can be improved by finding the lowest free energy structure common to three sequences rather than two sequences, two classes of RNA with known secondary structures were studied: tRNA and 5S rRNA.

5-1 Performance Measures

In order to compare the predicted structures from X-Dynalign to the observed structures for each sequence, we need a reference. We call reference, the secondary structure that is obtained from a curated database. For each sequence in a structural alignment, the two secondary structures (reference and predicted) can be compared in the following ways by counting:

- The number of pairs for which both structures have base pairs between the same positions (True Positives TP).
- The number of pairs for which the prediction has base pairs and the reference does not (False Positives FP).
- The number of pairs for which the prediction does not have base pair but the reference does (False Negatives FN).

Base offsets are not allowed.

Three indices have been considered to measure the performance of the algorithm: sensitivity, PPV, and MCC.

Sensitivity (coverage) is the fraction of the base pairs of the reference structure that are predicted correctly:
Sensitivity = TP / (TP + FN).

PPV, or Positive Predictive Value (specificity or accuracy) is the number of predicted base pairs that are also in the reference structure:
PPV = TP / (TP + FP).

Finally, the Matthews' Correlation Coefficient (MCC) is a commonly used measure in bioinformatics. We used the approximation proposed by Gorodkin, Stricklin and Stormo [13]:

$$MCC = \sqrt{[TP / (TP + FN)] \times [TP / (TP + FP)]}$$

5-2 Experiments

The input sequences for our experiments were selected based on the following criteria:

1. Sequences are selected in a way that they can be aligned with a small value of M ($M \leq 6$).
2. The input sequences are filtered so that the maximum pairwise identity is less than 90%.
3. Selected sequences have length of 200 nt or less.
4. We selected sequences that are considered challenging cases for MFOLD – a widely used computer program for determination of RNA secondary structure using a single sequence as input.
5. For each triple of our inputs, we required that there is at least one pair that has less than 80% accuracy with Dynalign.

We used sequences from [1] and [2]. As our first dataset, we selected 10 tRNA sequences out of 13 from Dynalign paper [19]. The pairwise sequence identity of this dataset varies from 27.3 to 68.8 %. The second dataset is a set of 13 5S rRNA sequences taken from the comparative RNA web sites [1] and [2]. The pairwise sequence identity of this dataset varies from 47.2 to 88.2 %. Table 5 and Table 6 present the input RNA sequences used for our experiments.

id	Length	Description
RD0260	77	Asp Phage T5 (Virus)
RD0500	76	Asp *Haloferax volcanii* (Archae)
RD4800	71	Asp *Aedes albopictus* (Mitochondria, Animal)
RE2140	76	Glu *Synechocystis sp.* (Eubacteria)
RE6781	76	Glu *Hordeum vulgare* (Chloroplast)
RF6320	76	Phe *Schizosaccharomyces pombe* (Cytoplasm, Fungi)
RL0503	88	Leu *Haloferax volcanii* (Archae)
RL1141	89	Leu *Mycoplasma capricolum* (Eubacteria)
RS0380	88	Ser *Halobacterium cutirubrum* (Archae)
RS1141	92	Ser *Mycoplasma capricolum* (Eubacteria)

Table 5 tRNA dataset

id	Length	Description
AJ131594	117	*Delftia acidovorans*
AJ251080	117	*Geobacillus stearothermophilus*
K02682	120	*Micrococcus luteus*
M10816	119	*Geobacillus stearothermophilus*
M16532	121	*Thermus sp.*
M25591	117	*Geobacillus stearothermophilus*
V00336	120	*Escherichia coli*
X02024	119	*Sporosarcina pasteurii*
X02627	120	*Agrobacterium tumefaciens*
X04585	119	*Rhodobacter capsulatus*
X08000	122	*Arthrobacter oxydans*
X08002	122	*Arthrobacter globiformis*

Table 6 5 S rRNA dataset

The following hypotheses are tested [16]:

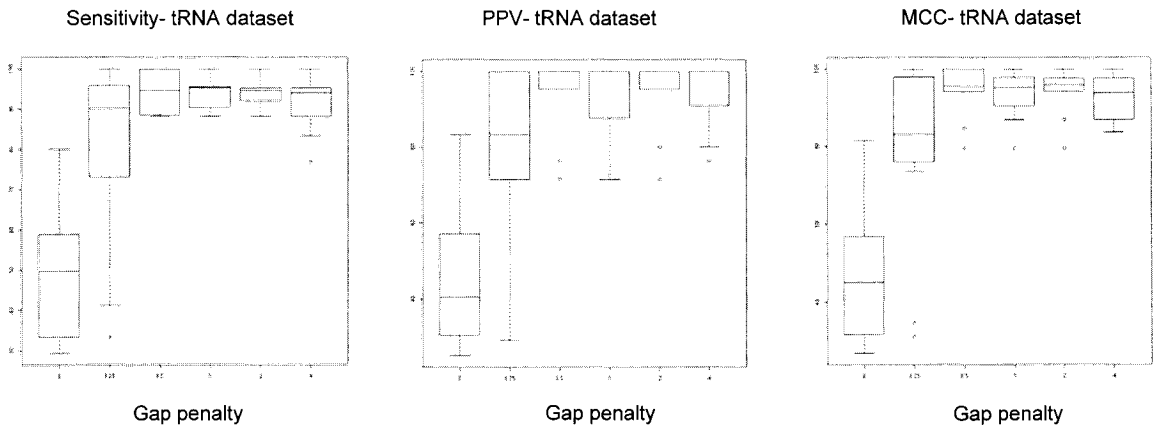
1. Using three input sequences — instead of two — should improve the average accuracy. The more the input sequences, the less likely they will all fold into a bad minimum free energy.
2. The worst prediction should be more accurate when three input sequences are used rather than two.
3. On the other hand, the common structure of three sequences should be less representative of the structure of each individual sequence, as a result the average coverage should be less than that of Dynalign.

5-2-1 Calibrating Gap Penalties

Dynalign experiments showed that the optimal gap penalty depends on the type of RNA. Specifically, the optimal gap penalties were 2.0 kcal/mol for tRNA and 0.4 kcal/mol for 5S rRNA. Accordingly, we performed two sets of experiments to measure the effect of various gap penalties on the performance measures. Since these experiments are time consuming, only six gap penalty scores were tested: 0.0, 0.25, 0.5, 1.0, 2.0, and 4.0. Also, triples that can be aligned with a small value of M, here 5, were selected. In total, 105 predictions were made for tRNA, and 90 predictions were made for 5S rRNA. The box plots show that all three performance measures vary greatly for gap penalties less than 0.5; the effect is less on the 5S entries, see Figure 6 and Figure 7. The optimal positive predictive value and sensitivity seem to occur for the same gap penalty, 1.0 kcal/mol. Furthermore, the optimal value is the same for both types, tRNA and 5S rRNA [17]. This presents another advantage of using three sequences (instead of two), which suggests that the optimal value of gap penalty is less dependent on the type of sequences when compared to Dynalign.

Although the calibration experiments indicate that a gap penalty of 1.0 kcal/mol should be used for both tRNA and 5S rRNA, the results presented in next section are all based on a gap penalty value of 2.0 kcal/mol. The picking of the gap penalty (2.0 kcal/mol) was done on the basis of the results obtained by Mathews and Turner using Dynalign, prior to performing the calibration experiments. X-Dynalign runs in $O(N^3 M^6)$. Long sequences,

with large M, are taking many days, sometimes weeks to run. Together, the experiments presented in the next section are representing several months of calculations, and therefore, we did not repeat the experiments. First, the impact of this choice on the results should be small. Indeed, Figure 6 and Figure 7 are clearly showing a large plateau with small variance for gap penalties in the range 1.0 to 4.0 kcal/mol. Second, for the tRNA dataset, this value actually corresponds to the maximum accuracy.



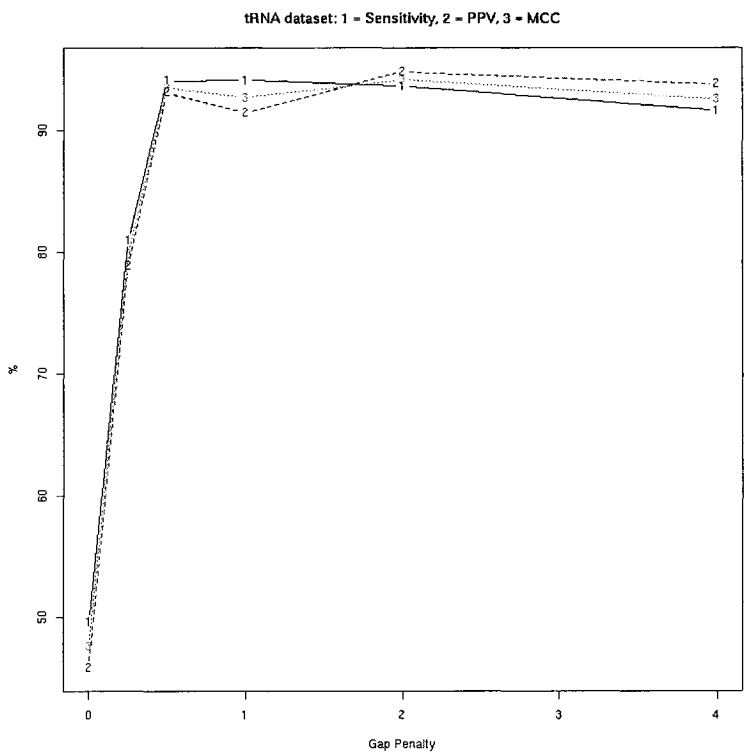
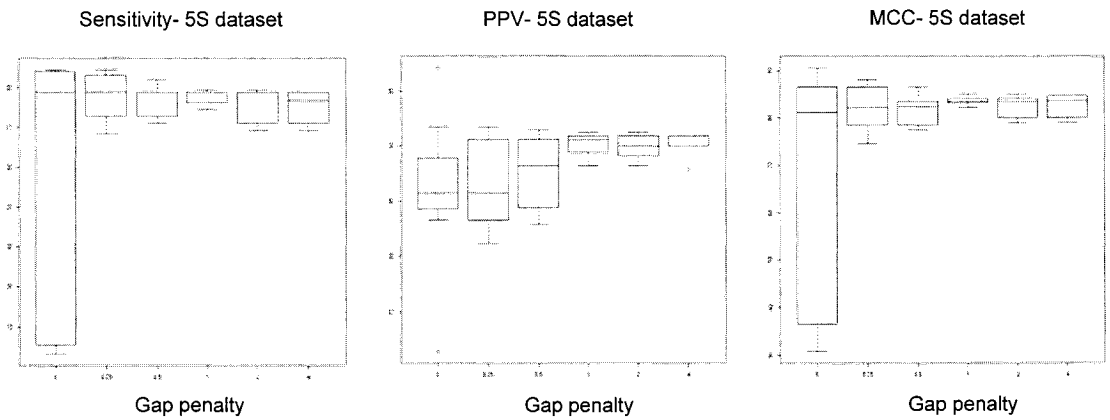


Figure 6 **Effect of various gap penalty scores on sensitivity, PPV and MCC for tRNA dataset**



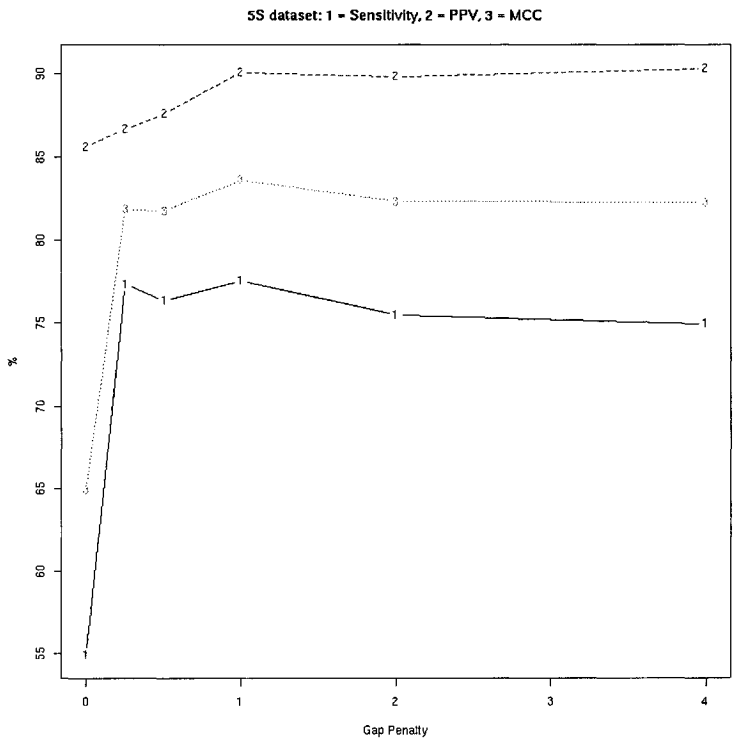


Figure 7 **Effect of various gap penalty scores on sensitivity, PPV and MCC for 5S rRNA dataset**

5-3 Results

First, we investigated the possibility that X-Dynalign could degrade the accuracy of Dynalign predictions. We made 15 runs, 45 predictions, using X-Dynalign. There is no single incidence of lower accuracy; however the sensitivity (coverage) is degraded slightly for 4 out of 15 predictions as presented in Table 7.

	X-Dynalign		M	sensitivity	PPV	MCC	Dynalign		M	sensitivity	PPV	MCC
1	RD0500_RD0260_RD1140	RD0500	3	95.2	100	97.6	RD0500_RD0260	RD0500	3	95.2	100	97.6
	RD0500_RD0260_RD1140	RD0260		95.2	100	97.6	RD0500_RD0260	RD0260		95.2	100	97.6
	RD0500_RD0260_RD1140	RD1140		95.2	100	97.6	RD0500_RD1140	RD0500		95.2	100	97.6
							RD0500_RD1140	RD1140		95.2	100	97.6
							RD0260_RD1140	RD0260		100	100	100
							RD0260_RD1140	RD1140		100	100	100
2	RD0500_RD2640_RD1140	RD0500	3	95.2	100	97.6	RD0500_RD2640	RD0500	3	95.2	100	97.6
	RD0500_RD2640_RD1140	RD2640		95.2	100	97.6	RD0500_RD2640	RD2640		95.2	100	97.6
	RD0500_RD2640_RD1140	RD1140		95.2	100	97.6	RD0500_RD1140	RD0500		95.2	100	97.6
							RD0500_RD1140	RD1140		95.2	100	97.6
							RD2640_RD1140	RD2640		100	100	100
							RD2640_RD1140	RD1140		100	100	100
3	RE2140_RD0500_RD0260	RE2140	3	95.2	100	97.6	RE2140_RD0500	RE2140	3	95.2	100	97.6
	RE2140_RD0500_RD0260	RD0500		95.2	100	97.6	RE2140_RD0500	RD0500		95.2	100	97.6
	RE2140_RD0500_RD0260	RD0260		95.2	100	97.6	RE2140_RD0260	RE2140		100	100	100
							RE2140_RD0260	RD0260		100	100	100
							RD0500_RD0260	RD0500		95.2	100	97.6
							RD0500_RD0260	RD0260		95.2	100	97.6
4	RD0500_RE2140_RD2640	RD0500	3	95.2	100	97.6	RD0500_RE2140	RD0500	3	95.2	100	97.6
	RD0500_RE2140_RD2640	RE2140		95.2	100	97.6	RD0500_RE2140	RE2140		95.2	100	97.6
	RD0500_RE2140_RD2640	RD2640		95.2	100	97.6	RD0500_RD2640	RD0500		95.2	100	97.6
							RD0500_RD2640	RD2640		95.2	100	97.6
							RE2140_RD2640	RE2140		100	100	100
							RE2140_RD2640	RD2640		100	100	100

Table 7 **X-Dynalign lower sensitivity vs Dynalign predictions (Gap penalty =2)**

The other 11 runs, presented in Table 8, show no degradation in accuracy or coverage. For this set of experiments, we carefully selected sequences from Dynalign experiments such that they all generate good predictions.

	X-Dynalign		M	sensitivity	PPV	MCC	Dynalign		M	sensitivity	PPV	MCC
1	RD4800_RD2640_RD1140	RD4800	6	100	100	100	RD4800_RD2640	RD4800	6	100	100	100
	RD4800_RD2640_RD1140	RD2640		100	100	100	RD4800_RD2640	RD2640		100	100	100
	RD4800_RD2640_RD1140	RD1140		100	100	100	RD4800_RD1140	RD4800		100	100	100
							RD4800_RD1140	RD1140		100	100	100
							RD2640_RD1140	RD2640		100	100	100
							RD2640_RD1140	RD1140		100	100	100
2	RD4800_RE2140_RD2640	RD4800	6	100	100	100	RD4800_RE2140	RD4800	6	100	100	100
	RD4800_RE2140_RD2640	RE2140		100	100	100	RD4800_RE2140	RE2140		100	100	100
	RD4800_RE2140_RD2640	RD2640		100	100	100	RD4800_RD2640	RD4800		100	100	100
							RD4800_RD2640	RD2640		100	100	100
							RE2140_RD2640	RE2140		100	100	100
							RE2140_RD2640	RD2640		100	100	100
3	RD4800_RE6781_RE2140	RD4800	5	100	100	100	RD4800_RE6781	RD4800	5	100	100	100
	RD4800_RE6781_RE2140	RE6781		100	100	100	RD4800_RE6781	RE6781		100	100	100
	RD4800_RE6781_RE2140	RE2140		100	100	100	RD4800_RE2140	RD4800		100	100	100
							RD4800_RE2140	RE2140		100	100	100
							RE6781_RE2140	RE6781		100	100	100
							RE6781_RE2140	RE2140		100	100	100
4	RF6320_RE6781_RE2140	RF6320	3	100	100	100	RF6320_RE6781	RF6320	3	100	100	100
	RF6320_RE6781_RE2140	RE6781		100	100	100	RF6320_RE6781	RE6781		100	100	100
	RF6320_RE6781_RE2140	RE2140		100	100	100	RF6320_RE2140	RF6320		100	100	100
							RF6320_RE2140	RE2140		100	100	100
							RE6781_RE2140	RE6781		100	100	100
							RE6781_RE2140	RE2140		100	100	100
5	RE2140_RE6781_RD0260	RE2140	3	100	100	100	RE2140_RE6781	RE2140	3	100	100	100
	RE2140_RE6781_RD0260	RE6781		100	100	100	RE2140_RE6781	RE6781		100	100	100
	RE2140_RE6781_RD0260	RD0260		100	100	100	RE2140_RD0260	RE2140		100	100	100
							RE2140_RD0260	RD0260		100	100	100
							RE6781_RD0260	RE6781		100	100	100
							RE6781_RD0260	RD0260		100	100	100
6	RE6781_RE2140_RD1140	RE6781	3	100	100	100	RE6781_RE2140	RE6781	3	100	100	100
	RE6781_RE2140_RD1140	RE2140		100	100	100	RE6781_RE2140	RE2140		100	100	100
	RE6781_RE2140_RD1140	RD1140		100	100	100	RE6781_RD1140	RE6781		100	100	100
							RE6781_RD1140	RD1140		100	100	100
							RE2140_RD1140	RE2140		100	100	100
							RE2140_RD1140	RD1140		100	100	100
7	RF6320_RE6781_RD2640	RF6320	3	100	100	100	RF6320_RE6781	RF6320	3	100	100	100
	RF6320_RE6781_RD2640	RE6781		100	100	100	RF6320_RE6781	RE6781		100	100	100
	RF6320_RE6781_RD2640	RD2640		100	100	100	RF6320_RD2640	RF6320		100	100	100
							RF6320_RD2640	RD2640		100	100	100
							RE6781_RD2640	RE6781		100	100	100
							RE6781_RD2640	RD2640		100	100	100
8	RF6320_RD2640_RD1140	RF6320	3	100	100	100	RF6320_RD2640	RF6320	3	100	100	100
	RF6320_RD2640_RD1140	RD2640		100	100	100	RF6320_RD2640	RD2640		100	100	100
	RF6320_RD2640_RD1140	RD1140		100	100	100	RF6320_RD1140	RF6320		100	100	100
							RF6320_RD1140	RD1140		100	100	100
							RD2640_RD1140	RD2640		100	100	100
							RD2640_RD1140	RD1140		100	100	100

Table 8 **X-Dynalign does not degrade already good predictions of Dynalign (Gap penalty=2)**

	X-Dynalign		M	sensitivity	PPV	MCC	Dynalign		M	sensitivity	PPV	MCC
9	RE6781_RD0260_RD1140	RE6781	3	100	100	100	RE6781_RD0260	RE6781	3	100	100	100
	RE6781_RD0260_RD1140	RD0260		100	100	100	RE6781_RD0260	RD0260		100	100	100
	RE6781_RD0260_RD1140	RD1140		100	100	100	RE6781_RD1140	RE6781		100	100	100
							RE6781_RD1140	RD1140		100	100	100
							RD0260_RD1140	RD0260		100	100	100
							RD0260_RD1140	RD1140		100	100	100
10	RL0503_RS0380_RL1141	RL0503	3	95.2	100	97.6	RL0503_RS0380	RL0503	3	95.8	92	93.9
	RL0503_RS0380_RL1141	RS0380		95.2	100	97.6	RL0503_RS0380	RS0380		92	92	92
	RL0503_RS0380_RL1141	RL1141		95.2	100	97.6	RL0503_RL1141	RL0503		95.8	100	97.9
							RL0503_RL1141	RL1141		92	100	95.9
							RS0380_RL1141	RS0380		92	100	95.9
							RS0380_RL1141	RL1141		92	100	95.9
11	RF6320_RE6781_RD0260	RF6320	3	100	100	100	RF6320_RE6781	RF6320	3	100	100	100
	RF6320_RE6781_RD0260	RE6781		100	100	100	RF6320_RE6781	RE6781		100	100	100
	RF6320_RE6781_RD0260	RD0260		100	100	100	RF6320_RD0260	RF6320		100	100	100
							RF6320_RD0260	RD0260		100	100	100
							RE6781_RD0260	RE6781		100	100	100
							RE6781_RD0260	RD0260		100	100	100

Table 8 continued

5-3-1 Comparative Analysis

This section presents the results of a comparative analysis of X-Dynalign and Dynalign for two datasets, tRNA and 5S rRNA.

5-3-1-1 tRNA Dataset

We made 9 runs, 27 predictions, using X-Dynalign, and 19 runs, 38 predictions, using Dynalign. The mean PPV, sensitivity, and MCC and their variance are presented in Table 9.

Mean	X-Dynalign	Dynalign
Sensitivity	94.4±7.5	89.1±15.7
PPV	96.8±7.6	92.1±14.6
MCC	95.6±7.3	90.5±15.0

Table 9 Mean sensitivity, PPV, and MCC for tRNA dataset

It is observed from Table 9 that X-Dynalign not only improves all three measures but also reduces their variance for this particular dataset.

Table 10 and Table 11 present the sensitivity and PPV per sequence. N is the number of predictions. The data has been organized so as to illustrate three scenarios: the best case, the worst case, and the average case.

Dynalign performed well in the best-case scenario, as it was possible to find a pair of input sequences having a high positive predictive value for all the sequences. The maximum PPV for all entries is 100, except for RS0380. Further analysis shows that the structure of RS0380 has an extra stem in the variable loop that X-Dynalign predicted more accurately, as shown in Figure 10. The maximum sensitivity for X-Dynalign is equal or more than that of Dynalign in 9 out of 10 sequences.

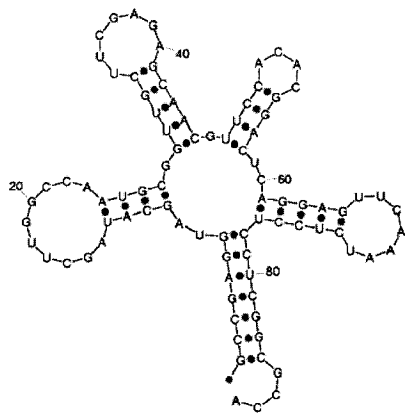


Figure 8 **Reference structure for RS0380**

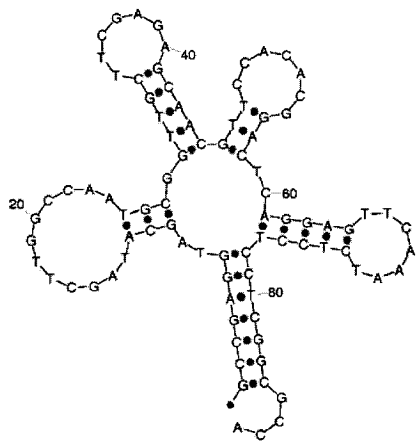


Figure 9 **Dynalign prediction for RS0380 (dataset: RS0380, RL1141)**

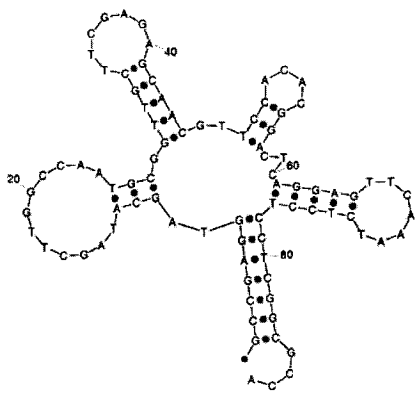


Figure 10 **X-Dynalign prediction for RS0380 (dataset: RS0380, RL1141, RS1141)**

The strategy for both algorithms is to find a structure that minimizes a linear combination of free energy of each input sequence given the common structure. Using three input sequences should have a positive impact on the worse case scenario. It should be less likely that all three input sequences jointly fold into the wrong minimum free energy structure than with two input sequences. Our results support this observation as well. All the entries for the minimum PPV for X-Dynalign are the same or better than that of Dynalign. The minimum PPV for 8 out of 10 sequences is 100. The two sequences leading to the worse predictions are RD0500 and RF6320. Figure 12 and Figure 13 present the RD0500 reference structure and the predictions made by Dynalign and X-Dynalign respectively. Dynalign generates an elongated structure. However, by using a third sequence (X-Dynalign) the accuracy increased by more than 30 percentage points. The structure produced by X-Dynalign has the overall cloverleaf shape; however, some of the nucleotides are shifted. The minimum coverage is generally good. For all the sequences the coverage is 75% or better. For all the sequences the coverage obtained using X-Dynalign is the same or better than the coverage obtained using Dynalign.

	X-Dynalign				Dynalign			
id	N	Min	Max	Ave	N	Min	Max	Ave
RD0260	4	95	100	99	5	57	100	91
RD0500	4	76	95	81	5	47	95	80
RD4800	5	95	100	99	5	57	100	91
RE2140	2	100	100	100	4	95	100	99
RE6781	2	100	100	100	4	81	100	95
RF6320	4	95	100	96	5	47	100	90
RL0503	1	95	95	95	2	95	95	95
RL1141	2	92	92	92	3	68	92	84
RS0380	1	92	92	92	2	80	80	80
RS1141	2	88	88	88	3	65	92	82

Table 10 Sensitivity for tRNA dataset

	X-Dynalign				Dynalign			
id	N	Min	Max	Ave	N	Min	Max	Ave
RD0260	4	100	100	100	5	80	100	96
RD0500	4	76	100	82	5	45	100	81
RD4800	5	100	100	100	5	80	100	96
RE2140	2	100	100	100	4	100	100	100
RE6781	2	100	100	100	4	77	100	94
RF6320	4	95	100	96	5	45	100	89
RL0503	1	100	100	100	2	100	100	100
RL1141	2	100	100	100	3	70	100	90
RS0380	1	100	100	100	2	83	87	85
RS1141	2	100	100	100	3	70	100	90

Table 11 PPV for tRNA dataset

	X-Dynalign				Dynalign			
id	N	Min	Max	Ave	N	Min	Max	Ave
RD0260	4	97	100	99	5	67	100	93
RD0500	4	76	97	82	5	46	97	80
RD4800	5	97	100	100	5	67	100	94
RE2140	2	100	100	100	4	97	100	99
RE6781	2	100	100	100	4	79	100	95
RF6320	4	95	100	96	5	46	100	89
RL0503	1	97	97	97	2	97	97	97
RL1141	2	95	95	95	3	69	95	87
RS0380	1	95	95	95	2	81	83	82
RS1141	2	94	94	94	3	68	96	86

Table 12 MCC for tRNA dataset

	X-Dynalign		M	sensitivity	PPV	MCC	Dynalign		M	sensitivity	PPV	MCC
1	RD0500_RF6320_RE6781	RD0500	4	95.2	100	97.6	RE6781_RD0500	RE6781	4	81	77.3	79.1
	RD0500_RF6320_RE6781	RF6320		95.2	100	97.6	RE6781_RD0500	RD0500		81	77.3	79.1
	RD0500_RF6320_RE6781	RE6781		95.2	100	97.6	RE6781_RF6320	RE6781		100	100	100
							RE6781_RF6320	RF6320		100	100	100
							RD0500_RF6320	RD0500		57.1	54.5	55.8
							RD0500_RF6320	RF6320		57.1	54.5	55.8
2	RS0380_RL1141_RS1141	RS0380	4	92	100	95.9	RS0380_RL1141	RS0380	4	92	100	95.9
	RS0380_RL1141_RS1141	RL1141		92	100	95.9	RS0380_RL1141	RL1141		92	100	95.9
	RS0380_RL1141_RS1141	RS1141		88.5	100	94.1	RS0380_RS1141	RS0380		96	100	98
							RS0380_RS1141	RS1141		92.3	100	96.1
							RL1141_RS1141	RL1141		68	70.8	69.4
							RL1141_RS1141	RS1141		65.4	70.8	68.1
3	RL0503_RL1141_RS1141	RL0503	4	95.8	100	97.9	RL0503_RL1141	RL0503	4	95.8	100	97.9
	RL0503_RL1141_RS1141	RL1141		92	100	95.9	RL0503_RL1141	RL1141		92	100	95.9
	RL0503_RL1141_RS1141	RS1141		88.5	100	94.1	RL0503_RS1141	RL0503		95.8	100	97.9
							RL0503_RS1141	RS1141		88.5	100	94.1
							RL1141_RS1141	RL1141		68	70.8	69.4
							RL1141_RS1141	RS1141		65.4	70.8	68.1
4	RD0500_RF6320_RE2140	RD0500	4	95.2	100	97.6	RE2140_RD0500	RE2140	4	95.2	100	97.6
	RD0500_RF6320_RE2140	RF6320		95.2	100	97.6	RE2140_RD0500	RD0500		95.2	100	97.6
	RD0500_RF6320_RE2140	RE2140		95.2	100	97.6	RE2140_RF6320	RE2140		100	100	100
							RE2140_RF6320	RF6320		100	100	100
							RD0500_RF6320	RD0500		57.1	54.5	55.8
							RD0500_RF6320	RF6320		57.1	54.5	55.8
5	RD4800_RD0500_RF6320	RD4800	5	100	100	100	RD4800_RD0500	RD4800	5	100	100	100
	RD4800_RD0500_RF6320	RD0500		76.2	76.2	76.2	RD4800_RD0500	RD0500		81	81	81
	RD4800_RD0500_RF6320	RF6320		95.2	95.2	95.2	RD4800_RF6320	RD4800		100	100	100
							RD4800_RF6320	RF6320		100	100	100
							RD0500_RF6320	RD0500		57.1	54.5	55.8
							RD0500_RF6320	RF6320		57.1	54.5	55.8
6	RD4800_RE6781_RD0260	RD4800	6	100	100	100	RD4800_RD0260	RD4800	6	57.1	80	67.6
	RD4800_RE6781_RD0260	RE6781		100	100	100	RD4800_RD0260	RD0260		57.1	80	67.6
	RD4800_RE6781_RD0260	RD0260		100	100	100	RD4800_RE6781	RD4800		100	100	100
							RD4800_RE6781	RE6781		100	100	100
							RE6781_RD0260	RE6781		100	100	100
							RE6781_RD0260	RD0260		100	100	100
7	RD4800_RD0260_RE2140	RD4800	6	100	100	100	RD4800_RD0260	RD4800	6	57.1	80	67.6
	RD4800_RD0260_RE2140	RD0260		100	100	100	RD4800_RD0260	RD0260		57.1	80	67.6
	RD4800_RD0260_RE2140	RE2140		100	100	100	RE2140_RD0260	RE2140		100	100	100
							RE2140_RD0260	RD0260		100	100	100
							RD4800_RE2140	RE2140		100	100	100
							RD4800_RE2140	RD4800		100	100	100

Table 13 **X-Dynalign improves accuracy vs. Dynalign for tRNA (Gap penalty = 2)**

	X-Dynalign		M	sensitivity	PPV	MCC	Dynalign		M	sensitivity	PPV	MCC
8	RD4800_RD0500_RD0260	RD4800	6	95.2	100	97.6	RD4800_RD0260	RD4800	6	57.1	80	67.6
	RD4800_RD0500_RD0260	RD0500		95.2	100	97.6	RD4800_RD0260	RD0260		57.1	80	67.6
	RD4800_RD0500_RD0260	RD0260		95.2	100	97.6	RD4800_RD0500	RD4800		50	50	50
							RD4800_RD0500	RD0500		57.1	60	58.6
							RD0500_RD0260	RD0500		95.2	100	97.6
							RD0500_RD0260	RD0260		95.2	100	97.6
9	RD4800_RF6320_RD0260	RD4800	6	100	100	100	RD4800_RD0260	RD4800	6	57.1	80	67.6
	RD4800_RF6320_RD0260	RF6320		100	100	100	RD4800_RD0260	RD0260		57.1	80	67.6
	RD4800_RF6320_RD0260	RD0260		100	100	100	RF6320_RD0260	RF6320		100	100	100
							RF6320_RD0260	RD0260		100	100	100
							RD4800_RF6320	RD4800		100	100	100
							RD4800_RF6320	RF6320		100	100	100

Table 13 Continued

5-3-1-2 5S rRNA Dataset

We made 19 runs, 57 predictions, using X-Dynalign, and 29 runs, 58 predictions, using Dynalign. The mean PPV, sensitivity, and MCC and their variance are presented in Table 14. Interestingly, it is observed that what is gained in accuracy is lost in sensitivity.

Mean	X-Dynalign	Dynalign
Sensitivity	76.6±5.3	79.2±6.7
PPV	90.3±5.8	87.7±7.4
MCC	83.2±5.5	83.3±6.7

Table 14 Mean sensitivity, PPV, and MCC for 5 S rRNA dataset

For this particular dataset, the performance of both algorithms is comparable based on Matthews Correlation Coefficient. Table 15, Table 16 and Table 17 show detailed data for the sensitivity, PPV and MCC, respectively for each sequence.

	X-Dynalign				Dynalign			
id	N	Min	Max	Ave	N	Min	Max	Ave
AJ131594	2	86	86	86	3	86	89	88
AJ251080	6	76	78	77	5	76	84	79
D11460	6	73	76	75	5	63	81	71
K02682	8	53	84	76	9	79	89	84
M10816	3	76	78	77	4	76	84	81
M16532	1	82	82	82	2	71	76	74
M25591	6	76	78	77	5	76	84	79
V00336	3	62	82	76	4	57	90	79
X02024	9	76	78	77	6	73	84	77
X02627	1	84	84	84	2	87	89	89
X04585	2	63	84	74	3	63	81	75
X08000	5	74	74	74	5	74	79	78
X08002	5	74	74	74	5	74	79	78

Table 15 Sensitivity for 5 S rRNA dataset

	X-Dynalign				Dynalign			
id	N	Min	Max	Ave	N	Min	Max	Ave
AJ131594	2	100	100	100	3	91	100	95
AJ251080	6	88	90	90	5	82	86	85
D11460	6	87	87	87	5	66	88	79
K02682	8	63	100	89	9	88	97	92
M10816	3	90	90	90	4	85	88	87
M16532	1	94	94	94	2	77	85	82
M25591	6	87	90	89	5	82	86	85
V00336	3	75	100	92	4	65	100	91
X02024	9	88	90	90	6	82	88	86
X02627	1	100	100	100	2	92	100	96
X04585	2	72	94	83	3	68	93	82
X08000	5	90	90	90	5	88	90	89
X08002	5	90	90	90	5	88	90	89

Table 16 PPV for 5 S rRNA dataset

	X-Dynalign				Dynalign			
id	N	Min	Max	Ave	N	Min	Max	Ave
AJ131594	2	93	93	93	3	89	93	91
AJ251080	6	83	84	83	5	79	85	82
D11460	6	80	81	80	5	64	85	75
K02682	8	58	92	82	9	85	93	88
M10816	3	83	84	83	4	80	86	84
M16532	1	87	87	87	2	74	81	78
M25591	6	81	83	83	5	79	85	82
V00336	3	68	90	84	4	61	94	85
X02024	9	83	84	83	6	79	86	81
X02627	1	92	92	92	2	90	93	92
X04585	2	67	89	78	3	65	87	79
X08000	5	82	82	82	5	82	83	83
X08002	5	82	82	82	5	82	83	83

Table 17 MCC for 5 S rRNA

Using three input sequences improves the worst PPV prediction for 12 out of 13 sequences. Also, for 10 out of 13 sequences, the minimum PPV obtained is 85% or more. The maximum sensitivity is the same or improved for 11 out of 13 sequences. However, the maximum sensitivity exceeds that of Dynalign for 2 out of 13 sequences.

The prediction of K02682 has an accuracy of 63% only. We believe this is due to the fact that single base pair insertion has not been implemented yet in X-Dynalign. In the triple (K02682, V00336, X04585), the sequence X04585 has a shorter helix IV than the other two sequences, 7 base pairs, compared to 8 for the first sequences. Furthermore, only 3 out the 7 base pairs are canonical, the other pairs are: two GAs, one GU and one GG base pair. X-Dynalign, Dynalign, and similar programs are not predicting the non-canonical pairs.

	X-Dynalign		M	sensitivity	PPV	MCC	Dynalign		M	sensitivity	PPV	MCC
1	AJ131594_X02627_V00336	AJ131594	5	86.8	100	93.2	X04585_V00336	X04585	5	63.2	68.6	65.8
	AJ131594_X02627_V00336	X02627		84.6	100	92	X04585_V00336	V00336		57.5	65.7	61.5
	AJ131594_X02627_V00336	V00336		82.5	100	90.8	AJ131594_V00336	AJ131594		86.8	100	93.2
							AJ131594_V00336	V00336		82.5	100	90.8
							AJ131594_X02627	AJ131594		89.5	91.9	90.7
							AJ131594_X02627	X02627		89.7	92.1	90.9
2	AJ251080_K02682_D11460	AJ251080	5	78.9	90.9	84.7	AJ251080_D11460	AJ251080	5	78.9	83.3	81.1
	AJ251080_K02682_D11460	K02682		76.9	90.9	83.6	AJ251080_D11460	D11460		81.1	83.3	82.2
	AJ251080_K02682_D11460	D11460		76.3	87.9	81.9	AJ251080_K02682	AJ251080		84.2	86.5	85.3
							AJ251080_K02682	K02682		84.6	89.2	86.9
							D11460_K02682	D11460		81.6	88.6	85
							D11460_K02682	K02682		82.1	91.4	86.6
3	AJ251080_K02682_X02024	AJ251080	5	78.9	88.2	83.5	AJ251080_X02024	AJ251080	5	76.3	82.9	79.5
	AJ251080_K02682_X02024	K02682		79.5	91.2	85.1	AJ251080_X02024	X02024		76.3	82.9	79.5
	AJ251080_K02682_X02024	X02024		78.9	88.2	83.5	AJ251080_K02682	AJ251080		84.2	86.5	85.3
							AJ251080_K02682	K02682		84.6	89.2	86.9
							X02024_K02682	X02024		84.2	88.9	86.5
							X02024_K02682	K02682		84.6	91.7	88.1
4	AJ251080_X08000_D11460	AJ251080	5	76.3	90.6	83.2	D11460_X08000	D11460	5	73.7	87.5	80.3
	AJ251080_X08000_D11460	X08000		74.4	90.6	82.1	D11460_X08000	X08000		74.4	90.6	82.1
	AJ251080_X08000_D11460	D11460		73.7	87.5	80.3	AJ251080_X08000	AJ251080		78.9	85.7	82.3
							AJ251080_X08000	X08000		79.5	88.6	83.9
							AJ251080_D11460	AJ251080		78.9	83.3	81.1
							AJ251080_D11460	D11460		81.1	83.3	82.2
5	AJ251080_X08000_X02024	AJ251080	5	76.3	90.6	83.2	AJ251080_X08000	AJ251080	5	78.9	85.7	82.3
	AJ251080_X08000_X02024	X08000		74.4	90.6	82.1	AJ251080_X08000	X08000		79.5	88.6	83.9
	AJ251080_X08000_X02024	X02024		76.3	90.6	83.2	AJ251080_X02024	AJ251080		76.3	82.9	79.5
							AJ251080_X02024	X02024		76.3	82.9	79.5
							X02024_X08000	X02024		73.7	87.5	80.3
							X02024_X08000	X08000		74.4	90.6	82.1

Table 18 **X-Dynalign improves accuracy vs. Dynalign for 5 S rRNA (Gap penalty=2)**

	X-Dynalign		M	sensitivity	PPV	MCC	Dynalign		M	sensitivity	PPV	MCC
6	AJ251080_X08002_D11460	AJ251080	5	76.3	90.6	83.2	D11460_X08002	D11460	5	73.7	87.5	80.3
	AJ251080_X08002_D11460	X08002		74.4	90.6	82.1	D11460_X08002	X08002		74.4	90.6	82.1
	AJ251080_X08002_D11460	D11460		73.7	87.5	80.3	AJ251080_X08002	AJ251080		78.9	85.7	82.3
							AJ251080_X08002	X08002		79.5	88.6	83.9
							AJ251080_D11460	AJ251080		78.9	83.3	81.1
							AJ251080_D11460	D11460		81.1	83.3	82.2
7	AJ251080_X08002_X02024	AJ251080	5	76.3	90.6	83.2	AJ251080_X08002	AJ251080	5	78.9	85.7	82.3
	AJ251080_X08002_X02024	X08002		74.4	90.6	82.1	AJ251080_X08002	X08002		79.5	88.6	83.9
	AJ251080_X08002_X02024	X02024		76.3	90.6	83.2	AJ251080_X02024	AJ251080		76.3	82.9	79.5
							AJ251080_X02024	X02024		76.3	82.9	79.5
							X02024_X08002	X02024		73.7	87.5	80.3
							X02024_X08002	X08002		74.4	90.6	82.1
8	K02682_AJ131594_V00336	K02682	5	84.6	100	92	K02682_V00336	K02682	5	89.7	97.2	93.4
	K02682_AJ131594_V00336	AJ131594		86.8	100	93.2	K02682_V00336	V00336		90	100	94.9
	K02682_AJ131594_V00336	V00336		82.5	100	90.8	AJ131594_K02682	AJ131594		86.8	91.7	89.2
							AJ131594_K02682	K02682		84.6	91.7	88.1
							AJ131594_V00336	AJ131594		86.8	100	93.2
							AJ131594_V00336	V00336		82.5	100	90.8
9	K02682_M10816_X02024	K02682	5	76.9	90.9	83.6	X02024_K02682	X02024	5	84.2	88.9	86.5
	K02682_M10816_X02024	M10816		78.9	90.9	84.7	X02024_K02682	K02682		84.6	91.7	88.1
	K02682_M10816_X02024	X02024		78.9	90.9	84.7	X02024_M10816	X02024		76.3	85.3	80.7
							X02024_M10816	M10816		76.3	85.3	80.7
							M10816_K02682	M10816		84.2	88.9	86.5
							M10816_K02682	K02682		82.1	88.9	85.4
10	K02682_M16532_X04585	K02682	5	82.1	94.1	87.9	X04585_M16532	X04585	5	78.9	85.7	82.3
	K02682_M16532_X04585	M16532		82.1	94.1	87.9	X04585_M16532	M16532		76.9	85.7	81.2
	K02682_M16532_X04585	X04585		84.2	94.1	89	X04585_K02682	X04585		81.6	93.9	87.5
							X04585_K02682	K02682		79.5	93.9	86.4
							K02682_M16532	K02682		87.2	94.4	90.7
							K02682_M16532	M16532		87.2	94.4	90.7
11	K02682_M25591_D11460	K02682	5	76.9	90.9	83.6	M25591_D11460	M25591	5	78.9	83.3	81.1
	K02682_M25591_D11460	M25591		76.3	87.9	81.9	M25591_D11460	D11460		81.1	83.3	82.2
	K02682_M25591_D11460	D11460		76.3	87.9	81.9	M25591_K02682	M25591		84.2	86.5	85.3
							M25591_K02682	K02682		84.6	89.2	86.9
							D11460_K02682	D11460		81.6	88.6	85
							D11460_K02682	K02682		82.1	91.4	86.6
12	K02682_M25591_X02024	K02682	5	79.5	91.2	85.1	M25591_K02682	M25591	5	84.2	86.5	85.3
	K02682_M25591_X02024	M25591		78.9	88.2	83.5	M25591_K02682	K02682		84.6	89.2	86.9
	K02682_M25591_X02024	X02024		78.9	88.2	83.5	X02024_K02682	X02024		84.2	88.9	86.5
							X02024_K02682	K02682		84.6	91.7	88.1
							M25591_X02024	M25591		76.3	82.9	79.5
							M25591_X02024	X02024		76.3	82.9	79.5
13	K02682_V00336_X04585	K02682	5	53.8	63.6	58.5	X04585_K02682	X04585	5	81.6	93.9	87.5
	K02682_V00336_X04585	V00336		62.5	75.8	68.8	X04585_K02682	K02682		79.5	93.9	86.4
	K02682_V00336_X04585	X04585		63.2	72.7	67.8	X04585_V00336	X04585		63.2	68.6	65.8
							X04585_V00336	V00336		57.5	65.7	61.5
							K02682_V00336	K02682		89.7	97.2	93.4
							K02682_V00336	V00336		90	100	94.9
14	M25591_X08000_D11460	M25591	5	76.3	90.6	83.2	M25591_D11460	M25591	5	78.9	83.3	81.1
	M25591_X08000_D11460	X08000		74.4	90.6	82.1	M25591_D11460	D11460		81.1	83.3	82.2
	M25591_X08000_D11460	D11460		73.7	87.5	80.3	M25591_X08000	M25591		78.9	85.7	82.3
							M25591_X08000	X08000		79.5	88.6	83.9
							D11460_X08000	D11460		73.7	87.5	80.3
							D11460_X08000	X08000		74.4	90.6	82.1

Table 18 continued

	X-Dynalign		M	sensitivity	PPV	MCC	Dynalign		M	sensitivity	PPV	MCC
15	M25591_X08000_X02024	M25591	5	76.3	90.6	83.2	X02024_X08000	X02024	5	73.7	87.5	80.3
	M25591_X08000_X02024	X08000		74.4	90.6	82.1	X02024_X08000	X08000		74.4	90.6	82.1
	M25591_X08000_X02024	X02024		76.3	90.6	83.2	M25591_X02024	M25591		76.3	82.9	79.5
							M25591_X02024	X02024		76.3	82.9	79.5
							M25591_X08000	M25591		78.9	85.7	82.3
							M25591_X08000	X08000		79.5	88.6	83.9
16	M25591_X08002_D11460	M25591	5	76.3	90.6	83.2	M25591_X08002	M25591	5	78.9	85.7	82.3
	M25591_X08002_D11460	X08002		74.4	90.6	82.1	M25591_X08002	X08002		79.5	88.6	83.9
	M25591_X08002_D11460	D11460		73.7	87.5	80.3	D11460_X08002	D11460		73.7	87.5	80.3
							D11460_X08002	X08002		74.4	90.6	82.1
							M25591_D11460	M25591		78.9	83.3	81.1
							M25591_D11460	D11460		81.1	83.3	82.2
17	M25591_X08002_X02024	M25591	5	76.3	90.6	83.2	M25591_X08002	M25591	5	78.9	85.7	82.3
	M25591_X08002_X02024	X08002		74.4	90.6	82.1	M25591_X08002	X08002		79.5	88.6	83.9
	M25591_X08002_X02024	X02024		76.3	90.6	83.2	M25591_X02024	M25591		76.3	82.9	79.5
							M25591_X02024	X02024		76.3	82.9	79.5
							X02024_X08002	X02024		73.7	87.5	80.3
							X02024_X08002	X08002		74.4	90.6	82.1
18	X08000_M10816_X02024	X08000	5	74.4	90.6	82.1	X02024_X08000	X02024	5	73.7	87.5	80.3
	X08000_M10816_X02024	M10816		76.3	90.6	83.2	X02024_X08000	X08000		74.4	90.6	82.1
	X08000_M10816_X02024	X02024		76.3	90.6	83.2	M10816_X08000	M10816		81.6	88.6	85
							M10816_X08000	X08000		79.5	88.6	83.9
							X02024_M10816	X02024		76.3	85.3	80.7
							X02024_M10816	M10816		76.3	85.3	80.7
19	X08002_M10816_X02024	X08002	5	74.4	90.6	82.1	X02024_X08002	X02024	5	73.7	87.5	80.3
	X08002_M10816_X02024	M10816		76.3	90.6	83.2	X02024_X08002	X08002		74.4	90.6	82.1
	X08002_M10816_X02024	X02024		76.3	90.6	83.2	X02024_M10816	X02024		76.3	85.3	80.7
							X02024_M10816	M10816		76.3	85.3	80.7
							M10816_X08002	M10816		81.6	88.6	85
							M10816_X08002	X08002		79.5	88.6	83.9

Table 18 continued

The results obtained for the two datasets, using X-Dynalign and Dynalign, are remarkably accurate. We wanted to know if those examples represented easy targets for energy minimization approaches in general. Therefore, we also ran MFOLD, which is the most widely used RNA secondary structure prediction program. We used the version 3.1.2. with the default set of parameters. Table 19 and Table 20 show that these sequences are challenging cases for MFOLD.

MFOLD	Sensitivity	PPV	MCC
RD0260	33.3	29.2	31.2
RD0500	47.6	43.5	45.5
RD4800	42.9	56.2	49.1
RE2140	95.2	87	91
RE6781	33.3	28	30.6
RF6320	0	0	0
RL0503	0	0	0
RL1141	40	43.5	41.7
RS0380	52	56.5	54.2
RS1141	19.2	25	21.9

Table 19 MFOLD results for tRNA sequences

MFOLD	Sensitivity	PPV	MCC
AJ131594	23.7	60	37.7
AJ251080	26.3	45.5	34.6
D11460	15.8	37.5	24.3
K02682	20.5	40	28.6
M10816	31.6	70.6	47.2
M16532	10.3	21.1	14.7
M25591	26.3	45.5	34.6
V00336	37.5	65.2	49.5
X02024	15.8	37.5	24.3
X02627	38.5	68.2	51.2
X04585	0	0	0
X08000	0	0	0
X08002	0	0	0

Table 20 MFOLD results for 5 S rRNA sequences

Here is an example of X-Dynalign producing more accurate results than Dynalign.

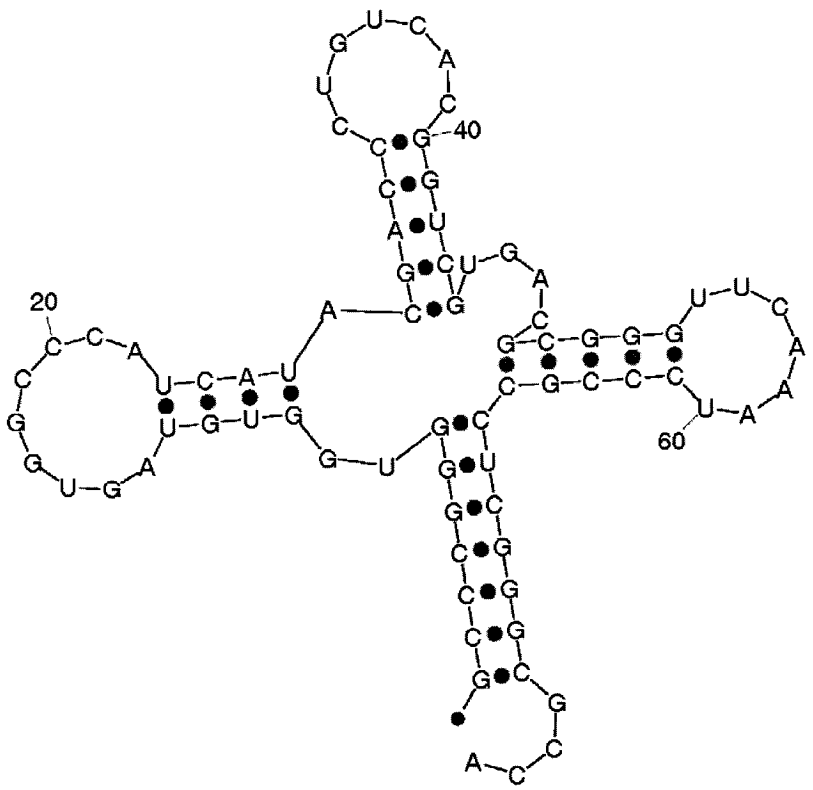


Figure 11 **Reference structure for RD0500 (*Archae*)**

Figure 11 shows the reference structure for the RNA sequence RD0500. Just like the other members of the tRNA family, RD0500 has four arms, and it is commonly referred to as the cloverleaf structure [27]. For this particular sequence, Dynalign predicts an elongated structure (two arms), as can be seen in Figure 12. However, X-Dynalign's prediction, Figure 13, is very similar to the original cloverleaf shape.

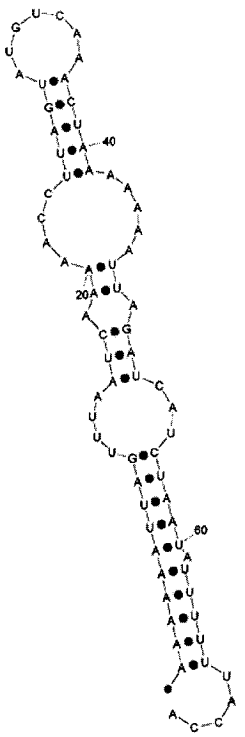


Figure 12 **Dynalign prediction for RD0500 from the input sequences RD4800 and RD0500**

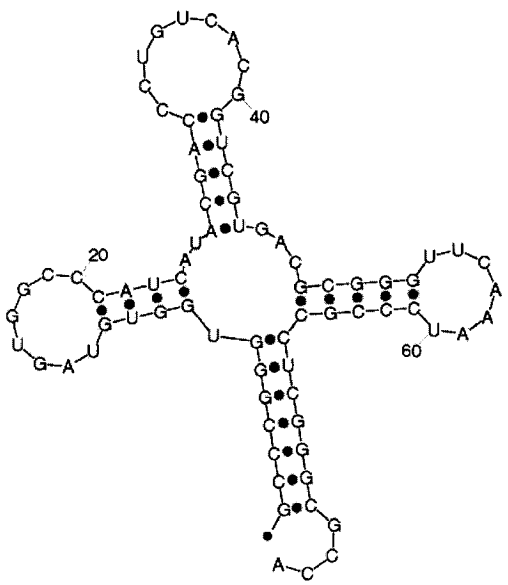


Figure 13 **X-Dynalign prediction for RD0500 from the input sequences RD4800, RD0500, RF6320)**

CHAPTER 6

CONCLUSION AND FUTURE WORK

IN THIS CHAPTER

- Conclusion
- Future work

CHAPTER SIX

6-1 Conclusion

We have extended the software system Dynalign to use three input sequences, rather than two. The resulting system is called eXtended-Dynalign (X-Dynalign for short). Its time/space complexity limits its application to:

1. Short sequences (say less than 200 nucleotides), and
2. Sequences that can be aligned optimally with a small value of M (less than 6), where M is the maximum distance of aligned positions.

The strengths of Dynalign carry over to the new system. Namely, it improves the accuracy of secondary structure predictions compared to predictions based on a single input sequence. It requires no sequence homology because the identity of aligned nucleotides is not part of the scoring function [19]. It also shares some of its limitations. In particular, the gap penalties are treated as a separate term in the objective function. The optimal value has to be determined empirically. In [19], it was found that the optimal value for this term depends on the class of RNA studied. In our limited experiments, the dependency seems less important. It also seems that there is large plateau where several gap penalty scores are leading to a nearly optimal solution w.r.t. PPV, for example.

Our key conclusions are:

- The lowest PPV for any prediction is generally improved when using three input sequences;
- The average accuracy is improved;
- The average sensitivity of the algorithm slightly degraded for the 5S rRNA dataset; however, a per-sequence analysis shows that the majority of the lowest sensitivity scores are higher for X-Dynalign than Dynalign;
- X-Dynalign is able to reproduce subtle details, such as the prediction of a stem in the variable region of certain tRNAs.

The detailed knowledge of RNA secondary structure is essential for understanding the sequence-structure-function relationships. X-Dynalign takes advantage of the paramount of data that is accumulating in sequence databases. Because it requires no sequence homology, X-Dynalign should be useful to comparative RNA sequence analyses.

6-2 Future work

Current thermodynamic parameters were obtained by performing melting studies on small oligonucleotides, but the stabilizing or destabilizing effects of longer sequences are still unknown [7]. More extensive and detailed melting studies, especially on longer loops, to determine additional stabilization values for hairpins and destabilization values for bulges and internal loops, could be very helpful to improve current algorithms. Secondary structure prediction algorithms can only be as accurate as the thermodynamic parameters allow them to be; therefore, better thermodynamic values will help generating more accurate predictions [11]. Furthermore, most current algorithms assume that the total free energy of an RNA secondary structure can be computed by summing the contributions of the components, but this may not be accurate in many cases [4].

There are a few obvious extensions for this class of algorithms, such as handling pseudo-knots and reporting suboptimal structures. However, one of the most urgent improvements is to reduce the time/space complexity. Several runs presented herein take up to a week to compute on some of the fastest processors (Opteron) available today.

As a result of the many ongoing sequencing research projects, several homologous sequences are often available. For example, in the case of the Hepatitis *delta* virus there are more than 75 complete non-redundant sequences available. Consequently arise the need 1) to devise algorithms to select sequence triplets, 2) to study the effect of the selection process on the accuracy of the results and 3) to propose a methodology for integrating the results. One of the possibilities for selecting the sequences would be to group the sequences such that the members of a group of three sequences have a low pairwise sequence identity and that a maximum number of sequences are used for the experiments. We also need to study the impact of the sequence similarity on the results. At least intuitively, it seems that selecting sequences that are highly similar will not have as a positive impact as selecting divergent sequences. At the other side of the scale, we should verify if there is a certain threshold such that the algorithm does not perform well if the sequences are too divergent. The relationship, if any, in between the sequence identity and the accuracy and coverage of the algorithm should be investigated. Finally, whenever there are several ways to select the input sequences, it is very likely that there will also be more than one family of answers. How can these results be most effectively reconciled? Tools should be developed to 1) identify families of structures within the set of outputs (clustering), 2) count the number of times the base pairs are formed in all of the solutions, and 3) help the user visualising the solution space.

Because of its time complexity, X-Dynalign has only been tested on short sequences. Beyond 150 nucleotides, its behaviour is unknown. Until faster computers become available or novel algorithmic developments are made, longer sequences have to be studied using a sliding window approach. Typically, a window of size 150 nt is moved along the sequence, with increments of 20 positions, until the end of the sequence is reached. Since the windows are overlapping, for any given region of the input sequence(s) there could be several, conflicting or not, solutions. Algorithms need to be developed to reconcile these results. Moreover, the impact of the window size and increment has not been studied yet for X-Dynalign. Large window sizes could create "blind spots". However, the impact of the window size and increment should be systematically studied using annotated data.

Finally, whenever several competing solutions are produced, either because there were many ways to select the input sequences or because overlapping windows are used, there should be effective methods for ranking the results, from the most likely to the least likely candidate, so as to facilitate the experimental validation of the results. Ideally, it should be possible to devise a test to estimate the likelihood of obtaining of a particular structure/energy value. The raw free energy value may not be the best indicator. Most likely, its value depends on the composition of the input sequences. The development of randomised tests for estimating the likelihood of a structure should be explored.

APPENDIX A

Appendix A

This dissertation has presented a computational approach to RNA secondary structure prediction. The software system developed for this work finds a common secondary structure and alignment for three input sequences that minimizes the sum of the free energy of each sequence, given the common secondary structure, plus gaps. The formalism used to characterize the free energy is called the nearest-neighbour model. This Appendix gives a detailed presentation of recurrence equations that constitute the objective function minimized by our software system.

The nearest-neighbour model defines the total free energy of an RNA molecule as the sum of the free energies of *k*-cycles. The parameter *k* is defined as the number of accessible base pairs plus 1. See Section 1-3-1-2 for the definitions of cycles and accessible base pairs. For k=1, the resulting cycle is a hairpin, for k=2 and u>0 the resulting cycle is either a bulge or an interior loop, for k=2 and u=0 to the cycle is a stacked base pair, and for k≥3 the cycle is a multibranch loop [23], here u refers to the number of nucleotides in between the surrounding base pair and the accessible base pair. In the presentation that follows, the first number of the subscript refers to the parameter *k*, for instance, V_1 describes the total free energy for a hairpin loop.

Detailed recurrence equations for W, V and W9 are as follows:

V_1 considers hairpin loops closed by base-pairs *i:j* and *k:l* and *m:n*:

$$V_1 = \Delta G_{hairpin}(i, j) + \Delta G_{hairpin}(k, l) + \Delta G_{hairpin}(m, n) + (no.\ of\ gaps)\ \Delta G_{gap}$$

V_2 is the lowest sum of the free energies for a helix extension, bulge loop, or internal loop in the common structure.

$$V_2 = min\,(V(i', j', k', l', m', n') + \Delta G_{motif1} + \Delta G_{motif2} + \Delta G_{motif3}$$

V_3 is the lowest sum of free energies for a multibranch loop closed by pairs *i:j*, *k:l* and *m:n*.

$$V_{3\text{-}1} = W(i+1, i', k+1, k', m+1, m') + W(i'+1, j-1, k'+1, l-1, m'+1, n-1) + 3\ \Delta G_{MBL\ closure} + 3\ \Delta G_{helix\ terminating\ in\ MBL}$$

$V_{3\text{-}2} = W(i+1, i', k+1, k', m+1, m') + W(i'+1, j-1, k'+1, l-1, m'+1, n-2) + \Delta G_{unpaired\ nucleotides\ in\ MBL\ loop} + 3\ \Delta G_{MBL\ closure} + 3\ \Delta G_{helix\ terminating\ in\ MBL} + 2\ \Delta G_{gap} + \Delta G_{dangle\ n-1}$

$V_{3\text{-}3} = W(i+1, i', k+2, k', m+1, m') + W(i'+1, j-1, k'+1, l-1, m'+1, n-1) + \Delta G_{unpaired\ nucleotides\ in\ MBL\ loop} + 3\ \Delta G_{MBL\ closure} + 3\ \Delta G_{helix\ terminating\ in\ MBL} + 2\ \Delta G_{gap} + \Delta G_{dangle\ k+1}$

$V_{3\text{-}4} = W(i+1, i', k+1, k', m+2, m') + W(i'+1, j-1, k'+1, l-1, m'+1, n-2) + 2\ \Delta G_{unpaired\ nucleotides\ in\ MBL\ loop} + 3\ \Delta G_{MBL\ closure} + 3\ \Delta G_{helix\ terminating\ in\ MBL} + 4\Delta G_{gap} + \Delta G_{dangle\ m+1} + \Delta G_{dangle\ n-1}$

$V_{3\text{-}5} = W(i+1, i', k+1, k', m+1, m') + W(i'+1, j-1, k'+1, l-2, m'+1, n-1) + \Delta G_{unpaired\ nucleotides\ in\ MBL\ loop} + 3\ \Delta G_{MBL\ closure} + 3\ \Delta G_{helix\ terminating\ in\ MBL} + 2\ \Delta G_{gap} + \Delta G_{dangle\ l-1}$

$V_{3\text{-}6} = W(i+1, i', k+1, k', m+1, m') + W(i'+1, j-1, k'+1, l-2, m'+1, n-2) + 2\ \Delta G_{unpaired\ nucleotides\ in\ MBL\ loop} + 3\ \Delta G_{MBL\ closure} + 3\ \Delta G_{helix\ terminating\ in\ MBL} + \Delta G_{gap} + \Delta G_{dangle\ l-1} + \Delta G_{dangle\ n-1}$

$V_{3\text{-}7} = W(i+1, i', k+1, k', m+2, m') + W(i'+1, j-1, k'+1, l-2, m'+1, n-1) + 2\ \Delta G_{unpaired\ nucleotides\ in\ MBL\ loop} + 3\ \Delta G_{MBL\ closure} + 3\ \Delta G_{helix\ terminating\ in\ MBL} + 4\ \Delta G_{gap} + \Delta G_{dangle\ l-1} + \Delta G_{dangle\ m+1}$

$V_{3\text{-}8} = W(i+1, i', k+1, k', m+2, m') + W(i'+1, j-1, k'+1, l-2, m'+1, n-2) + 3\ \Delta G_{unpaired\ nucleotides\ in\ MBL\ loop} + 3\ \Delta G_{MBL\ closure} + 3\ \Delta G_{helix\ terminating\ in\ MBL} + 3\ \Delta G_{gap} + \Delta G_{dangle\ l-1} + \Delta G_{dangle\ m+1} + \Delta G_{dangle\ n-1}$

$V_{3\text{-}9} = W(i+1, i', k+2, k', m+1, m') + W(i'+1, j-1, k'+1, l-1, m'+1, n-1) + \Delta G_{unpaired\ nucleotides\ in\ MBL\ loop} + 3\ \Delta G_{MBL\ closure} + 3\ \Delta G_{helix\ terminating\ in\ MBL} + 2\ \Delta G_{gap} + \Delta G_{dangle\ k+1}$

$V_{3\text{-}10} = W(i+1, i', k+2, k', m+1, m') + W(i'+1, j-1, k'+1, l-1, m'+1, n-2) + 2\ \Delta G_{unpaired\ nucleotides\ in\ MBL\ loop} + 3\ \Delta G_{MBL\ closure} + 3\ \Delta G_{helix\ terminating\ in\ MBL} + 4\ \Delta G_{gap} + \Delta G_{dangle\ k+1} + \Delta G_{dangle\ n-1}$

$V_{3\text{-}11} = W(i+1, i', k+2, k', m+2, m') + W(i'+1, j-1, k'+1, l-1, m'+1, n-1) + 2\Delta G_{unpaired\ nucleotides\ in\ MBL\ loop} + 3\ \Delta G_{MBL\ closure} + 3\ \Delta G_{helix\ terminating\ in\ MBL} + \Delta G_{gap} + \Delta G_{dangle\ k+1} + \Delta G_{dangle\ m+1}$

$V_{3\text{-}12} = W(i+1, i', k+2, k', m+2, m') + W(i'+1, j-1, k'+1, l-1, m'+1, n-2) + 3\ \Delta G_{unpaired\ nucleotides\ in\ MBL\ loop} + 3\ \Delta G_{MBL\ closure} + 3\ \Delta G_{helix\ terminating\ in\ MBL} + 3\ \Delta G_{gap} + \Delta G_{dangle\ k+1} + \Delta G_{dangle\ m+1} + \Delta G_{dangle\ n-1}$

$$V_{3\text{-}13} = W(i+1, i', k+2, k', m+1, m') + W(i'+1, j-1, k'+1, l-2, m'+1, n-1) + 2\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 3\,\Delta G_{\text{MBL closure}} + 3\,\Delta G_{\text{helix terminating in MBL}} + 4\,\Delta G_{gap} + \Delta G_{\text{dangle } k+1} + \Delta G_{\text{dangle } l-1}$$

$$V_{3\text{-}14} = W(i+1, i', k+2, k', m+1, m') + W(i'+1, j-1, k'+1, l-2, m'+1, n-2) + 3\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 3\,\Delta G_{\text{MBL closure}} + 3\,\Delta G_{\text{helix terminating in MBL}} + 3\,\Delta G_{gap} + \Delta G_{\text{dangle } k+1} + \Delta G_{\text{dangle } l-1} + \Delta G_{\text{dangle } n-1}$$

$$V_{3\text{-}15} = W(i+1, i', k+2, k', m+2, m') + W(i'+1, j-1, k'+1, l-2, m'+1, n-1) + 3\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 3\,\Delta G_{\text{MBL closure}} + 3\,\Delta G_{\text{helix terminating in MBL}} + 3\,\Delta G_{gap} + \Delta G_{\text{dangle } k+1} + \Delta G_{\text{dangle } l-1} + \Delta G_{\text{dangle } m+1}$$

$$V_{3\text{-}16} = W(i+1, i', k+2, k', m+2, m') + W(i'+1, j-1, k'+1, l-2, m'+1, n-2) + 4\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 3\,\Delta G_{\text{MBL closure}} + 3\,\Delta G_{\text{helix terminating in MBL}} + 2\,\Delta G_{gap} + \Delta G_{\text{dangle } k+1} + \Delta G_{\text{dangle } l-1} + \Delta G_{\text{dangle } m+1} + \Delta G_{\text{dangle } n-1}$$

$$V_{3\text{-}17} = W(i+1, i', k+1, k', m+1, m') + W(i'+1, j-2, k'+1, l-1, m'+1, n-1) + \Delta G_{\text{unpaired nucleotides in MBL loop}} + 3\,\Delta G_{\text{MBL closure}} + 3\,\Delta G_{\text{helix terminating in MBL}} + 2\,\Delta G_{gap} + \Delta G_{\text{dangle } j-1}$$

$$V_{3\text{-}18} = W(i+1, i', k+1, k', m+1, m') + W(i'+1, j-2, k'+1, l-1, m'+1, n-2) + 2\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 3\,\Delta G_{\text{MBL closure}} + 3\,\Delta G_{\text{helix terminating in MBL}} + \Delta G_{gap} + \Delta G_{\text{dangle } j-1} + \Delta G_{\text{dangle } n-1}$$

$$V_{3\text{-}19} = W(i+1, i', k+1, k', m+2, m') + W(i'+1, j-2, k'+1, l-1, m'+1, n-1) + 2\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 3\,\Delta G_{\text{MBL closure}} + 3\,\Delta G_{\text{helix terminating in MBL}} + 4\,\Delta G_{gap} + \Delta G_{\text{dangle } j-1} + \Delta G_{\text{dangle } m+1}$$

$$V_{3\text{-}20} = W(i+1, i', k+1, k', m+2, m') + W(i'+1, j-2, k'+1, l-1, m'+1, n-2) + 3\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 3\,\Delta G_{\text{MBL closure}} + 3\,\Delta G_{\text{helix terminating in MBL}} + 3\,\Delta G_{gap} + \Delta G_{\text{dangle } j-1} + \Delta G_{\text{dangle } m+1} + \Delta G_{\text{dangle } n-1}$$

$$V_{3\text{-}21} = W(i+1, i', k+1, k', m+1, m') + W(i'+1, j-2, k'+1, l-2, m'+1, n-1) + 2\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 3\,\Delta G_{\text{MBL closure}} + 3\,\Delta G_{\text{helix terminating in MBL}} + \Delta G_{gap} + \Delta G_{\text{dangle } j-1} + \Delta G_{\text{dangle } l-1}$$

$$V_{3\text{-}22} = W(i+1, i', k+1, k', m+1, m') + W(i'+1, j-2, k'+1, l-2, m'+1, n-2) + 3\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 3\,\Delta G_{\text{MBL closure}} + 3\,\Delta G_{\text{helix terminating in MBL}} + \Delta G_{\text{dangle } j-1} + \Delta G_{\text{dangle } l-1} + \Delta G_{\text{dangle } n-1}$$

$$V_{3\text{-}23} = W(i+1, i', k+1, k', m+2, m') + W(i'+1, j-2, k'+1, l-2, m'+1, n-1) + 3\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 3\,\Delta G_{\text{MBL closure}} + 3\,\Delta G_{\text{helix terminating in MBL}} + 3\,\Delta G_{gap} + \Delta G_{\text{dangle } j-1} + \Delta G_{\text{dangle } l-1} + \Delta G_{\text{dangle } m+1}$$

$$V_{3\text{-}24} = W(i+1, i', k+1, k', m+2, m') + W(i'+1, j-2, k'+1, l-2, m'+1, n-2) + 4\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 3\,\Delta G_{\text{MBL closure}} + 3\,\Delta G_{\text{helix terminating in MBL}} + 2\,\Delta G_{gap} + \Delta G_{\text{dangle } j-1} + \Delta G_{\text{dangle } l-1} + \Delta G_{\text{dangle } m+1} + \Delta G_{\text{dangle } n-1}$$

$$V_{3\text{-}25} = W(i+1, i', k+2, k', m+1, m') + W(i'+1, j-2, k'+1, l-1, m'+1, n-1) + 2\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 3\,\Delta G_{\text{MBL closure}} + 3\,\Delta G_{\text{helix terminating in MBL}} + 4\,\Delta G_{gap} + \Delta G_{\text{dangle } j-1} + \Delta G_{\text{dangle } k+1}$$

$$V_{3\text{-}26} = W(i+1, i', k+2, k', m+1, m') + W(i'+1, j-2, k'+1, l-1, m'+1, n-2) + 3\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 3\,\Delta G_{\text{MBL closure}} + 3\,\Delta G_{\text{helix terminating in MBL}} + 3\,\Delta G_{gap} + \Delta G_{\text{dangle } j-1} + \Delta G_{\text{dangle } k+1} + \Delta G_{\text{dangle } n-1}$$

$$V_{3\text{-}27} = W(i+1, i', k+2, k', m+2, m') + W(i'+1, j-2, k'+1, l-1, m'+1, n-1) + 3\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 3\,\Delta G_{\text{MBL closure}} + 3\,\Delta G_{\text{helix terminating in MBL}} + 3\,\Delta G_{gap} + \Delta G_{\text{dangle } j-1} + \Delta G_{\text{dangle } k+1} + \Delta G_{\text{dangle } m+1}$$

$$V_{3\text{-}28} = W(i+1, i', k+2, k', m+2, m') + W(i'+1, j-2, k'+1, l-1, m'+1, n-2) + 4\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 3\,\Delta G_{\text{MBL closure}} + 3\,\Delta G_{\text{helix terminating in MBL}} + 2\,\Delta G_{gap} + \Delta G_{\text{dangle } j-1} + \Delta G_{\text{dangle } k+1} + \Delta G_{\text{dangle } m+1} + \Delta G_{\text{dangle } n-1}$$

$$V_{3\text{-}29} = W(i+1, i', k+2, k', m+1, m') + W(i'+1, j-2, k'+1, l-2, m'+1, n-1) + 3\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 3\,\Delta G_{\text{MBL closure}} + 3\,\Delta G_{\text{helix terminating in MBL}} + 3\,\Delta G_{gap} + \Delta G_{\text{dangle } j-1} + \Delta G_{\text{dangle } k+1} + \Delta G_{\text{dangle } l-1}$$

$$V_{3\text{-}30} = W(i+1, i', k+2, k', m+1, m') + W(i'+1, j-2, k'+1, l-2, m'+1, n-2) + 4\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 3\,\Delta G_{\text{MBL closure}} + 3\,\Delta G_{\text{helix terminating in MBL}} + 2\,\Delta G_{gap} + \Delta G_{\text{dangle } j-1} + \Delta G_{\text{dangle } k+1} + \Delta G_{\text{dangle } l-1} + \Delta G_{\text{dangle } n-1}$$

$$V_{3\text{-}31} = W(i+1, i', k+2, k', m+2, m') + W(i'+1, j-2, k'+1, l-2, m'+1, n-1) + 4\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 3\,\Delta G_{\text{MBL closure}} + 3\,\Delta G_{\text{helix terminating in MBL}} + 2\,\Delta G_{gap} + \Delta G_{\text{dangle } j-1} + \Delta G_{\text{dangle } k+1} + \Delta G_{\text{dangle } l-1} + \Delta G_{\text{dangle } m+1}$$

$$V_{3\text{-}32} = W(i+1, i', k+2, k', m+2, m') + W(i'+1, j-2, k'+1, l-2, m'+1, n-2) + 5\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 3\,\Delta G_{\text{MBL closure}} + 3\,\Delta G_{\text{helix terminating in MBL}} + \Delta G_{gap} + \Delta G_{\text{dangle } j-1} + \Delta G_{\text{dangle } k+1} + \Delta G_{\text{dangle } l-1} + \Delta G_{\text{dangle } m+1} + \Delta G_{\text{dangle } n-1}$$

$$V_{3\text{-}33} = W(i+2, i', k+1, k', m+1, m') + W(i'+1, j-1, k'+1, l-1, m'+1, n-1) + \Delta G_{\text{unpaired nucleotides in MBL loop}} + 3\,\Delta G_{\text{MBL closure}} + 3\,\Delta G_{\text{helix terminating in MBL}} + 2\,\Delta G_{gap} + \Delta G_{\text{dangle } i+1}$$

$$V_{3\text{-}34} = W(i+2, i', k+1, k', m+1, m') + W(i'+1, j-1, k'+1, l-1, m'+1, n-2) + 2\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 3\,\Delta G_{\text{MBL closure}} + 3\,\Delta G_{\text{helix terminating in MBL}} + 4\,\Delta G_{gap} + \Delta G_{\text{dangle } i+1} + \Delta G_{\text{dangle } n-1}$$

$$V_{3\text{-}35} = W(i+2, i', k+1, k', m+2, m') + W(i'+1, j-1, k'+1, l-1, m'+1, n-1) + 2\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 3\,\Delta G_{\text{MBL closure}} + 3\,\Delta G_{\text{helix terminating in MBL}} + \Delta G_{gap} + \Delta G_{\text{dangle } i+1} + \Delta G_{\text{dangle } m+1}$$

$$V_{3\text{-}36} = W(i+2, i', k+1, k', m+2, m') + W(i'+1, j-1, k'+1, l-1, m'+1, n-2) + 3\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 3\,\Delta G_{\text{MBL closure}} + 3\,\Delta G_{\text{helix terminating in MBL}} + 3\,\Delta G_{gap} + \Delta G_{\text{dangle } i+1} + \Delta G_{\text{dangle } m+1} + \Delta G_{\text{dangle } n-1}$$

$$V_{3\text{-}37} = W(i+2, i', k+1, k', m+1, m') + W(i'+1, j-1, k'+1, l-2, m'+1, n-1) + 2\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 3\,\Delta G_{\text{MBL closure}} + 3\,\Delta G_{\text{helix terminating in MBL}} + 4\,\Delta G_{gap} + \Delta G_{\text{dangle } i+1} + \Delta G_{\text{dangle } l-1}$$

$$V_{3\text{-}38} = W(i+2, i', k+1, k', m+1, m') + W(i'+1, j-1, k'+1, l-2, m'+1, n-2) + 3\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 3\,\Delta G_{\text{MBL closure}} + 3\,\Delta G_{\text{helix terminating in MBL}} + 3\,\Delta G_{gap} + \Delta G_{\text{dangle } i+1} + \Delta G_{\text{dangle } l-1} + \Delta G_{\text{dangle } n-1}$$

$$V_{3\text{-}39} = W(i+2, i', k+1, k', m+2, m') + W(i'+1, j-1, k'+1, l-2, m'+1, n-1) + 3\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 3\,\Delta G_{\text{MBL closure}} + 3\,\Delta G_{\text{helix terminating in MBL}} + 3\,\Delta G_{gap} + \Delta G_{\text{dangle } i+1} + \Delta G_{\text{dangle } l-1} + \Delta G_{\text{dangle } m+1}$$

$V_{3\text{-}40} = W(i+2, i', k+1, k', m+2, m') + W(i'+1, j-1, k'+1, l-2, m'+1, n-2) + 4\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 3\,\Delta G_{\text{MBL closure}} + 3\,\Delta G_{\text{helix terminating in MBL}} + 2\,\Delta G_{gap} + \Delta G_{\text{dangle } i+1} + \Delta G_{\text{dangle } l-1} + \Delta G_{\text{dangle } m+1} + \Delta G_{\text{dangle } n-1}$

$V_{3\text{-}41} = W(i+2, i', k+2, k', m+1, m') + W(i'+1, j-1, k'+1, l-1, m'+1, n-1) + 2\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 3\,\Delta G_{\text{MBL closure}} + 3\,\Delta G_{\text{helix terminating in MBL}} + \Delta G_{gap} + \Delta G_{\text{dangle } i+1} + \Delta G_{\text{dangle } k+1}$

$V_{3\text{-}42} = W(i+2, i', k+2, k', m+1, m') + W(i'+1, j-1, k'+1, l-1, m'+1, n-2) + 3\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 3\,\Delta G_{\text{MBL closure}} + 3\,\Delta G_{\text{helix terminating in MBL}} + 3\,\Delta G_{gap} + \Delta G_{\text{dangle } i+1} + \Delta G_{\text{dangle } k+1} + \Delta G_{\text{dangle } n-1}$

$V_{3\text{-}43} = W(i+2, i', k+2, k', m+2, m') + W(i'+1, j-1, k'+1, l-1, m'+1, n-1) + 3\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 3\,\Delta G_{\text{MBL closure}} + 3\,\Delta G_{\text{helix terminating in MBL}} + \Delta G_{\text{dangle } i+1} + \Delta G_{\text{dangle } k+1} + \Delta G_{\text{dangle } m+1}$

$V_{3\text{-}44} = W(i+2, i', k+2, k', m+2, m') + W(i'+1, j-1, k'+1, l-1, m'+1, n-2) + 4\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 3\,\Delta G_{\text{MBL closure}} + 3\,\Delta G_{\text{helix terminating in MBL}} + 2\,\Delta G_{gap} + \Delta G_{\text{dangle } i+1} + \Delta G_{\text{dangle } k+1} + \Delta G_{\text{dangle } m+1} + \Delta G_{\text{dangle } n-1}$

$V_{3\text{-}45} = W(i+2, i', k+2, k', m+1, m') + W(i'+1, j-1, k'+1, l-2, m'+1, n-1) + 3\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 3\,\Delta G_{\text{MBL closure}} + 3\,\Delta G_{\text{helix terminating in MBL}} + 3\,\Delta G_{gap} + \Delta G_{\text{dangle } i+1} + \Delta G_{\text{dangle } k+1} + \Delta G_{\text{dangle } l-1}$

$V_{3\text{-}46} = W(i+2, i', k+2, k', m+1, m') + W(i'+1, j-1, k'+1, l-2, m'+1, n-2) + 4\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 3\,\Delta G_{\text{MBL closure}} + 3\,\Delta G_{\text{helix terminating in MBL}} + 2\,\Delta G_{gap} + \Delta G_{\text{dangle } i+1} + \Delta G_{\text{dangle } k+1} + \Delta G_{\text{dangle } l-1} + \Delta G_{\text{dangle } n-1}$

$V_{3\text{-}47} = W(i+2, i', k+2, k', m+2, m') + W(i'+1, j-1, k'+1, l-2, m'+1, n-1) + 4\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 3\,\Delta G_{\text{MBL closure}} + 3\,\Delta G_{\text{helix terminating in MBL}} + 2\,\Delta G_{gap} + \Delta G_{\text{dangle } i+1} + \Delta G_{\text{dangle } k+1} + \Delta G_{\text{dangle } l-1} + \Delta G_{\text{dangle } m+1}$

$V_{3\text{-}48} = W(i+2, i', k+2, k', m+2, m') + W(i'+1, j-1, k'+1, l-2, m'+1, n-2) + 5\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 3\,\Delta G_{\text{MBL closure}} + 3\,\Delta G_{\text{helix terminating in MBL}} + \Delta G_{gap} + \Delta G_{\text{dangle } i+1} + \Delta G_{\text{dangle } k+1} + \Delta G_{\text{dangle } l-1} + \Delta G_{\text{dangle } m+1} + \Delta G_{\text{dangle } n-1}$

$V_{3\text{-}49} = W(i+2, i', k+1, k', m+1, m') + W(i'+1, j-2, k'+1, l-1, m'+1, n-1) + 2\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 3\,\Delta G_{\text{MBL closure}} + 3\,\Delta G_{\text{helix terminating in MBL}} + 4\,\Delta G_{gap} + \Delta G_{\text{dangle } i+1} + \Delta G_{\text{dangle } j-1}$

$$V_{3\text{-}50} = W(i+2, i', k+1, k', m+1, m') + W(i'+1, j-2, k'+1, l-1, m'+1, n-2) + 3\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 3\,\Delta G_{\text{MBL closure}} + 3\,\Delta G_{\text{helix terminating in MBL}} + 3\,\Delta G_{gap} + \Delta G_{\text{dangle } i+1} + \Delta G_{\text{dangle } j-1} + \Delta G_{\text{dangle } n-1}$$

$$V_{3\text{-}51} = W(i+2, i', k+1, k', m+2, m') + W(i'+1, j-2, k'+1, l-1, m'+1, n-1) + 3\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 3\,\Delta G_{\text{MBL closure}} + 3\,\Delta G_{\text{helix terminating in MBL}} + 3\,\Delta G_{gap} + \Delta G_{\text{dangle } i+1} + \Delta G_{\text{dangle } j-1} + \Delta G_{\text{dangle } m+1}$$

$$V_{3\text{-}52} = W(i+2, i', k+1, k', m+2, m') + W(i'+1, j-2, k'+1, l-1, m'+1, n-2) + 4\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 3\,\Delta G_{\text{MBL closure}} + 3\,\Delta G_{\text{helix terminating in MBL}} + 2\,\Delta G_{gap} + \Delta G_{\text{dangle } i+1} + \Delta G_{\text{dangle } j-1} + \Delta G_{\text{dangle } m+1} + \Delta G_{\text{dangle } n-1}$$

$$V_{3\text{-}53} = W(i+2, i', k+1, k', m+1, m') + W(i'+1, j-2, k'+1, l-2, m'+1, n-1) + 3\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 3\,\Delta G_{\text{MBL closure}} + 3\,\Delta G_{\text{helix terminating in MBL}} + 3\,\Delta G_{gap} + \Delta G_{\text{dangle } i+1} + \Delta G_{\text{dangle } j-1} + \Delta G_{\text{dangle } l-1}$$

$$V_{3\text{-}54} = W(i+2, i', k+1, k', m+1, m') + W(i'+1, j-2, k'+1, l-2, m'+1, n-2) + 4\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 3\,\Delta G_{\text{MBL closure}} + 3\,\Delta G_{\text{helix terminating in MBL}} + 2\,\Delta G_{gap} + \Delta G_{\text{dangle } i+1} + \Delta G_{\text{dangle } j-1} + \Delta G_{\text{dangle } l-1} + \Delta G_{\text{dangle } n-1}$$

$$V_{3\text{-}55} = W(i+2, i', k+1, k', m+2, m') + W(i'+1, j-2, k'+1, l-2, m'+1, n-1) + 4\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 3\,\Delta G_{\text{MBL closure}} + 3\,\Delta G_{\text{helix terminating in MBL}} + 2\,\Delta G_{gap} + \Delta G_{\text{dangle } i+1} + \Delta G_{\text{dangle } j-1} + \Delta G_{\text{dangle } l-1} + \Delta G_{\text{dangle } m+1}$$

$$V_{3\text{-}56} = W(i+2, i', k+1, k', m+2, m') + W(i'+1, j-2, k'+1, l-2, m'+1, n-2) + 5\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 3\,\Delta G_{\text{MBL closure}} + 3\,\Delta G_{\text{helix terminating in MBL}} + \Delta G_{gap} + \Delta G_{\text{dangle } i+1} + \Delta G_{\text{dangle } j-1} + \Delta G_{\text{dangle } l-1} + \Delta G_{\text{dangle } m+1} + \Delta G_{\text{dangle } n-1}$$

$$V_{3\text{-}57} = W(i+2, i', k+2, k', m+1, m') + W(i'+1, j-2, k'+1, l-1, m'+1, n-1) + 3\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 3\,\Delta G_{\text{MBL closure}} + 3\,\Delta G_{\text{helix terminating in MBL}} + 3\,\Delta G_{gap} + \Delta G_{\text{dangle } i+1} + \Delta G_{\text{dangle } j-1} + \Delta G_{\text{dangle } k+1}$$

$$V_{3\text{-}58} = W(i+2, i', k+2, k', m+1, m') + W(i'+1, j-2, k'+1, l-1, m'+1, n-2) + 4\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 3\,\Delta G_{\text{MBL closure}} + 3\,\Delta G_{\text{helix terminating in MBL}} + 2\,\Delta G_{gap} + \Delta G_{\text{dangle } i+1} + \Delta G_{\text{dangle } j-1} + \Delta G_{\text{dangle } k+1} + \Delta G_{\text{dangle } n-1}$$

$$V_{3\text{-}59} = W(i+2, i', k+2, k', m+2, m') + W(i'+1, j-2, k'+1, l-1, m'+1, n-1) + 4\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 3\,\Delta G_{\text{MBL closure}} + 3\,\Delta G_{\text{helix terminating in MBL}} + 2\,\Delta G_{gap} + \Delta G_{\text{dangle } i+1} + \Delta G_{\text{dangle } j-1} + \Delta G_{\text{dangle } k+1} + \Delta G_{\text{dangle } m+1}$$

$$V_{3\text{-}60} = W(i+2, i', k+2, k', m+2, m') + W(i'+1, j-2, k'+1, l-1, m'+1, n-2) + 5\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 3\,\Delta G_{\text{MBL closure}} + 3\,\Delta G_{\text{helix terminating in MBL}} + \Delta G_{gap} + \Delta G_{\text{dangle } i+1} + \Delta G_{\text{dangle } j-1} + \Delta G_{\text{dangle } k+1} + \Delta G_{\text{dangle } m+1} + \Delta G_{\text{dangle } n-1}$$

$$V_{3\text{-}61} = W(i+2, i', k+2, k', m+1, m') + W(i'+1, j-2, k'+1, l-2, m'+1, n-1) + 4\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 3\,\Delta G_{\text{MBL closure}} + 3\,\Delta G_{\text{helix terminating in MBL}} + 2\,\Delta G_{gap} + \Delta G_{\text{dangle } i+1} + \Delta G_{\text{dangle } j-1} + \Delta G_{\text{dangle } k+1} + \Delta G_{\text{dangle } l-1}$$

$$V_{3\text{-}62} = W(i+2, i', k+2, k', m+1, m') + W(i'+1, j-2, k'+1, l-2, m'+1, n-2) + 5\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 3\,\Delta G_{\text{MBL closure}} + 3\,\Delta G_{\text{helix terminating in MBL}} + \Delta G_{gap} + \Delta G_{\text{dangle } i+1} + \Delta G_{\text{dangle } j-1} + \Delta G_{\text{dangle } k+1} + \Delta G_{\text{dangle } l-1} + \Delta G_{\text{dangle } n-1}$$

$$V_{3\text{-}63} = W(i+2, i', k+2, k', m+2, m') + W(i'+1, j-2, k'+1, l-2, m'+1, n-1) + 5\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 3\,\Delta G_{\text{MBL closure}} + 3\,\Delta G_{\text{helix terminating in MBL}} + \Delta G_{gap} + \Delta G_{\text{dangle } i+1} + \Delta G_{\text{dangle } j-1} + \Delta G_{\text{dangle } k+1} + \Delta G_{\text{dangle } l-1} + \Delta G_{\text{dangle } m+1}$$

$$V_{3\text{-}64} = W(i+2, i', k+2, k', m+2, m') + W(i'+1, j-2, k'+1, l-2, m'+1, n-2) + 6\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 3\,\Delta G_{\text{MBL closure}} + 3\,\Delta G_{\text{helix terminating in MBL}} + \Delta G_{\text{dangle } i+1} + \Delta G_{\text{dangle } j-1} + \Delta G_{\text{dangle } k+1} + \Delta G_{\text{dangle } l-1} + \Delta G_{\text{dangle } m+1} + \Delta G_{\text{dangle } n-1}$$

W_1 represents adding unpaired nucleotides to a multibranch loop.

$$W_{1\text{-}2} = W(i, j, k, l, m, n\text{-}1) + \Delta G_{\text{unpaired nucleotides in MBL loop}} + 2\,\Delta G_{gap}$$

$$W_{1\text{-}3} = W(i, j, k, l, m\text{-}1, n) + \Delta G_{\text{unpaired nucleotides in MBL loop}} + 2\,\Delta G_{gap}$$

$$W_{1\text{-}4} = W(i, j, k, l, m\text{-}1, n\text{-}1) + 2\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 4\,\Delta G_{gap}$$

$$W_{1\text{-}5} = W(i, j, k, l\text{-}1, m, n) + \Delta G_{\text{unpaired nucleotides in MBL loop}} + 2\,\Delta G_{gap}$$

$$W_{1\text{-}6} = W(i, j, k, l\text{-}1, m, n\text{-}1) + 2\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + \Delta G_{gap}$$

$$W_{1\text{-}7} = W(i, j, k, l\text{-}1, m\text{-}1, n) + 2\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 4\,\Delta G_{gap}$$

$$W_{1\text{-}8} = W(i, j, k, l\text{-}1, m\text{-}1, n\text{-}1) + 3\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 3\,\Delta G_{gap}$$

$$W_{1\text{-}9} = W(i, j, k\text{-}1, l, m, n) + \Delta G_{\text{unpaired nucleotides in MBL loop}} + 2\,\Delta G_{gap}$$

$$W_{1\text{-}10} = W(i, j, k\text{-}1, l, m, n\text{-}1) + 2\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 4\,\Delta G_{gap}$$

$$W_{1\text{-}11} = W(i, j, k\text{-}1, l, m\text{-}1, n) + 2\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + \Delta G_{gap}$$

$$W_{1\text{-}12} = W(i, j, k\text{-}1, l, m\text{-}1, n\text{-}1) + 3\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 3\,\Delta G_{gap}$$

$$W_{1\text{-}13} = W(i, j, k\text{-}1, l\text{-}1, m, n) + 2\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 4\,\Delta G_{gap}$$

$$W_{1\text{-}14} = W(i, j, k\text{-}1, l\text{-}1, m, n\text{-}1) + 3\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 3\,\Delta G_{gap}$$

$$W_{1\text{-}15} = W(i, j, k\text{-}1, l\text{-}1, m\text{-}1, n) + 3\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 3\,\Delta G_{gap}$$

$$W_{1\text{-}16} = W(i, j, k\text{-}1, l\text{-}1, m\text{-}1, n\text{-}1) + 4\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 2\,\Delta G_{gap}$$

$$W_{1\text{-}17} = W(i, j\text{-}1, k, l, m, n) + \Delta G_{\text{unpaired nucleotides in MBL loop}} + 2\,\Delta G_{gap}$$

$$W_{1\text{-}18} = W(i, j\text{-}1, k, l, m, n\text{-}1) + 2\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + \Delta G_{gap}$$

$$W_{1\text{-}19} = W(i, j\text{-}1, k, l, m\text{-}1, n) + 2\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 4\,\Delta G_{gap}$$

$$W_{1\text{-}20} = W(i, j\text{-}1, k, l, m\text{-}1, n\text{-}1) + 3\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 3\,\Delta G_{gap}$$

$$W_{1\text{-}21} = W(i, j\text{-}1, k, l\text{-}1, m, n) + 2\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + \Delta G_{gap}$$

$$W_{1\text{-}22} = W(i, j\text{-}1, k, l\text{-}1, m, n\text{-}1) + 3\,\Delta G_{\text{unpaired nucleotides in MBL loop}}$$

$$W_{1\text{-}23} = W(i, j\text{-}1, k, l\text{-}1, m\text{-}1, n) + 3\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 3\,\Delta G_{gap}$$

$$W_{1\text{-}24} = W(i, j\text{-}1, k, l\text{-}1, m\text{-}1, n\text{-}1) + 4\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 2\,\Delta G_{gap}$$

$$W_{1\text{-}25} = W(i, j\text{-}1, k\text{-}1, l, m, n) + 2\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 4\,\Delta G_{gap}$$

$$W_{1\text{-}26} = W(i, j\text{-}1, k\text{-}1, l, m, n\text{-}1) + 3\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 3\,\Delta G_{gap}$$

$$W_{1\text{-}27} = W(i, j\text{-}1, k\text{-}1, l, m\text{-}1, n) + 3\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 3\,\Delta G_{gap}$$

$$W_{1\text{-}28} = W(i, j\text{-}1, k\text{-}1, l, m\text{-}1, n\text{-}1) + 4\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 2\,\Delta G_{gap}$$

$$W_{1\text{-}29} = W(i, j\text{-}1, k\text{-}1, l\text{-}1, m, n) + 3\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 3\,\Delta G_{gap}$$

$$W_{1\text{-}30} = W(i, j\text{-}1, k\text{-}1, l\text{-}1, m, n\text{-}1) + 4\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 2\,\Delta G_{gap}$$

$$W_{1\text{-}31} = W(i, j\text{-}1, k\text{-}1, l\text{-}1, m\text{-}1, n) + 4\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 2\,\Delta G_{gap}$$

$$W_{1\text{-}32} = W(i, j\text{-}1, k\text{-}1, l\text{-}1, m\text{-}1, n\text{-}1) + 5\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + \Delta G_{gap}$$

$$W_{1\text{-}33} = W(i\text{-}1, j, k, l, m, n) + \Delta G_{\text{unpaired nucleotides in MBL loop}} + 2\,\Delta G_{gap}$$

$$W_{1\text{-}34} = W(i\text{-}1, j, k, l, m, n\text{-}1) + 2\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 4\,\Delta G_{gap}$$

$$W_{1\text{-}35} = W(i\text{-}1, j, k, l, m\text{-}1, n) + 2\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + \Delta G_{gap}$$

$$W_{1\text{-}36} = W(i\text{-}1, j, k, l, m\text{-}1, n\text{-}1) + 3\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 3\,\Delta G_{gap}$$

$$W_{1\text{-}37} = W(i\text{-}1, j, k, l\text{-}1, m, n) + 2\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 4\,\Delta G_{gap}$$

$$W_{1\text{-}38} = W(i\text{-}1, j, k, l\text{-}1, m, n\text{-}1) + 3\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 3\,\Delta G_{gap}$$

$$W_{1\text{-}39} = W(i\text{-}1, j, k, l\text{-}1, m\text{-}1, n) + 3\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 3\,\Delta G_{gap}$$

$$W_{1\text{-}40} = W(i\text{-}1, j, k, l\text{-}1, m\text{-}1, n\text{-}1) + 4\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 2\,\Delta G_{gap}$$

$$W_{1\text{-}41} = W(i\text{-}1, j, k\text{-}1, l, m, n\text{-}1) + 2\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + \Delta G_{gap}$$

$$W_{1\text{-}42} = W(i\text{-}1, j, k\text{-}1, l, m, n\text{-}1) + 3\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 3\,\Delta G_{gap}$$

$$W_{1\text{-}43} = W(i\text{-}1, j, k\text{-}1, l, m\text{-}1, n) + 3\,\Delta G_{\text{unpaired nucleotides in MBL loop}}$$

$$W_{1\text{-}44} = W(i\text{-}1, j, k\text{-}1, l, m\text{-}1, n\text{-}1) + 4\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 2\,\Delta G_{gap}$$

$$W_{1\text{-}45} = W(i\text{-}1, j, k\text{-}1, l\text{-}1, m, n) + 3\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 3\,\Delta G_{gap}$$

$$W_{1\text{-}46} = W(i\text{-}1, j, k\text{-}1, l\text{-}1, m, n\text{-}1) + 4\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 2\,\Delta G_{gap}$$

$$W_{1\text{-}47} = W(i\text{-}1, j, k\text{-}1, l\text{-}1, m\text{-}1, n) + 4\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 2\,\Delta G_{gap}$$

$$W_{1\text{-}48} = W(i\text{-}1, j, k\text{-}1, l\text{-}1, m\text{-}1, n\text{-}1) + 5\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + \Delta G_{gap}$$

$$W_{1\text{-}49} = W(i\text{-}1, j\text{-}1, k, l, m, n) + 2\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 4\,\Delta G_{gap}$$

$$W_{1\text{-}50} = W(i\text{-}1, j\text{-}1, k, l, m, n\text{-}1) + 3\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 3\,\Delta G_{gap}$$

$$W_{1\text{-}51} = W(i\text{-}1, j\text{-}1, k, l, m\text{-}1, n) + 3\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 3\,\Delta G_{gap}$$

$$W_{1\text{-}52} = W(i\text{-}1, j\text{-}1, k, l, m\text{-}1, n\text{-}1) + 4\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 2\,\Delta G_{gap}$$

$$W_{1\text{-}53} = W(i\text{-}1, j\text{-}1, k, l\text{-}1, m, n) + 3\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 3\,\Delta G_{gap}$$

$$W_{1\text{-}54} = W(i\text{-}1, j\text{-}1, k, l\text{-}1, m, n\text{-}1) + 4\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 2\,\Delta G_{gap}$$

$$W_{1\text{-}55} = W(i\text{-}1, j\text{-}1, k, l\text{-}1, m\text{-}1, n) + 4\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 2\,\Delta G_{gap}$$

$$W_{1\text{-}56} = W(i\text{-}1, j\text{-}1, k, l\text{-}1, m\text{-}1, n\text{-}1) + 5\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + \Delta G_{gap}$$

$$W_{1\text{-}57} = W(i\text{-}1, j\text{-}1, k\text{-}1, l, m, n) + 3\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 3\,\Delta G_{gap}$$

$$W_{1\text{-}58} = W(i\text{-}1, j\text{-}1, k\text{-}1, l, m, n\text{-}1) + 4\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 2\,\Delta G_{gap}$$

$$W_{1\text{-}59} = W(i\text{-}1, j\text{-}1, k\text{-}1, l, m\text{-}1, n) + 4\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 2\,\Delta G_{gap}$$

$$W_{1\text{-}60} = W(i\text{-}1, j\text{-}1, k\text{-}1, l, m\text{-}1, n\text{-}1) + 5\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + \Delta G_{gap}$$

$$W_{1\text{-}61} = W(i\text{-}1, j\text{-}1, k\text{-}1, l\text{-}1, m, n) + 4\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + 2\,\Delta G_{gap}$$

$$W_{1\text{-}62} = W(i\text{-}1, j\text{-}1, k\text{-}1, l\text{-}1, m, n\text{-}1) + 5\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + \Delta G_{gap}$$

$$W_{1\text{-}63} = W(i\text{-}1, j\text{-}1, k\text{-}1, l\text{-}1, m\text{-}1, n) + 5\,\Delta G_{\text{unpaired nucleotides in MBL loop}} + \Delta G_{gap}$$

$$W_{1\text{-}64} = W(i\text{-}1, j\text{-}1, k\text{-}1, l\text{-}1, m\text{-}1, n\text{-}1) + 6\,\Delta G_{\text{unpaired nucleotides in MBL loop}}$$

W_2 accounts for helix termini.

$W_2 = V(a, b, c, d, e, f) + x\ \Delta G_{unpaired\ nucleotides\ in\ MBL\ loop} + y\ \Delta G_{gap} + 3\ \Delta G_{MBL\ closure}$

W_3 accounts for bifurcations in the structure.

$W_3 = min\ (\ W(i, i', k, k', m, m') + W(i'+1, j, k'+1, l, m'+1, n)\ )$

$W9(i,k,m)$ is the lowest free energy sum for the prefix alignment of the nucleotide fragments from 1 to i in the first sequence, 1 to k in the second sequence, and 1 to m in the third sequence.

$W9_1 = V(i', d, k', e, m', f) + W9(a, b, c) + x\ \Delta G_{unpaired\ nucleotide\ in\ MBL\ loop} + 3\ \Delta G_{MBL\ closure} + y\ \Delta G_{gap} + \Delta G(dangling\ ends)$

$W9_2 = W9(i, k, m-1) + 2\ \Delta Ggap$

$W9_3 = W9(i, k-1, m) + 2\ \Delta Ggap$

$W9_4 = W9(i, k-1, m-1) + \Delta Ggap$

$W9_5 = W9(i-1, k, m) + 2\ \Delta Ggap$

$W9_6 = W9(i-1, k, m-1) + \Delta Ggap$

$W9_7 = W9(i-1, k-1, m) + \Delta Ggap$

$W9_8 = W9(i-1, k-1, m-1) + \Delta Ggap$

REFERENCES

References

1. *Comparative RNA web site retrieved from http://ww.rna.icmb.utexas.edu.* 2004.
2. *Sanger Institute retrieved from http://www.sanger.ac.uk/.* 2004.
3. *Small-molecule gene therapy nearing trials retrieved from http://www.DrugResearcher.com.* 2004.
4. Barciszewski, J. and B.F.C. Clark, *RNA Biochemistry and Biotechnology.* 1998: Kluwer. 370.
5. Bengert, P., et al., *RNA motifs and regulatory elements.* 2002: Springer. 233.
6. Chen, J.H., S.Y. Le, and J.V. Maizel, *Prediction of common secondary structures of RNAs: a genetic algorithm approach.* Nucleic Acids Res, 2000. **28**(4): p. 991-999.
7. Doshi, K.J., et al., *Evaluation of the suitability of free-energy minimization using nearest-neighbor energy parameters for RNA secondary structure prediction.* BMC Bioinformatics, 2004. **5**(1): p. 105.
8. Dumas, J.P. and J. Ninio, *Efficient algorithms for folding and comparing nucleic acid sequences.* Nucleic Acids Res, 1982. **10**(1): p. 197-206.
9. Eggleston, D.S., C.D. Prescott, and N.D. Pearson, *The many faces of RNA.* 1998: Academic Press. 228.
10. Freier, S.M., et al., *Improved free-energy parameters for predictions of RNA duplex stability.* Proc Natl Acad Sci U S A, 1986. **83**(24): p. 9373-9377.
11. Gardner, P.P. and R. Giegerich, *A comprehensive comparison of comparative RNA structure prediction approaches.* BMC Bioinformatics, 2004. **5**(1): p. 140.
12. Gesteland, R.F., T.R. Cech, and J.F. Atkins, *The RNA world: the nature of modern RNA suggests a prebiotic RNA.* Cold Spring Harbor monograph series, 0270-1847 ; monograph 37. 1999: Cold Spring Harbor Laboratory Press. 709.
13. Gorodkin, J., S.L. Stricklin, and G.D. Stormo, *Discovering common stem-loop motifs in unaligned RNA sequences.* Nucleic Acids Res, 2001. **29**(10): p. 2135-2144.

14. Kumar, S. and A. Rzhetsky, *Evolutionary relationships of eukaryotic kingdoms.* J Mol Evol, 1996. **42**(2): p. 183-193.

15. Lyngso, R.B., M. Zuker, and C.N. Pedersen, *Fast evaluation of internal loops in RNA secondary structure prediction.* Bioinformatics, 1999. **15**(6): p. 440-445.

16. Masoumi, B. and M. Turcotte. *Simultaneous Alignment and Structure prediction of RNAs: Are Three input sequences better than Two?* in *International Conference on Computational Science (ICCS 2005)*. 2005. Atlanta, USA, p. 936-943.

17. Masoumi, B. and M. Turcotte, *Simultaneous alignment and structure prediction of three RNA sequences.* International Journal of Bioinformatics Research and Applications, In Press - 2005.

18. Mathews, D.H., et al., *Expanded sequence dependence of thermodynamic parameters improves prediction of RNA secondary structure.* J Mol Biol, 1999. **288**(5): p. 911-940.

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

20. N.K. Verma and C.S. Dey, *RNA-mediated gene silencing: mechanisms and its therapeutic applications.* Journal of clinical pharmacy and therapeutics, 2004(29): p. 395-404.

21. Neidle, S., *Nucleic Acid structure and recognition.* 2002: Oxford University Press. 188.

22. Rivas, E. and S.R. Eddy, *The language of RNA: a formal grammar that includes pseudoknots.* Bioinformatics, 2000. **16**(4): p. 334-340.

23. Sankoff, D., *Simultaneous solution of RNA folding, alignment and protosequence problems.* Siam J. Appl. Math., 1985. **45**(5): p. 810-825.

24. Sankoff, D., et al., *Fast algorithms to determine RNA secondary structures containing multiple loops*, in *Time warps, string edits, and macromolecules : [theory and practice of sequence comparison]*. 1999, Center for the Study of Language and Information. p. 382.

25. Sensen, C.W., *Essentials of genomics and bioinformatics.* 2002: Wiley-VCH. 419.

26. Serra, M.J. and D.H. Turner, *Predicting thermodynamic properties of RNA.* Methods Enzymol, 1995. **259**: p. 242-261.

27. Soler, F. and K. Jankowski, *Modeling RNA secondary structures. II. The geometric structural solution for tRNA.* Math Biosci, 1991. **105**(2): p. 191-206.

28. T.Xia, D.H. Mathews, and D.H. Turner, *Thermodynamics of RNA secondary structure formation*, in *COMPREHENSIVE NATURAL PRODUCTS CHEMISTRY*, D.H.R. Barton, K. Nakanishi, and O. Meth-Cohn, Editors. 1999, Elsevier. p. 337.

29. Xia, T., et al., *Thermodynamic parameters for an expanded nearest-neighbor model for formation of RNA duplexes with Watson-Crick base pairs.* Biochemistry, 1998. **37**(42): p. 14719-14735.

30. Zuker, M., *Computer prediction of RNA structure.* Methods Enzymol, 1989. **180**: p. 262-288.

31. Zuker, M. and P. Stiegler, *Optimal computer folding of large RNA sequences using thermodynamics and auxiliary information.* Nucleic Acids Res, 1981. **9**(1): p. 133-148.