EXPLORING RNA AND PROTEIN 3D STRUCTURES BY GEOMETRIC ALGORITHMS

Xueyi Wang

A dissertation submitted to the faculty of the University of North Carolina at Chapel Hill in partial fulfillment of the requirements for the degree of Doctor of Philosophy in the Department of Computer Science.

Chapel Hill
2008

Approved by

Advisor: Jack Snoeyink

Reader: Jane Richardson

Reader: Wei Wang

Reader: Jan Prins

Reader: Alexander Tropsha

Reader: Nikolay Dokholyan

ABSTRACT

XUEYI WANG: Exploring RNA and Protein 3D Structures by Geometric Algorithms
(Under the direction of Jack Snoeyink)

Many problems in RNA and protein structures are related with their specific geometric properties. Geometric algorithms can be used to explore the possible solutions of these problems. This dissertation investigates the geometric properties of RNA and protein structures and explores three different ways that geometric algorithms can help to the study of the structures.

Determine accurate structures. Accurate details in RNA structures are important for understanding RNA function, but the backbone conformation is difficult to determine and most existing RNA structures show serious steric clashes ($\geq$ 0.4 Å overlap). I developed a program called RNABC (RNA Backbone Correction) that searches for alternative clash-free conformations with acceptable geometry. It rebuilds a suite (unit from sugar to sugar) by anchoring phosphorus and base positions, which are clearest in crystallographic electron density, and reconstructing other atoms using forward kinematics and conjugate gradient methods. Two tests show that RNABC improves backbone conformations for most problem suites in S-motifs and for many of the worst problem suites identified by members of the Richardson lab.

Display structure commonalities. Structure alignment commonly uses root mean squared distance (RMSD) to measure the structural similarity. I first extend RMSD to

weighted RMSD (wRMSD) for multiple structures and show that using wRMSD with multiplicative weights implies the average is a consensus structure. Although I show that finding the optimal translations and rotations for minimizing wRMSD cannot be decoupled for multiple structures, I develop a near-linear iterative algorithm to converge to a local minimum of wRMSD. Finally I propose a heuristic algorithm to iteratively reassign weights to reduce the effect of outliers and find well-aligned positions that determine structurally conserved regions.

Distinguish local structural features. Identifying common motifs (fragments of structures common to a group of molecules) is one way to further our understanding of the structure and function of molecules. I apply a graph database mining technique to identify RNA tertiary motifs. I abstract RNA molecules as labeled graphs, use a frequent subgraph mining algorithm to derive tertiary motifs, and present an iterative structure alignment algorithm to classify tertiary motifs and generate consensus motifs. Tests on ribosomal and transfer RNA families show that this method can identify most known RNA tertiary motifs in these families and suggest candidates for novel tertiary motifs.

ACKNOWLEDGEMENTS

I am in debt to my advisor Professor Jack Snoeyink, who kindly guides me through the five great years at UNC Chapel Hill, who shows great love, thoughtfulness, integrity, scrupulousness and inspiration to me, and whom I always respect and admire.

I am also in debt to my collaborators Professors Jane & David Richardson, whose earnest, passionate, encouraging, assiduous and cheerful attitudes I will benefit in my life, and Professor Wei Wang, whose amiable, sincere and diligent attitudes I will learn always.

I would like to express my appreciation to my committee members Professors Alexander Tropsha, Jan Prins and Nikolay Dokholyan, whose encouragement, creative- and critical-thinking have been a great benefit to me.

I would like to thank my other collaborators, Michael Word, Andrew Leaver-Fay, Jun Huan, Yuanxin Liu, Gary Kapral, Laura Murray, Ian Davis, and Kevin Weeks, for their selfless contributions and genuine help.

I also would like to give sincere thanks to my friends Bryan Arendall, Deepak Bandyopadhyay, Jeff Headd, Bob Immormino, Dan Keedy, Lizbeth Videau, Liangjun Zhang, and Qi Zhang for all kinds of help in my research and dissertation writing.

Last, I dedicate this dissertation to my parents Pengfei Wang and Baodi Zhu, without whom I will not be here, and to my wife Junling Nie, who loves and supports me and is my beloved.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

Computational geometry studies the design and analysis of algorithms for problems that are best stated in geometric form. The applications of geometric algorithms include computer graphics, computer-aided design and manufacturing, robotics, geographic information systems (GIS), and computational biology. The challenges for designing a geometric algorithm include how to represent a problem in terms of geometry, how to correctly obtain related geometric properties from the problem, and how to effectively build an algorithm to solve the problem by exploring its geometric properties.

This dissertation focuses on using geometric algorithms to solve problems in RNA and protein structures. I present three works that abstract geometric properties of RNA and protein structures at different scales. In the following subsections, I first introduce the RNA and protein molecules and then discuss applications of geometric algorithms in RNA and protein structures.

1.1 RNA and Protein Molecules

RNA and protein molecules are essential for life. Protein carries out many crucial biological functions in organisms [Voet01]. Protein catalyzes most metabolic reactions in organisms [Johnson74, Voet01], forms scaffolds to bring other proteins together or maintain

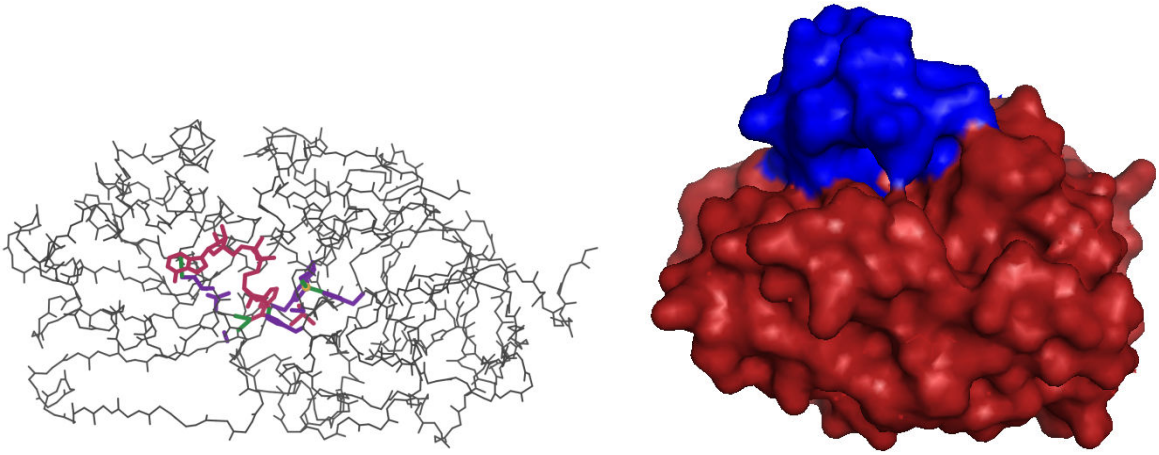
cell shape [Faux96, Shih06], stores and transports ions and other molecules [Weber01, Long05], decodes and transmits genetic information [Latchman97, Dame05], and plays other important roles such as cell signaling [Lin04, Mohamed05], immune responses [Roux99, Diaz02], cell adhesion [White97, Wilson01], and the cell cycle [Nigg95, Bates98].

RNA also plays many important roles in organisms, with new ones being discovered constantly [Soukup04, Nielson05, Salehi-Ashtiani06]. RNA stores and transmits genetic information [Crick70, Sussman76, Lolle05], provides and regulates molecular-binding interactions [Huang03, Lukavsky03, Mattick01], maintains chromosome length [Chen04], controls metabolic processes [Winkler02, Serganov06], and catalyzes chemical reactions [Nissen00, Lilley05, Klein06]. RNA plays a central role in all aspects of gene expression and its control [Claverie05], such as performing and regulating RNA interference [Tomari05], co-suppression and silencing [Mattick01], and especially splicing and alternative splicing of exons [Nilsen94, Murray99, Stahley05].

1.2 Geometric Algorithms for RNA and Protein Structures

The function of RNA and protein molecules is often closely related to the geometric arrangement of atoms. Understanding the details of the 3D structures of RNA and protein molecules is often a key to understanding their function. For example, the structures in figure 1 show clearly which atoms are interacting — information that is not available from the protein and RNA sequences. In figure 1.1a, NAD+ molecule (in place of alcohol molecule) binds to alcohol dehydrogenase, where a zinc atom and amino acids Cys-46, Ser-48, His-51, His-67, Cys-174, Ile-269, Val-292, Ala-317, and Phe-319 are involved to construct the active site [Hammes-Schiffer06], and in figure 1.1b, trypsin binds to its inhibitor to prevent trypsin

from breaking down other proteins [Conners07].



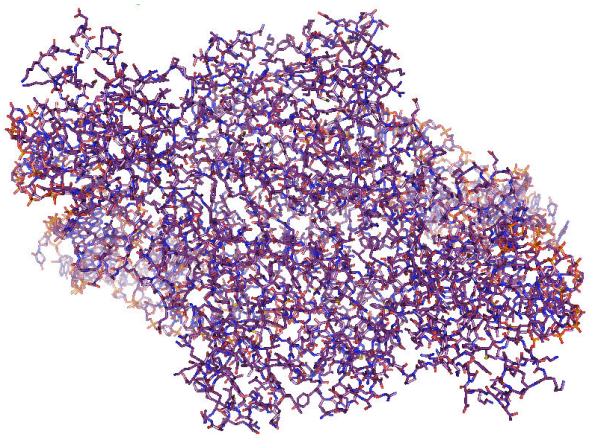
a. Structure of alcohol dehydrogenase catalyzing alcohols to aldehydes/ketones (black colored is backbone, purple is sidechain, red is NAD^+, green is hydrogen bond, and yellow is a zinc atom)

b. Molecular surfaces of trypsin and its inhibitor (red colored is trypsin and blue is inhibitor)

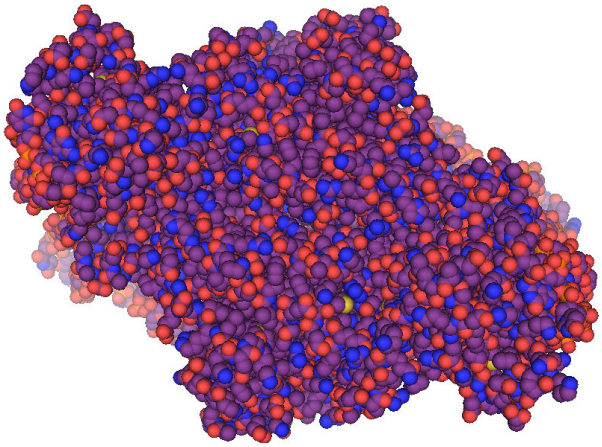
Figure 1.1 Examples of protein catalysis and protein-protein interaction

RNA and protein structures can be represented as different geometric models by abstracting the geometry at different scales. I can represent each atom as a point and each covalent bond between two atoms as an edge to build a ball-and-stick model (Figure 1.2a), represent each atom as a ball with van der Waals radius to build a space filling model (Figure 1.2b), represent accessible regions of all atoms as a surface to build a molecular surface model (Figure 1.2c), or represent each residue (an amino acid of protein or a nucleotide of RNA) as a point and the connectivity between two residues (covalent bond, hydrogen bond or spatial distance) as an edge to build a topological model (Figure 1.2d).

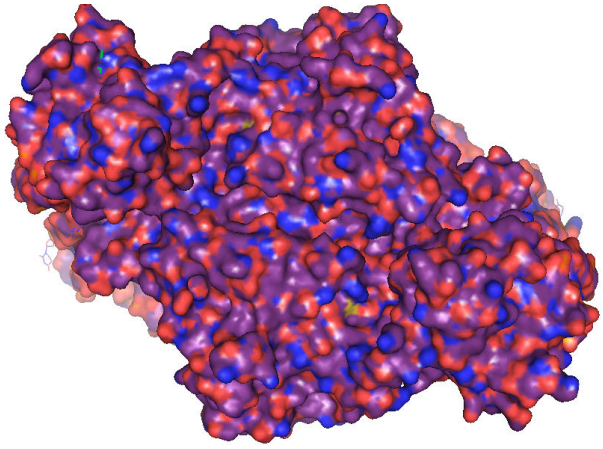
By abstracting the macromolecular structures as geometric models, geometric algorithms can be applied to analyze or simulate many functions of RNA and protein molecules. Examples include structure alignment and structure prediction.



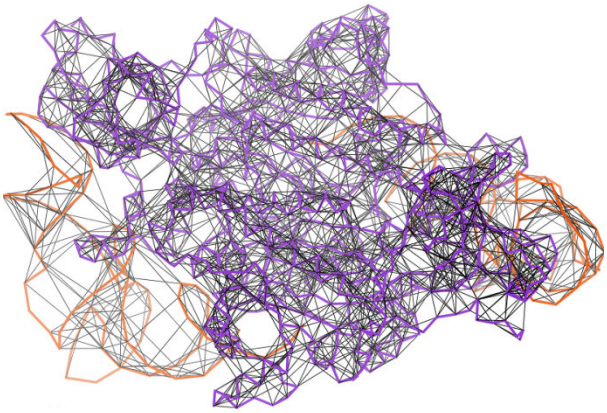
a. Ball and stick model



b. Space filling model



c. Molecular surface model



d. Contact edge model

Figure 1.2 Models of yeast aspartyl-tRNA synthetase (In a, b and c, purple colored is carbon, red is oxygen, blue is nitrogen, and brown is phosphorus. In d, purple colored is protein backbone, brown is RNA backbone, and black is contact edge)

Structure alignment explores the similarity of two or more molecules based on their 3D structures and is useful to query databases for similar structures, perform all-to-all structure comparison, detect dissimilar structure, determine structurally conserved regions, and calculate structure-based phylogenetic trees for RNA and protein families. There are two tasks in structure alignment: the first is to establish correspondence between atoms in different structures; the second is to transform all structures to minimize an alignment score

function (e.g. minimize the root mean squared distances for all atom pairs). Generally, establishing the correspondence is harder than optimizing the alignment. Most alignment methods regard each structure as a rigid body and allow only rotation and translation of the structure. Structure alignment methods have two categories: *pairwise structure alignment* aligns two structures and existing programs include DALI [Holm96] and MAMMOTH [Ortiz02]; *multiple structure alignment* aligns more than two structures and existing programs include MULTAL [Taylor94], CE [Guda01] and MUSTA [Leibowitz01].

Structure prediction is one of the most important problems in bioinformatics and is of great importance in medicine and biotechnology, e.g. drug design and novel enzyme design. The goal is to predict 3D structure of an RNA or protein molecule from its sequence. Structure prediction searches the space of possible structures and identifies the most probable structure by minimizing an energy function. The predicted structure is subject to many geometric constraints, such as preserving the covalent bonds between atoms, limiting the lengths of the covalent bond and the angles of contiguous bonds in small ranges to canonical values, and preventing remote atoms from getting too close.

Structure prediction methods have three categories: homology modeling, threading, and *ab initio* modeling. Homology modeling builds a structure from known structures having similar sequences (e.g. > 30% sequence identity for proteins), based on the assumption that similar sequences deliver similar structures. Examples of programs include SWISS-MODEL [Schwede03] and MODELLER [Fiser03]. Threading is based on the observation that there are only a limited number of distinct folds and an unknown protein structure is very likely similar to a known protein structure, although their sequence similarities are low (e.g. < 30% sequence identity for protein). It searches a database of known structures to find a structure

whose sequence may be compatible to the sequence of an unknown structure and optimizes the structure. Examples of threading programs include 3D-PSSM [Kelley00] and 3D-Jury [Ginalski03]. *Ab initio* modeling builds 3D molecular structure without reference to existing structures. It is considered the hardest method and the examples of programs include ROSETTA [Bonneau01] and TOUCHSTONE [Zhang03].

In this dissertation, I present three works that apply geometric algorithms in RNA and protein structures. These three works abstract the geometric properties of RNA and protein structures at different scales: 1) finding alternative clash-free conformations with acceptable geometry for RNA crystal structures, which focuses on the atomic details of RNA structures, 2) optimizing multiple structure alignment, which focuses on both local and global rigid geometry of RNA and protein structures, and 3) mining RNA tertiary motifs, which focuses on the topological geometry of RNA structures.

In Chapter 2, I review the biochemical properties of macromolecules, especially the RNA molecules, and underline the geometric properties of macromolecules that are related to my works. In Chapter 3, I present a program called RNABC to find alternative clash-free conformations with acceptable geometry for RNA crystal structures [Wang08a]. In Chapter 4, I extend (RMSD) to weighted RMSD for multiple structure alignment and present two algorithms to optimize gapped multiple structure alignment and find structurally conserved regions [Wang06, Wang07b, Wang08b]. In Chapter 5, I propose a novel application of graph database mining to identify RNA tertiary motifs [Wang07a]. In Chapter 6, I summarize the results and discuss future research.

CHAPTER 2

RNA AND PROTEIN STRUCTURES

In this chapter, I discuss the basic biochemical properties of the RNA and protein structures and the methods to obtain and evaluate their structures.

2.1 Biochemical Properties

2.1.1 Atoms and Chemical Bonds

Hydrogen (H), carbon (C), nitrogen (N), oxygen (O), sulfur (S) and phosphorus (P) are the six most abundant atoms in RNA and protein molecules (sulfur occurs mostly in protein and phosphorus occurs mostly in RNA). RNA and protein molecules can also bond with some metal ions such as magnesium (Mg), zinc (Zn), and iron (Fe). All atoms other than hydrogen may be called heavy atoms.

Various chemical bonds hold RNA and protein structures together. The common chemical bonds in RNA and protein structures include covalent bonds, ionic bonds, and hydrogen bonds. A *covalent bond* binds atoms together through the sharing of electron pairs between atoms. Most covalent bonds in RNA and protein structures are single and double bonds, where atoms share one and two pairs of electron. When one atom in a covalent bond has a greater affinity for the electrons, then this atom is called *electronegative* and the bond is called a polar covalent bond; examples include O-H, N-H, and S-H. Otherwise, the bond is

called a non-polar covalent bond; examples include C-C, O-O and H-H. The typical bond length for a covalent bond with heavy atoms is 1.5Å and for a covalent bond with hydrogen atom is 1.1Å. The typical bond angle formed by two contiguous covalent bonds is 109°. An *ionic bond* often forms between metal and non-metal ions. Metal ions in RNA and protein structures form ion bonds with several types of non-metal atoms, such as nitrogen, oxygen, sulfur and phosphorus. The length and angle of ionic bonds vary. A *hydrogen bond* is a special type of bond formed between an electronegative atom (e.g. nitrogen, oxygen, sulfur or phosphorus) and a hydrogen atom bonded with another electronegative atom. A hydrogen atom bound to a carbon atom (i.e. C-H bond) normally cannot form a hydrogen bond with other atoms because the difference of electron affinity for carbon and hydrogen atoms is small and C-H bond is normally considered non-polar. The normal length of a hydrogen bond is 1.97Å.

2.1.2 Protein

A protein molecule is a linear polymer of 20 different amino acids. Protein has four levels of structural organization: *Primary structure* is the linear sequence of amino acids, *secondary structure* is the common recurring patterns of inter-residue interactions, including α-helix and β-sheet, *tertiary structure* is the overall shape of a protein molecule, and *quaternary structure* is the organization of two or more protein molecules. The primary structure is formed by the covalent bonds, whereas the secondary, tertiary and quaternary structures are formed mainly by hydrogen bonds and ionic bonds.

Each amino acid contains three components: amino group, carboxyl acid group and R group (sidechain) attaching to the Cα atom, as shown in Figure 2.1. For the 20 amino acids,

the amino and carboxyl groups are the same (except for Proline) and the R groups differ. The amino groups and carboxyl groups from contiguous amino acids are joined by peptide bonds (with the removal of water molecules) and form the protein backbone.

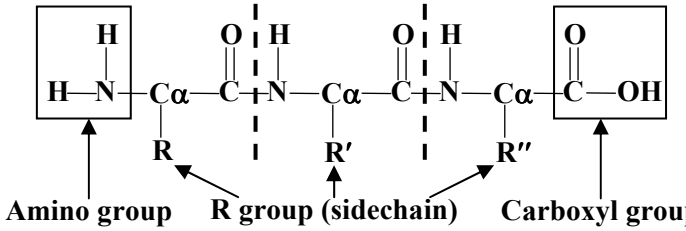


Figure 2.1 A sequence of three amino acids joined by peptide bonds, showing amino, carboxyl and R groups attaching to the α-carbons

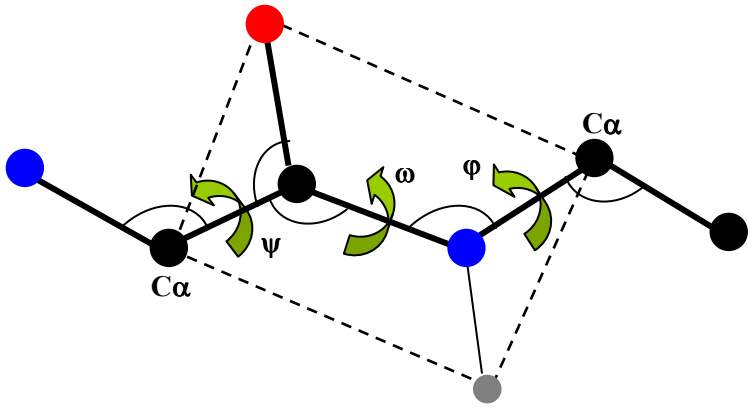


Figure 2.2 Bond lengths, angles and dihedral angles in the amino acids. Black colored atoms are carbon, red is oxygen, blue is nitrogen, and gray is hydrogen (not all hydrogens are shown). For heavy atoms (i.e. non-hydrogen atoms), thick solid lines are bond lengths, curves are angles and arrow curves are dihedral angles

In a protein structure, *bond lengths* (for covalent bonds) and *bond angles* (by two contiguous covalent bonds) are relatively rigid — only limited flexibility is allowed, but *dihedral angles* (formed by three contiguous covalent bonds) are flexible. Figure 2.2 shows the bond lengths, angles and dihedral angles in protein backbone. Each amino acid backbone has three dihedral angles φ (C–N–Cα–C), ψ (N–Cα–C–N) and ω (Cα–C–N–Cα). The φ and ψ angles are relatively free to rotate; the 2D Ramachandran plot shows the allowable ranges

of both angles [Morris92, Lovell03]. The ω angle around the peptide bond is relatively rigid because the peptide bond is a partial double bond but not a single bond. The ω angle can be either close to 0° in *cis* form (both Cα atoms are at the same side of C-N bond) or close to 180° in the more common *trans* form (Cα atoms are at the different side of C-N bond).

2.1.3 RNA

An RNA molecule is a linear polymer of 4 different nucleotides. Like protein, RNA also has four levels of structural organization: primary structure is the linear sequence of nucleotides, secondary structure is the collection of pairs of bases in 3D structure, tertiary structure is the overall shape of an RNA molecule, and quaternary structure is the organization of two or more RNA molecules.

Each nucleotide has three components: phosphate, ribose, and base, as shown in Figure 2.3. The RNA backbone is comprised of alternating phosphate and ribose groups. The ribose is a five carbon sugar that connects the phosphate and the base. Each nucleotide has one of the four bases (A, C, G, and U) and the base extends out of the backbone as a sidechain. There are two alternative ways of defining a nucleotide: an *RNA residue* (the traditional way) goes from phosphate to phosphate, whereas an *RNA suite* goes from sugar to sugar [Murray03].

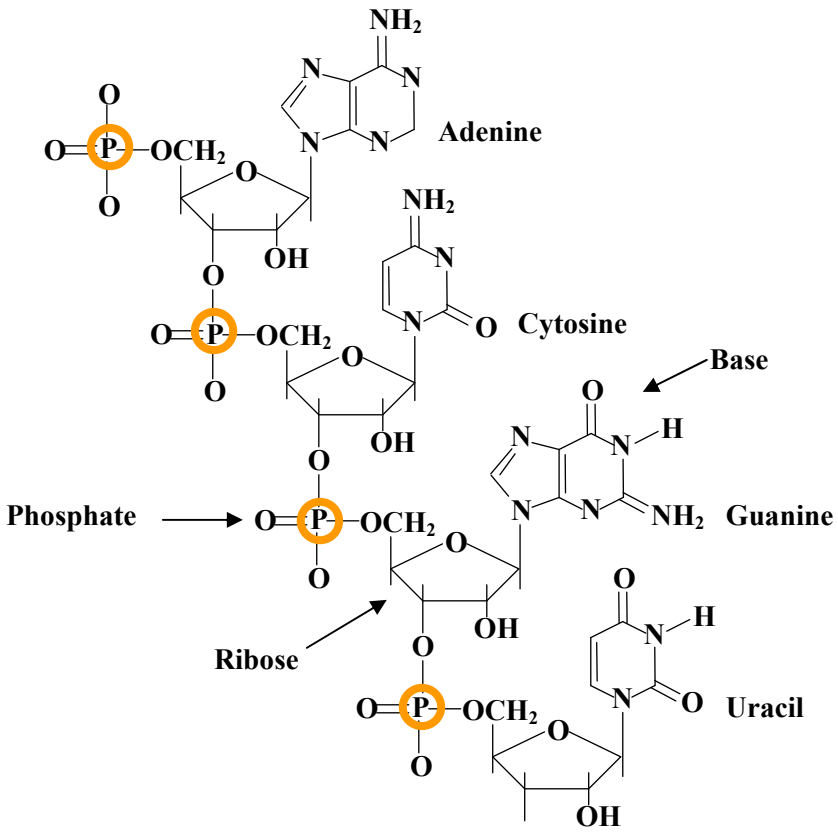


Figure 2.3 A fragment of RNA structure with four nucleotides: A, C, G and U. Each nucleotide has three components: phosphate, sugar and base

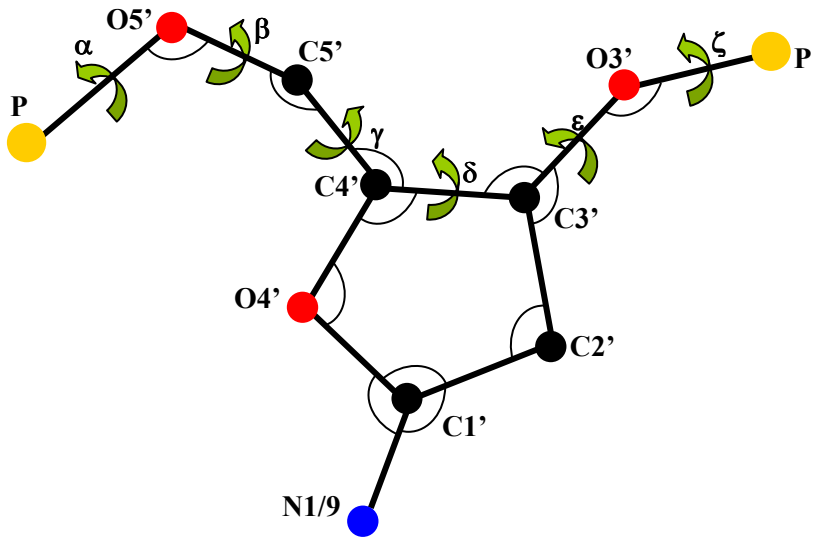
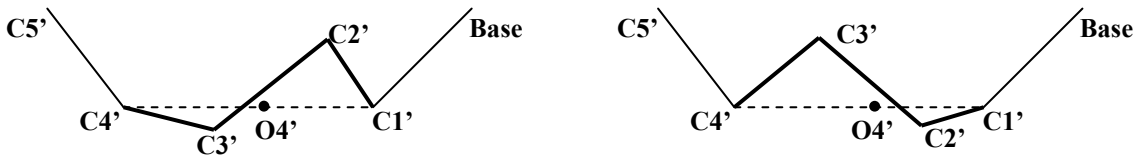


Figure 2.4 Bond lengths, angles and dihedral angles in a nucleotide. Black colored is carbon, red is oxygen, blue is nitrogen, and yellow is phosphorus. Thick solid lines are bond lengths, curves are angles and arrow curves are dihedral angles

In RNA structure, as in protein, bond lengths and angles are relatively rigid but dihedral angles are flexible. Figure 2.4 shows bond lengths, angles and dihedral angles in RNA backbone. Each nucleotide backbone has six dihedral angles, α, β, γ, δ, ε, and ζ, whose atoms and typical ranges are shown in Table 2.1. The δ angle is constrained by the ribose ring structure, but the other dihedral angles are more flexible and most of them show several peaks of allowable ranges [Murray03]. The ribose ring has a C2'-endo or C3'-endo pucker modes, in which either the C2' or C3' atom is extended out of the sugar plane and lies at the same side of C5', as shown in Figure 2.5.

Table 2.1 Typical ranges of 6 nucleotide backbone dihedral angles [Murray03]

Dihedral	Typical ranges
α (O3'-P-O5'-C5')	Peaks at 60°, –60° and 180°. Extra peak at –110° for C3'-endo
β (P-O5'-C5'-C4')	Peaks at 110°, –135° and 180°. Extra peaks at 80° and 135° for C3'-endo
γ (O5'-C5'-C4'-C3')	Peaks at 60° and 180°. Extra peak at –60° for C2'-endo
δ (C5'-C4'-C3'-O3')	Near 84° for C3'-endo and near 147° for C2'-endo
ε (C4'-C3'-O3'-P)	Peak at –150° for C3'-endo and peak at –100° for C2'-endo
ζ (C3'-O3'-P-O5')	Peaks at 60°, –60° and 180°. Extra peak at –140° for C3'-endo

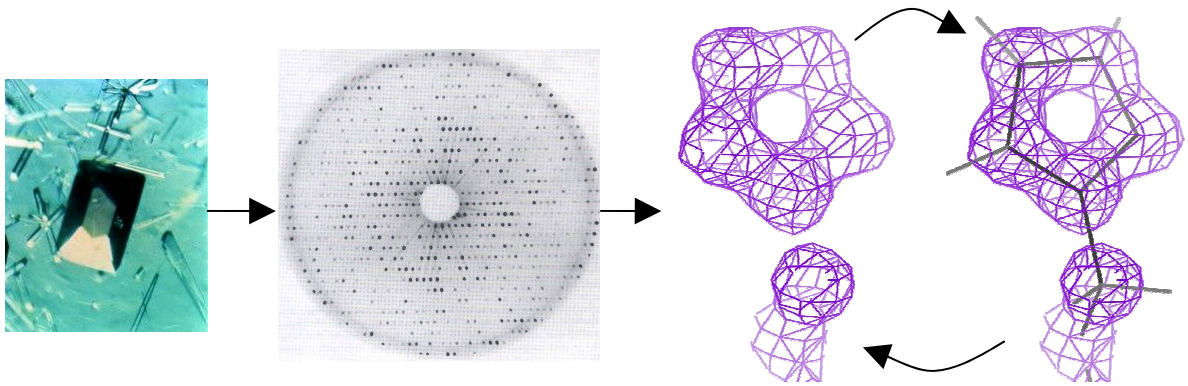


a. C2'-endo **b.** C3'-endo

Figure 2.5 C2'-endo and C3'-endo conformations

2.2 Structure Determination

X-ray crystallography is the most common method to obtain data on RNA/protein structures. X-ray crystallography can be described as a 4-step process [Rhodes06]: in step 1, a pure sample of the desired protein/RNA is coerced to form a crystal by biochemistry methods in the laboratory (Figure 2.6a), in step 2, X-ray beams are scattered by atomic electrons in the crystal to form a series of 2D diffraction patterns (each with a distinct orientation of the crystal), which records the reflections of atoms (Figure 2.6b), in step 3, 3D electron density map is calculated from the diffraction patterns through Fourier transform (FT) (Figure 2.6c), and in step 4, a structure model is fit to the electron density (Figure 2.6d). X-ray crystallography suffers a phase problem: performing Fourier transform needs to know the intensities and phases for all atoms besides the X-ray wavelength, but the diffraction patterns capture only the intensities and not the phases. To solve this problem, at first scientists use various methods to guess initial phases for the atoms and build initial electron density map in step 3, then fit an initial structure model to the electron density in step 4, and repeat steps 3 and 4 for multiple times (called structure refinement) to find a good structure model that fits the electron density map well.



a. Protein/RNA crystal **b.** Diffraction pattern **c.** Electron density map **d.** Atomic model

Figure 2.6 The procedures of X-ray crystallography to determine RNA/protein structure

Progress in protein crystal structure determination has lead to decision algorithms that can largely replace manual rebuilding in an automated refinement pipeline [Adams02]. Although RNA crystallography has also seen revolutionary progress [Ban00, Schluenzen00, Wimberly00, Batey04, Torres-Larios05, Martick06], determining RNA backbone remains a difficult task — RNA backbone has 6 dihedral angles per nucleotide and presents high degrees of freedom, while protein backbone has only 2 dihedral angles per amino acid.

Nuclear magnetic resonance spectroscopy (NMR) explores the quantum mechanical magnetic properties of atoms' nuclei to obtain the structure of a molecule [Keeler05]. When placed in the magnetic field, an atom's nucleus (e.g. ^{1}H, ^{13}C and ^{15}N) resonates at a certain frequency (e.g. proton resonates at 900 MHz). But when the atom is in a molecule, the resonant frequency of the atom's nucleus may change depending on the presence of nearby atoms. NMR spectroscopy performs a sequence of changes of directions and intensities of the magnetic fields (e.g. Nuclear Overhauser Effect Spectroscopy) to detect the resonant frequencies of atoms and then derive the atoms positions. In NMR spectroscopy, usually the molecules are placed in solution and both the 3D structures and the molecular dynamics can be obtained from the experiments. Currently NMR spectroscopy works well on large protein molecules up to 100 kDa (more than 800 amino acids in total) and RNA molecules up to 100 nucleotides and progress continues to be made to resolve the structures of larger molecules [Kolk98, Oberstrass06].

Electron microscopy uses electrons to obtain the images of objects [Frank06]. Like light microscopy, electron microscopy is limited by its wavelength, although it can magnify the image much larger than light microscopy. Electron microscopy obtains the 3D shape (i.e. the surface) of a molecular structure rather than the atomic details. When performing electron

microscopy, the specimens (molecules) are cooled to very low temperature (e.g. liquid nitrogen temperature) and are placed in high vacuum to remove the noises (i.e. radiations). Combining with molecular reconstruction methods, electron microscopy works well on studying the structures, dynamics and interactions of protein and RNA molecules [Frank03].

2.3 All-Atom Contact Analysis for Structure Validation

Various errors may occur when obtaining structures in X-ray crystallography and NMR methods, so structure validation methods are important to verify and correct the obtained structures. For both RNA and protein structures, common structure validation methods include the crystallographic residuals R and R_{free} [Brunger92], difference density (F_{obs}–F_{calc}), and all-atom contact analysis [Word99a, Davis04, Davis07]. The first method focuses on the validation of overall structures, while the last two methods focus on the validation of local structure details. Protein structures have 2D Ramachandran plots [Morris92, Lovell03] and rotamer libraries [Dunbrack97, Lovell00] for verifying local details, but no equivalent tools are available for RNA structures, although significant progress has been made recently [Murray03, Schneider04, Richardson08].

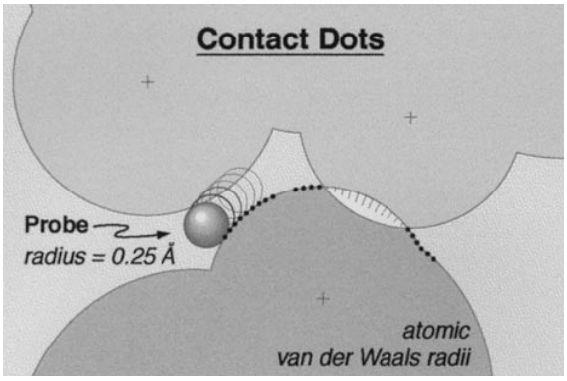


Figure 2.7 A diagram of the small-probe contact dot algorithm. Image courtesy of the Richardson Lab [Word99a]

Here I discuss the all-atom contact analysis method in details, because it is used to validate the RNA structures in Chapter 3. All-atom contact analysis measures and visualizes the goodness-of-fit of interactions of all atoms, especially the hydrogen atoms (for X-ray crystal structures, hydrogen atoms can be added by the program Reduce [Word99b]). It uses a 0.25Å probe sphere to roll over the van der Waals surface of each atom, leaving a contact dot only when the probe touches another not-covalently-bonded atom. The dots are colored by the local gap width between the two atoms: blue when near maximum 0.5Å separation, shading to bright green near perfect van der Waals contact (0Å gap). When suitable H-bond donor and acceptor atoms overlap, the dots are shown in pale green, forming lens or pillow shapes. When incompatible atoms interpenetrate, their overlap is emphasized with spikes instead of dots, and with colors ranging from yellow for negligible overlaps to hot pink for serious clash overlaps >0.4Å. All-atom contact analysis method has been proven to work well on both RNA and protein structures [Word99a, Davis04, Davis07], because it is easy to identify problematic regions. Figures 3.1, 3.8 and 3.9 show examples of all-atom contact analysis on the RNA structure, calculated by Probe on the MolProbity web service [Davis04, Davis07].

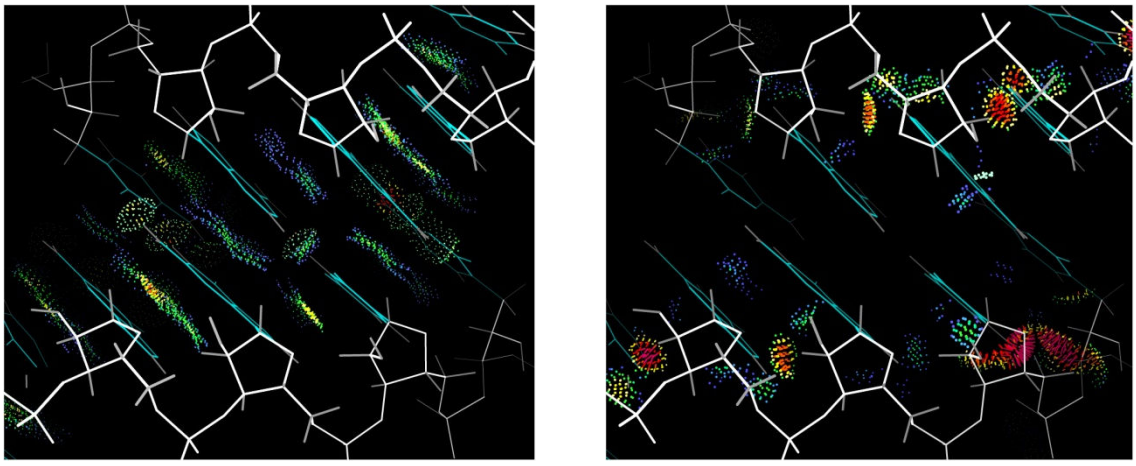
CHAPTER 3

REDUCING STERIC CLASHES IN RNA BACKBONE

3.1 Introduction

Large RNA or RNP (ribonucleoprotein) structures are typically determined at resolutions of 2.5Å or worse by X-ray crystallography; at that level of detail the phosphates and bases can be seen clearly and accurately positioned (see Figure 3.1a), but the remaining backbone atoms and the sugar puckers are underdetermined. All-atom-contact analysis [Word99a, Davis04, Davis07] of deposited RNA structures commonly shows steric clashes between backbone and base atoms or among backbone atoms, as illustrated in Figure 3.1b. Thus, there is a need for new methodology for backbone fitting.

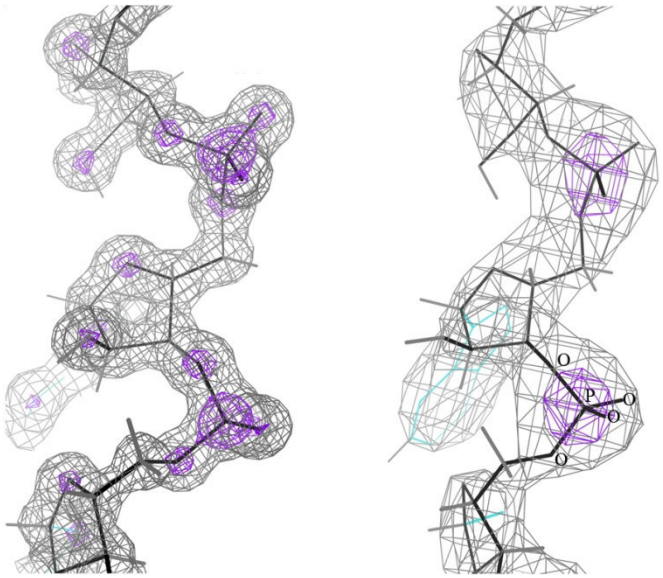
The reason determining RNA backbone conformation is problematic can be appreciated by comparing the full atomic detail seen in an electron density map at 1.04Å resolution (Figure 3.2a) with the same piece of structure in a map at 2.4Å resolution (Figure 3.2b). In the latter, the P (phosphorus) atom of the PO_4 (phosphate) group is still well located by a strong peak (in purple) but the surrounding O atoms cannot be seen individually; the base planes are still clear but sugar pucker cannot be observed directly; and between sugar and phosphate the density necks down evenly with no indication of the zigzag that determines the backbone dihedral angles.



a. all-atom-contact dots within bases.

b. all-atom-contact dots within backbone and between backbone and bases.

Figure 3.1 Selected all-atom-contacts in tr0002/1EVV (yeast phenylalanine tRNA [Jovine00]) at 2.0Å resolution (residues 28-32 and 40-44). The green and blue all-atom-contact dots in 3.1a show almost perfect van der Waals and H-bond contacts between the stacked and paired bases, while the red spikes in 3.1b show large steric clashes that indicate a locally misfit backbone



a. ur0035/1Q9A at 1.04Å resolution [Correll03] **b.** rr0033/1JJ2 at 2.4Å resolution [Klein01]

Figure 3.2 Contoured electron density maps and atomic models for the same piece of ribosomal RNA structure (part of the "sarcin loop") solved at quite different resolutions

Base pairing and stacking are the dominant features determining RNA structure and

energetics. However, the 3D structure of the RNA backbone is at least equally important in functional interactions such as drug binding [Hansen03], protein/RNA interactions [Klein04], aptamer binding [Huang03], and ribozyme catalysis [Doudna02], which often occurs at sites with unusual backbone conformations [Ferre-D'Amare98, Adams04, Golden05] that require careful and accurate analysis. The partner molecules in all these systems interact with the full all-angle, all-atom detail of the RNA, and the structural biology should aim to accurately determine that same level of detail.

The currently-available tools for fitting, refining, rebuilding, and validating crystal structures for proteins are significantly richer and more mature than those for RNA. For proteins, initial model building ("chain tracing") can be done automatically by ARP/wARP [Perrakis99] or Resolve [Terwilliger02], but for RNA, such tools do not yet exist. Almost all large RNA and RNP structures are refined in CNS [Brunger98], which has provided parameter sets and other support for nucleic acids. CNS optimizes agreement of model to data by minimization or simulated annealing protocols, using a simple atomic force field weighted relative to an experimental data term. Energy parameters, weightings, and procedural strategies are not yet fully optimized for RNA: for example, sugar puckers are restrained to the default C3'-endo configuration unless explicitly set by the user, and there are not yet good diagnostics to help make that decision. Model rebuilding between rounds of refinement is traditionally performed by visually comparing the model to the electron density map and manually adjusting it, in software such as O [Jones91], XFit [McRee99], or Coot [Emsley04]. This process is especially time-consuming and error-prone for RNA.

Some model evaluation measures work equally as well for nucleic acids as for proteins, including the crystallographic residuals R and R_{free} [Brunger92], difference density (F_{obs}–

F_{calc}), and all-atom steric clashes [Word99a, Davis04, Davis07]. Other tools that are effective on protein do not yet have equivalently versions for RNA rebuilding, including 2-D Ramachandran plots that compactly assess all available protein backbone dihedral angles [Morris92, Lovell03]. Protein backbones have the advantage of only 2 major degrees of freedom per residue (ϕ and ψ), while RNA backbones have at least 6 degrees of freedom per nucleotide (depending on how sugar pucker is represented), meaning that the equivalent plot for RNA would be 6-D or 7-D. Simplifications using 2-D projections of pairs of adjacent dihedral angle values [Sasisekharan69; Murthy99] have not led to practical tools. Simplification by defining virtual dihedral angles at 2 atoms per residue [Duarte03] is very valuable for locating structural motifs, largely because it is designed to be insensitive to errors. For that same reason, however, it is not useful for building or correcting the all-atom models needed for refining crystallographic or NMR experimental structures. Recent work has identified clusters of preferred RNA backbone conformations [Murray03, Schneider04, Richardson08], but these cannot be represented as a simple 2-D plot and have not yet been incorporated into rebuilding tools. Most steric clashes in refined protein structures are caused by incorrect positions of sidechain atoms, while most steric clashes in refined RNA structures are caused by incorrect positions of backbone atoms. Amino acid sidechains, which have one end fixed in both position and orientation, are easier to adjust than nucleic acid backbone fragments, which have both ends fixed in position, but not orientation.

As progress has been made for proteins that can largely replace manual rebuilding in an automated refinement pipeline [Adams02], I present a program called RNABC (RNA Backbone Correction) to respond the challenge of developing such an automated rebuilding functionality for RNA backbone structures, where the multi-dimensional fitting problem

makes it especially needed. RNABC produces new alternative conformations with equal or better geometry and fewer steric clashes. It first applies the robotics technique of *forward kinematics* [McCarthy90] (a technique determines the conformation of a robot or molecule given its parameters, which is considerably easier than the inverse kinematics problem of determining the parameters given the conformation.) to recalculate rough backbone conformation across a dinucleotide, subject to anchored positions of the best-known features: phosphates and base planes, and then applies conjugate gradient method [Shewchuk94] (a method finds local minimum nearest to the initial values of a function with n variables, in which the gradient of the function is computable) to build the dinucleotide for each of allowable rough backbone conformation. The user can specify most parameters and procedures, or use default values. RNABC finds and clusters all possible conformations within the specified constraints and outputs those with the best geometry and clash scores. The output conformations are scored and sorted based on their fitness to the electron density map. Multi-platform executables and source code of RNABC are available at http://kinemage.biochem.duke.edu/.

In Section 3.2, I describe the details of the RNABC program. In Section 3.3, I show the performance of RNABC program and the results from two extensive tests on sets of existing RNA structures at widely varying resolutions. One tests typical performance, reproducibility, and success at removing clashes in a set of locally similar S-motif structures. The second tests ability to improve the worst local conformations in a set of completely unrelated RNA structures.

3.2 Method

The goal of RNABC program is to remove steric clashes within an individual suite by considering the possible configurations of the dinucleotide that contains the suite, as shown in Figure 3.3.

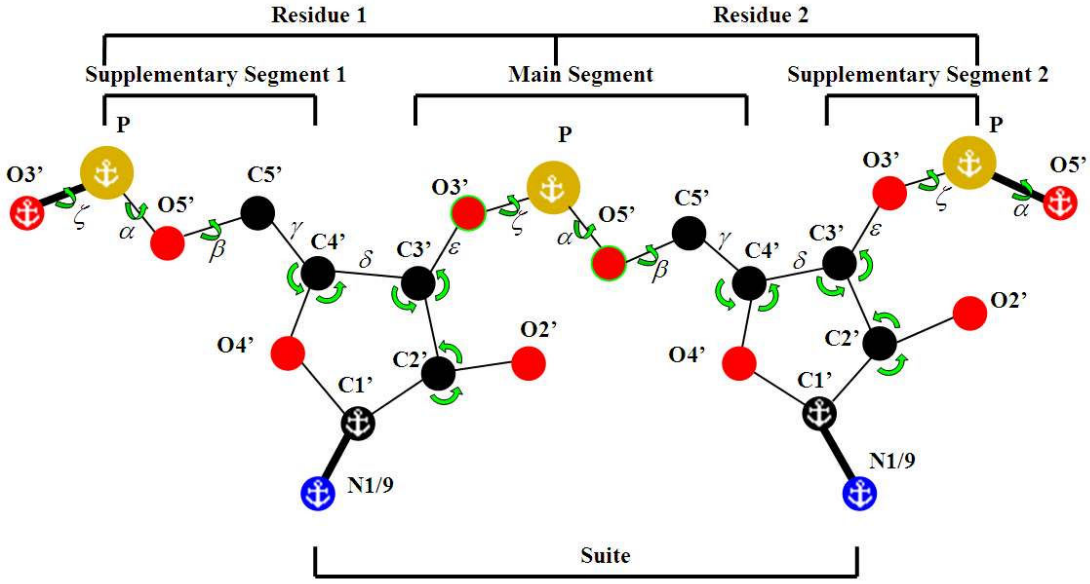


Figure 3.3 Atom labeling and nomenclature for reconstructing a suite within a dinucleotide span. Anchors mark atoms with fixed positions; green arrows mark the conformational degrees of freedom that are explored directly: dihedrals α, β, and ζ, PO_4 orientation around the anchored P, and two of the three bond angles around C2', C3', and C4'. Hydrogens are not shown but are used extensively in RNABC

There are many parameters needed to specify the conformation of a dinucleotide, so I begin the description by making clear which are obtained from the input, which are specified by the user or from standard values, which are constrained, and which are free to be determined by the program. It is important to note that parameters cannot be set arbitrarily because of the constraints that sugars are closed loops, that backbone remains connected, and that certain atom positions (particularly phosphorus and base planes) are usually defined by clear electron density. I conceptually break the bonds of the sugars, so that what remain are three *backbone segments*: *main segment* inside the suite and two *supplementary segments*

outside. RNABC samples the configurations of these segments and considers how they can be joined — it emphasizes early filtering to reduce the number of tested conformations.

3.2.1 Description of the Method

RNABC program reads PDB-format [Berman00] files for the coordinates of the RNA structure. The input file is assumed to include hydrogen atoms, which can be added and optimized conveniently using Reduce [Word99b] via the structure validation service provided by the MolProbity web site [Davis04, Davis07]. MolProbity can also help the user decide which backbone suites need attention by flagging serious clashes between atoms [Word99a] and suspicious sugar puckers. RNABC holds fixed the positions of the bases (defined by the C1'–N1/9 bond) and the phosphorus atoms, since these are the features of RNA structure seen most clearly in X-ray crystallography, and reconstruct the positions of all other backbone atoms in the dinucleotide. RNABC allows only small standard deviations (e.g. 3-4σ) of all the bond lengths and angles to the canonical values used by CNS [Parkinson96]. Alternatively, the user can specify the target bond lengths and angles directly (e.g., from parameter files of a different refinement program), or from the input values, or from the average of the input and canonical values. The user can specify sugar puckers explicitly, keep them from the original coordinates, or let the software determine them by geometric rules based on the perpendicular distance from 3' phosphorus to the C1'–N1/9 vector or to the base plane. The user can even move the position of a phosphorus or base to a specified new location (e.g., to a local peak in the density).

It is common to describe an RNA backbone conformation by the dihedral angles α-ζ illustrated in Figure 3.3. Because RNABC decomposes dinucleotide backbone into segments, it makes a different choice of dihedral and bond angles which is mathematically equivalent

but is easier to filter for disallowed atom positions. RNABC roughly samples dihedral angles α, β, and ζ, and phosphate orientations. It then determines one bond length (C4'–C3') and two bond angles (C5'–C4'–C3', C4'–C3'–O3') to satisfy geometry and generates the sugar puckers by allowable rough backbone atoms and C1' and N1/9 atoms using conjugate gradient method. Note that every atom type (e.g., C4') and every bond length, angle, and dihedral, occurs at least twice within a target dinucleotide. Conditions defined below presume that distances or angles are between nearest atoms of the given type (i.e., within a residue, or within a segment) and hold for all instances, unless otherwise specified.

RNABC uses four types of criteria for evaluating the positions of RNA backbone atoms.

1. NOCLASH: selected atoms should not have steric clashes with the atoms in the suite or the atoms out of the dinucleotide. NOCLASH has two categories:

NOCLASH_M: Atoms O5', C5', C4', C3', O3', OP1, OP2, H5', and H5'' in the *main segment* should have no steric clashes with the atoms in the suite or out of the dinucleotide.

NOCLASH_S: Atoms O4', C2', O2', H1', H2', HO2', H3', and H4' in the two *sugars* should have no steric clashes with the atoms in the suite or out of the dinucleotide.

Atoms within the dinucleotide but out of the suite being adjusted are allowed to clash because local flexibility is not enough to avoid clashes between these and atoms in the suite; clashes related to these atoms may be corrected by running RNABC on adjacent suites.

2. PUCKERTYPE: The two sugar puckers satisfy designated sugar pucker types. Each sextuple {C5', C4', C3', O3', C1', N1/9} generates one sugar pucker through conjugate gradient method. For C3'-endo sugar pucker, the perpendicular distance from C3' to plane C4'–O4'–C1' should be longer than the perpendicular distance from C2' to plane C4'–O4'–C1' by a threshold value (default = 0.2Å), and the perpendicular distance from C2' to plane

C4'–O4'–C1'should be shorter than a threshold value (default = 0.4Å). The δ dihedral is also kept within a range compatible with C3'-endo pucker, but quite permissive (51 to 110°). The C2'-endo sugar pucker has similar criteria.

3. INRANGE: distances of atom pairs, angles of certain atom triples and dihedrals of certain atom quadruples that are not pre-specified should be in certain ranges. INRANGE has two categories:

INRANGE_BB: Backbone atoms O5', C5', C4', C3' and O3' in the main and supplementary segments satisfy: the 2-bond to 4-bond distances of O5'–C1', C4'–C1', C5'–C1', C4'–N1/9, C3'–C1', O3'–C1' and C3'–N1/9 and the multi-bond virtual angles of C5'–C4'–C1', C4'–C1'–N1/9, O3'–C3'–C1' and C3'–C1'–N1/9 should be within certain ranges (e.g. within 3 or 4 standard deviations (σ) of the range implied by combining specified values of the intervening parameters; see section 2.2.3), and multi-bond virtual dihedrals C5'–C4'–C1'–N1/9 and O3'–C3'–C1'–N1/9 should be within certain ranges (see section 2.2.3).

INRANGE_ SB: In the sugars on the backbone, bond length C4'–C3' and bond angles C5'–C4'–C3' and C4'–C3'–O3' in each nucleotide should be within the specified ranges.

4. CGRANGE: sum of squared distances (of atom pairs for bond lengths and of certain atom triples for bond angles) to designated values should be minimized in conjugate gradient method. *CGRANGE* has five categories:

CGRANGE_O4C2: The sugar atoms O4' and C2' satisfy: bond lengths O4'–C4', O4'–C1', C2'–C3', C2'–C1' and bond angles O4'–C4'–C5', O4'–C4'–C3', O4'–C1'–N1/9, C2'–C3'–C4', C2'–C3'–O3', C2'–C1'–N1/9, and O4'–C1'–C2' should be close to the designated values.

CGRANGE_C4C3: The backbone atoms C4' and C3' satisfy: bond lengths C4'–C5',

C4’–O4’, C3’–O3’, C3’–C2’, and C4’–C3’ and bond angles C4’–C5’–O5’, C4’–C3’–O3’, C4’–C3’–C2’, C4’–O4’–C1’, C3’–C4’–C5’, C3’–O3’–P, C3’–C4’–O4’, and C3’–C2’–C1’ should be close to the designated values.

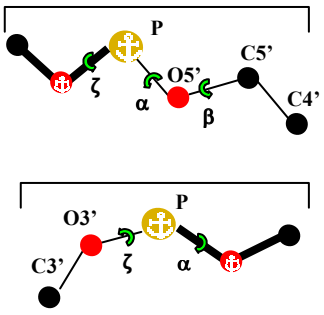
CGRANGE_O5C5: The backbone atoms O5’ and C5’ satisfy: bond lengths O5’–P, C5’–C4’, and O5’–C5’ and bond angles O5’–P–O3’, O5’–C5’–C4’, C5’–O5’–P, C5’–C4’–C3’, and C5’–C4’–O4’ should be close to the designated values.

CGRANGE_O3: The backbone atom O3’ satisfies: bond lengths O3’–C3’ and O3’–P and bond angles O3’–C3’–C4’, O3’–C3’–C2’, and O3’–P–O5’ should be close to the designated values.

CGRANGE_O3O5: The backbone atoms O3’ and O5’ satisfy: bond lengths O3’–C3’, O3’–P, O5’–P, and O5’–C5’ and bond angles O3’–C3’–C4’, O3’–C3’–C2’, O5’–C5’–C4’, and O3’–P–O5’ should be close to the designated values.

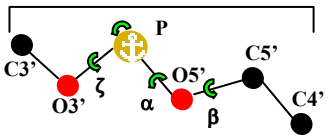
RNABC applies these criteria in first three of four steps: building backbone segments, building sugar geometry, and optimizing dinucleotide geometry.

3.2.1.1 Step 1: building backbone segments



In the first step, RNABC first samples positions of 5 outer atoms in the dinucleotide backbone (O5’, C5’ & C4’ in supplementary segment 1, and O3’ & C3' in supplementary segment 2) by changing dihedral angles, and use forward kinematics to calculate allowable positions of these atoms. Given fixed phosphorus positions and the bond lengths and angles, RNABC first calculates allowable positions of those 5 atoms and evaluate them using criterion INRANGE_BB, which relates them to the anchored atoms C1' and N1/9. To calculate the possible positions

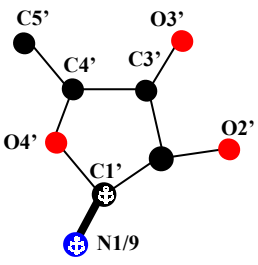
of atom C4’ in supplementary segment 1, for example, with given positions of atoms P, O5’ and C5’, RNABC rotates C4’ around bond O5’–C5’ (i.e. rotate dihedral angle β).



After calculating the allowed positions of atoms in the two supplementary segments, RNABC calculates allowed positions of atoms C3', O3', O5’, C5’, C4’ in the main segment and evaluate them using criteria INRANGE_BB, INRANGE_SB, and NOCLASH_M. The positions of atoms O5’ and O3’ are calculated from the anchored phosphorus by sampling three Euler angles, which represent the rotation of a 3D object by the angles of rotation around three chosen axes. This ensures that O5’ and O3’ are sampled from a sphere centered at P with angle O5’–P–O3’ fixed. The positions of atoms C5’, C4’, and C3’ are calculated from the positions of O5’ and O3’ and the relevant bond and dihedral angles.

In the implementation, I coarsely sample atom positions in steps of 10° (default). Larger rotation angle may not find allowable positions for certain atoms. Smaller rotation angle may generate many similar atom positions (i.e. generate same sugars in the second step) and slow down the program.

3.2.1.2 Step 2: building sugar geometry



In the second step, RNABC constructs the two sugars in the suite by conjugate gradient method from the coordinates of the two sextuples {C5’, C4’, C3’, O3’, C1’, N1/9} around them — these are the atoms in the three bonds that join a sugar to the rest of the structure. The first sextuple has C5’ and C4’ from supplementary segment 1 and C3’ and O3’ from the main segment. The second sextuple has C5’ and C4’ from the main segment and

C3’ and O3’ from supplementary segment 2. The positions of atoms C1’ and N1/9 are anchored. RNABC generates allowable sextuples by evaluating the combinations of main segment and two supplementary segments using criterion INRANGE_SB.

For each sextuple, RNABC first translates and rotates an ideal sugar with canonical bond lengths and angles to superimpose the sextuple, so that the positions of C1’ and the bonds C1’–N1/9 are coincident and the bonds C4’–C3’ parallel to each other. By adding C2’ and O4’ from the ideal sugar, the sextuple is expanded to an octuple {C5’, C4’, O4’, C3’, O3’, C2’, C1’, N1/9}.

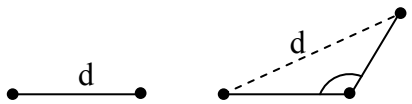
Next, RNABC optimizes two sugars by conjugate gradient method, using the octuple as initial atom positions. All atom positions except C1’ and N1/9 in the octuple are adjusted to make all bond lengths and angles close to the designated values. Conjugate gradient method may stick at an unfavorable local minimum when optimizing the positions of all atoms together, so RNABC divides the sugar atoms into four groups and runs the conjugate gradient method for each group, first optimizes the positions of O4’ and C2’ and adds O2’, H2’ and H1’, then optimizes C4’ and C3’ and adds H3’ and H4’, then optimizes O5’ and C5’ and adds H5’ and H5’’, and finally optimizes O3’, using the criteria CGRANGE_O4C2, CGRANGE_C4C3, CGRANGE_O5C5, and CGRANGE_O3, respectively. All sugar atoms are evaluated by criterion NOCLASH_S during the minimization. The whole optimization process is repeated five times (default) to make sure that all criteria are minimized.

I use non-linear conjugate gradient method that minimizes a continuous function $f(x)$ for which f' exists. The function $f(x)$ is a weighed sum of a series of quartic functions, in which each quartic function represents one bond length or angle constraint (see CGRANGE). I use two categories of quartic functions:

1. $f_i = ((x_1 - a_1)^2 + (x_2 - a_2)^2 + (x_3 - a_3)^2 - d^2)^2$, where $(x_1,\ x_2,\ x_3)$ is an unknown atom position, $(a_1,\ a_2,\ a_3)$ is a known atom position and *d* is the distance.

2. $f_i = ((x_1 - y_1)^2 + (x_2 - y_2)^2 + (x_3 - y_3)^2 - d^2)^2$, where $(x_1,\ x_2,\ x_3)$ and $(y_1,\ y_2,\ y_3)$ are two unknown atom positions and *d* is the distance.

For bond length constraint, *d* is the designated bond length; for bond angle constraint, if the three atoms related with bond angle constructing a triangle by designated values, then *d* is the length of the opposite edge of the angle.



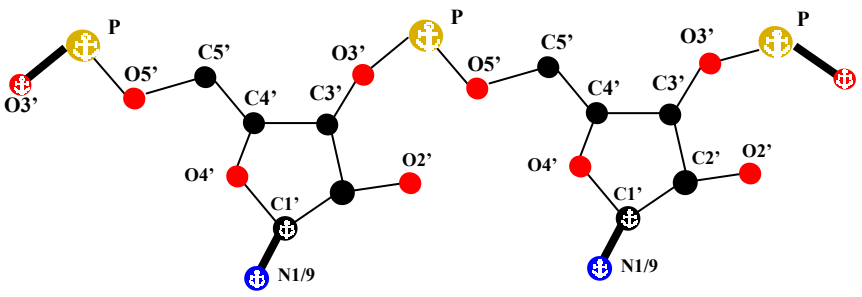
Choosing the above quartic functions has two reasons: first, these quartic functions satisfy the requirement that the bond lengths and bond angles (can be regarded as distances when two bonds are fixed) should be close to designated values, and second, these quartic functions are easy to calculate the derivatives — an essential step in the conjugate gradient method. In the implementation, I set higher weights to the functions for bond lengths (default = 4.0) because the bond length constraints are less flexible than bond angle constraints.

In the last round of minimization, if any of the bond lengths and angles is larger than a threshold scale of standard deviation (default = 3.0) to the designated value, then weight of the corresponding quartic function is increased by 2.0 in default and the conjugate gradient method runs again, in order to keep all the standard deviations small.

3.2.1.3 Step 3: optimizing dinucleotide geometry

In the third step, RNABC adjusts the whole dinucleotide to minimize all the bond lengths and angles to the designated values by conjugate gradient method. RNABC divides the whole dinucleotide into nine groups (some atoms may appear in two groups) and runs conjugate

gradient method for each group, first optimizes O3’ and O5’ and adds OP1 and OP2 in the main segment,



then optimizes O3’ in the main segment, then optimizes C4’ and C3’ and adds H4’ and H3’ in the first sugar, then optimizes O4’ and C2’ and adds O2’, H2’ and H1’ in the first sugar, then optimizes O5’ and C5’ and adds H5’ and H5’’ in the supplemental segment 1, then optimizes O5’ and C5’ and adds H5’ and H5’’ in the main segment, then optimizes C4’ and C3’ and adds H4’ and H3’ in the second sugar, then optimizes O4’ and C2’ and adds O2’, H2’ and H1’ in the second sugar, and finally optimizes O3’ in the supplemental segment 2. All atoms are evaluated by criteria NOCLASH_M and NOCLASH_S during the minimization. The whole process is repeated 10 times (default).

RNABC starts at O3’ and O5’ in the main segment because both sugars have been optimized but the bond length O3’–P and P–O5’ and bond angle O3’–P–O5 remain in designated values and may provide extra flexibility to optimize the whole dinucleotide. Similar to step 2, in the last round of minimization, RNABC increases the weight of certain function if the corresponding bond length or angle is larger than a threshold scale of standard deviation (default = 3.0) to the designated value.

RNABC evaluates and accepts the optimized dinucleotide geometry when all the bond length or angle are less than 5 standard deviations (default) to the designated values and both sugar puckers satisfy criterion PUCKERTYPE.

Finally, RNABC calculates the positions of two HO2’ in both sugars and evaluate them

by criterion NOCLASH_S. RNABC leaves the calculation of HO2' to the last because the position of HO2' is very flexible and it can always find a good position for HO2'.

3.2.1.4 Step 4: clustering and comparing to the electron density map

In the fourth step, RNABC clusters similar suite conformations from the third step, calculates the error scores by comparing the conformations to the electron density map, sorts the conformations by the error scores, and outputs them.

To cluster similar conformations, RNABC calculates RMSD of heavy atoms for each conformation pair and considers the pair as equivalent if the RMSD is less than a threshold value (default = 0.4Å). For equivalent conformations, RNABC keeps the conformation with smaller maximum standard deviation value for all bond lengths and angles to the designated values, because a conformation having a large standard deviation for a certain bond length or angle is more prone to a bad geometry.

RNABC uses a standard procedure in X-ray crystallography for structure refinement to calculate the error score from the dinucleotide conformation to the electron density map [Diamond71, Chapman95]. The target function is $T = \sum_{g_i \in V} [S\rho_o(g_i) + k - \rho_c(g_i)]$, where S and k are scale factors and can be calculated during initialization using partial structural model, g_i is a grid point, V is the volume around the dinucleotide, ρ_o is the observed electron density values and ρ_c is the calculated electron density values. S and k are pre-calculated during each run of RNABC by minimizing a partial model of the RNA structure with the electron density map. In the implementation, I use all the phosphorus in the partial model, because phosphorus is the clearest in the RNA structures.

RNABC calculates ρ_c by Diamond's real space method [Diamond71], which assumes the

electron density of each atom as an isotropic 3D Gaussian and sums up the electron density values for all atoms. The function of Diamond's method is $\rho_c(g_i) = \sum_i Z_i G(a_i, r - r_i)$, where Z_i is the number of electrons associated with atom i, $r - r_i$ is the distance between the position of the atom and grid point, $G(a, r) = a^{-3} e^{-\pi r^2 / a^2}$ is a spherical Gaussian function. $a = \sqrt{B/4\pi}$ when the atomic scattering factor is $f = Ze^{-B\sin^2\theta/\lambda^2}$, where B is the B-factor and θ and λ are known values related with resolution.

For each output conformation, RNABC outputs PDB formatted ATOM items for all atoms, seven dihedral angle values for the suite conformation, standard deviations for all bond lengths and angles to the designated values, and kinemage formatted dinucleotide structure [Richardson01]. Future work is needed to assign each output conformation a backbone conformer name defined by RNA Ontology Consortium [Leontis06, Richardson08].

3.2.2 Implementation

RNABC is implemented in C++. The executables and source code are available at http://kinemage.biochem.duke.edu.

3.2.2.1 User-specifiable Parameters

Each command line invocation of RNABC works on one specified suite. RNABC provides a broad set of parameters, all with defaults but with the option of user specification. Table 3.1 shows some parameters that users can change by flags on the command line. A fuller listing of flags, syntax and choices is given by typing "RNABC -help" in the command line.

Table 3.1 Parameters often specified by RNABC users

Flag	Parameter details
-RESID	Residue ID of central P atom in the suite to be analyzed
-CHAIN	Chain ID character, default = first chain in file
-PUCKER	Pucker type or method for first [second] sugar in suite, default = both determined by 3'P perpendicular to C1'–N1/9 vector
-PARAMETER	Specifies reference bond lengths and angles. Users can choose canonical, original, average of canonical and original, or specify values in a file. Default = canonical
-COARSESPAN	Step size for sampling coarse rotation angles, default = 10°
-WITHINCHAIN	Check collisions with atoms only on the local chain
-OVERLAP	Overlap distance considered a steric clash, default = 0.4Å
-ADJUSTOUTLIER	Maximum allowable standard deviation of all bond lengths and angles, default = 5σ
-STDEV	Standard deviation limitations of all bond lengths and angles that conjugate gradient method attempts to achieve, default = 3σ
-ADJUSTSMALLCLASH	Use geometric method to remove small steric clashes. For hydrogen, adjust the overlap within 0.45Å (default); For others, adjust the overlap within 0.5Å (default)
-ADJUSTCLASHBYCG	Add additional function to remove steric clashes in conjugate gradient method
-CONFORMATION	Maximum number of suite conformations to be output, default = output all conformations

3.2.2.2 Output

For each run on a specified RNA suite, RNABC outputs a single text file containing both coordinates and kinemage graphics from zero (if no trials were successful) to all (default) alternative conformations that satisfy the specified steric clash and covalent geometry conditions. The first half of the file consists of PDB-format coordinates for each output

conformation (with its name and dihedral-angle values), while the second half is readable by the Mage and KiNG kinemage viewers [Richardson01, Davis04, Davis07] for 3D display of the original and new conformations. Mage and KiNG can ignore the first half of the file, and do not need it to have a specific extension (e.g., *.kin).

Mage (C) and KiNG (Java), available at http://kinemage.biochem.duke.edu, are open-source software for multi-platform display and modeling of molecules. Both can display RNABC output, along with electron density maps and MolProbity validation kinemages of the original structure. Mage can build a dockable dinucleotide with adjustable backbone rotamers, if further fitting is desired. KiNG reads more map formats, recontours and moves in them in real time, and can be used on-line in the MolProbity service of the above web site, by reading in the RNABC output file and the user's electron density map (or fetching a map from the Electron Density Server at http://eds.bmc.uu.se/eds/ [Kleywegt04]. When the user has selected a preferred new conformation, the corresponding coordinates can then be cut-and-pasted from the RNABC output file into the PDB file for the overall structure, for submission to further crystallographic refinement.

3.2.2.3 Early rejection

Although forward kinematics generates each segment conformation quickly, sampling many configurations to find segments that satisfy closure constraints can make this method slow. For example, in the first step, in order to calculate the positions of C4' in the main segment, RNABC needs to calculate the positions of O5' and C5' first. The positions of O5' are decided by three Euler angles, and the positions of C5' and C4' are decided by dihedrals α and β. With a coarse sampling of every 10° angle, the total of possible positions for C4' can be $(360/10)^5 > 10^7$.

To improve the performance, RNABC uses criterion INRANGE_BB to reject supplementary segments and main segment that contain disallowed atom positions as soon as they are calculated. For example, for the supplementary segment P–O3'–C3' in residue 2, after calculating a position of O3', RNABC checks the distance from O3' to C1' and if the distance is not within a valid range, it rejects O3' and needs not to calculate C3'.

In the INRANGE_BB, the distances C5'–C1', C4'–N1/9, O3'–C1' and C3'–N1/9 depend on the angles C5'–C4'–C1', C4'–C1'–N1/9, O3'–C3'–C1' and C3'–C1'–N1/9. These angles depend on the pucker state of the sugars and cannot be obtained directly from the other bond lengths and angles. Also I introduce two dihedral angles C5'–C4'–C1'–N1/9 and O3'–C3'–C1'–N1/9, which are used to reject disallowed sugar poses, because the distance and angle criteria allow symmetric sugar poses but the β-D-ribose sugar in RNA has a fixed chirality at the C1' atom. These angles are first measured from ideal C2'-endo and C3'-endo sugars and then extended to certain ranges to accommodate the influence of possibly changed bond lengths, angles and δ dihedral.

Early rejection prevents disallowed positions for most backbone atoms. Table 3.2 shows a typical example, listing the numbers of possible and allowed positions for suite 77-78 of chain9, rr0082/1S72 using the default coarse rotation angles 10°, and with ±5 standard deviations for bond length C4'–C3' and bond angles C5'–C4'–C3' and C4'–C3'–O3', Early rejection can reduce the total calculations by a factor 1.4×10^4.

Table 3.2 Comparison of total and allowed positions of backbone atoms found for suite 77-78 of chain 9, rr0082/1S72

		Sample step (every 10°)		
		Total positions	Allowed positions	Ratio(total/allow)
Supplementary segment 1	O5’	36	7	5
	C5’	1,296	21	62
	C4’	46,656	36	1,296
Supplementary segment 2	O3’	36	7	5
	C3’	1,296	40	32
Main segment†	O5’	46,656	362	129
	C5’	1,679,616	1,284	1,308
	C4’	60,466,176	1,723	35,094
	O3’	46,656	362	129
	C3’	1,679,616	733	2,291
Total		63,968,040	4,575	13,982

†The main-segment O3’ and O5’ are obtained by 3 Euler rotation angles around P, so the total number of positions of O3’ and O5’ could be $(360/10)^3 = 4.7\times10^4$.

3.2.2.4 Adjustment to avoid steric clashes

The conjugate gradient method described in Section 3.2.1.2 and 3.2.1.3 focuses on optimizing the geometry of dinucleotide but not steric clashes. Although in most cases, the optimized conformations are also clash-free, but some conformations may have steric clashes with other atoms and cannot be output. RNABC provides two flags, -ADJUSTSMALLCLASH and -ADJUSTCLASHBYCG, to try to remove steric clashes and preserve the geometry.

The flag -ADJUSTSMALLCLASH removes small steric clashes by moving the atoms to

clash-free positions using geometric method. For hydrogen atoms, the maximum overlap for steric clash is limited to 0.45Å (default); for heavy atoms, the maximum overlap for steric clash is limited to 0.5Å (default). RNABC moves the atom position so that the new overlap is 0.39Å. For heavy atoms, RNABC moves the atom directly, but for hydrogen atoms, RNABC moves the heavy atom bonded with the hydrogen atom to keep the hydrogen geometry. For example, in Figure 3.4, atom C has steric clash with atom D. If C is a heavy atom, RNABC calculates the vector CD and move atom C to C', and if C is a hydrogen atom, RNABC calculates the vector BD and move atom B to B' so atom C moves to C''. Since the adjustment is small, so it is highly possible that the standard deviations of influenced bond lengths and angles are still acceptable.

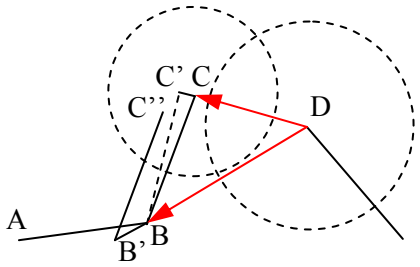


Figure 3.4 Examples of removing small steric clashes by geometry method

The flag -ADJUSTCLASHBYCG removes steric clashes by adding additional quartic function and rerunning the conjugate gradient method. The new quartic function is the same to the quartic function describing bond length or angle constraint (see Section 3.2.1.2) and the weight is assigned as 4.0 (default). Each steric clash is described by one quartic function. The initial value of *d* in the quartic function is set to the atom distance + 0.05Å (default) and if conjugate gradient method fails to remove the steric clash, RNABC increases the value of *d* by 0.02Å (default) and reruns the conjugate gradient method. The maximum runs of conjugate gradient method are limited to 20 times (default).

3.3 Results and Discussion

3.3.1 Running Time Performance

RNABC is tested on a desktop with a 3.0GHz Pentium 4 processor, 1GB memory and Windows XP operating system. I compare the current version with a preliminary version, which has been heavily tested on the performance and correcting steric clashes in RNA dinucleotide. The preliminary version uses exclusively forward kinematics method and runs in three steps. In the first step, it samples backbone conformation in two sub-steps: the first sub-step samples atom positions with steps of 5° and the second sub-step samples atom positions with steps of 1° in a ±2° span. Early rejection is used to speed up the program. In the second step, for each sextuple of backbone conformation, it uses geometric method to construct a sugar and satisfies all bond length and angle constraints. Various acceleration techniques are used to fast reject unfavorable sextuples and speed up the program.

To demonstrate the time that RNABC takes on a typical example, I choose suites 52, 75 and 51 of tr0002/1EVV (see Table 3.3), which exemplify three types of clashes that RNABC can resolve: (a) sugar clashes with base, (b) backbone clashes with base, and (c) sugar/backbone clashes with sugar/backbone. I report running times and number of sextuples for current and preliminary versions of RNABC. I run the preliminary version with three different allowable standard deviations (±3, ±4 and ±5σ) for all bond lengths and angles.

Table 3.3 Comparison of running time for three clash types in tr0002/1EVV for current and preliminary RNABC versions

Clash type		(a) sugar with base		(b) backbone clashes base		(c) sugar/backbone with sugar/backbone	
		# of sextuples	Running time (s)	# of sextuples	Running time (s)	# of sextuples	Running time (s)
Preliminary version	±3σ	20,000	2.7	28,000	2.7	15,000	2.6
	±4σ	184,000	14.2	221,000	11.9	152,000	11.4
	±5σ	1,1119,000	95.8	1,459,000	62.8	1,341,000	75.5
Current version		2,900	3.5	3,300	4.2	2,000	2.8

Table 3.3 shows that the current version is slightly slower than the preliminary version with ±3σ, 3-4 times faster than that with ±4σ, and 15-27 times faster than that with ±5σ. The current RNABC version allows maximum standard deviations of ±5 and minimizes all bond lengths and angles to the designated values (for the above three examples, the maximum standard deviation is 2.1σ), so the current version is more efficient than the preliminary one.

For preliminary version, clash type (a) takes more time than types (b) and (c) when allowable standard deviation increases, because there is a steric clash of 1H2' in the first residue with the second base and it is not clear whether the position of 1H2' is allowed until the positions of O4', C2' and O2' are calculated, so the running time is related with the actual geometry but not with the number of sextuples. But for current version, the conjugate gradient method takes all clash types as the same and optimizes the dinucleotide together, so the running time is more related with the number of sextuples instead of the clash types.

3.3.2 Methods for the Practical Tests

Coordinate files were downloaded either from the NDB (Nucleic acid Data Base [Berman92]) or the PDB (Protein Data Bank [Berman00]). In the text, files are described by both the 6-character NDB code and the 4-character PDB code (e.g., rr0082/1S72); here I list them by NDB code, for brevity, giving only the changing final number for codes with the same starting characters. For the S-motif test, files were: pr0015, 205; rr0009, 16, 20-23, 28-30, 33, 42-45, 47, 49, 52, 54-61, 67, 71, 76-82; ur0002, 7, 26, 33-35. For the test on 154 non-redundant suites, files were: ar0002, 4, 24, 28; dr0008, 10; pr0005, 11, 18, 26, 32, 67, 73, 81, 85, 90; prv001; rr0005, 10, 16, 19, 33; trna12; ur0012, 19.

Hydrogen atoms were added and optimized by Reduce [Word99b]. Residue numbers for S-motifs were obtained from the SCOR database [Klosterman04]. Problem suites were identified in the MolProbity web service [Davis04, Davis07] as having suspect sugar puckers or serious all-atom clashes. Bond length and angle deviations were checked within RNABC. RNABC defines an all-atom steric clash when the distance of two atoms i and j (including hydrogens; i and j > three bonds apart) is less then $vdw_i + vdw_j - 0.4$Å, where vdw_i is the van der Waals radius for atom i from Probe [Word99a]. Bad geometry is defined as a bond length or angle > 4 standard deviations away from canonical value [Parkinson96].

RNABC was run on each problem suite; first with default parameter choices (see Table 3.1). If RNABC failed to find an allowable output conformation at that level, it was rerun with other flags as well, such as -ADJUSTSMALLCLASH, -ADJUSTCLASHBYCG, and -ADJUSTOUTLIER. In the second test, adjacent suites were also run and their results combined, and explicit sugar puckers sometimes specified if needed. If RNABC still produced no output conformations, that example was considered a failure.

The output conformations (see section 3.2.2.2) were visualized in KiNG [Davis04, Davis07], along with a MolProbity multi-criterion kinemage of the starting structure and $2F_{obs} - F_{calc}$ electron density maps from the EDS server [Kleywegt04], if structure factors had been deposited. Conformations were discarded if they were very close to the original or if they were clearly a poorer fit to the electron density. For numerical analysis, Excel spreadsheets were populated with data on initial conformations and their indiscretions, RNABC run parameters, and output conformations, including dihedral values and pucker parameters from Dang [Word00]. For Figures 3.6, 3.7 and 3.8, the selected output coordinates were edited into the PDB file and a new all-atom contact kinemage produced in MolProbity and displayed in KiNG. Such comparison kinemages were used in the second test to judge the level of improvement over the original structure (e.g. quantitative changes in clashes or hydrogen bonding). Any suggested conformations remaining after all these filtering steps were considered reliable options for improving the structure.

3.3.2.1 Removing clashes in many similar S-motif structures

The S-motif (or sarcin-, S-turn-, bulged G-, or loop E-motif) is a distinctive and highly structured internal loop within an A-form RNA double helix, especially common in ribosomal RNAs; an example is shown in Figure 3.5. It includes several non-canonical base pairs and a base triple, and the backbone forms a pronounced S-shape on the primary strand and a small dent and a stack switch on the secondary strand. The S-motif is named for its occurrence in loop E of the 5S ribosomal RNA and especially in the highly conserved sarcin/ricin loop of the large ribosomal subunit, which binds essential translation factors. Toxins like sarcin, ricin, and restrictocin inactivate ribosomes by cleaving the sarcin loop; the S-motif is at the toxin binding site. Classic S-motifs and variants also occur elsewhere in

ribosomal and other RNAs, so there are many similar but not identical examples in the structural database, including a few at very high resolution (e.g., ur0035/1Q9A at 1.04Å resolution [Correll03] shown in Fig. 3.2a).

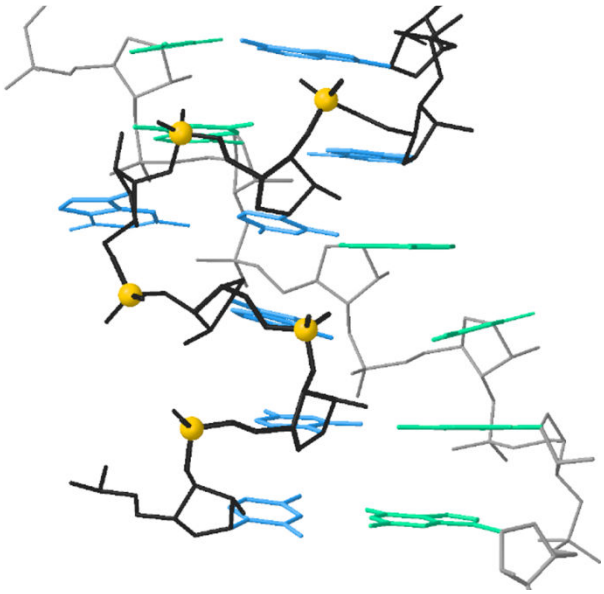


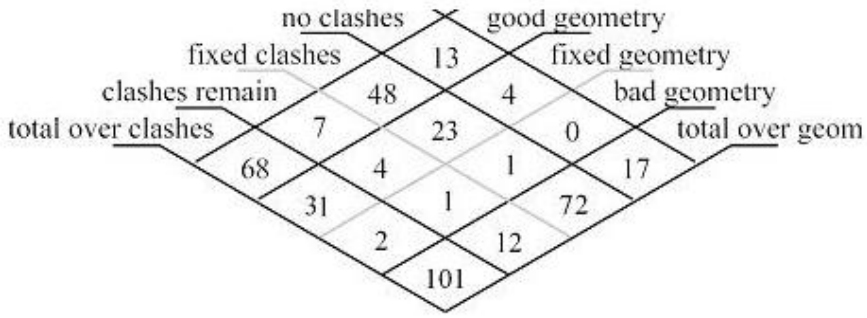
Figure 3.5 S-motif 587-589 in rr0082/1S72; primary strand (front) has black backbone and blue bases. Gold P-atom balls mark the 3-suite, "S"-shaped region studied, but this example was clash-free and thus refit was unnecessary

102 S-motifs in 42 crystal structures are listed by the SCOR database of RNA motifs [Klosterman04]. One S-motif (ur0002/430D a8-a12) has a steric clash between the residue 12 C1', whose position is held fixed by RNABC, and an out-of-suite N6 on residue 20, and was removed from the test set.

The test studied the three distinctive non-A-form suites on the primary strand. The sugar puckers are typically C3'-C2' for the first suite, C2'-C2' for the second, and C2'-C3' for the third. The backbone conformations differ in each suite; they are not easy to fit accurately, so they often show serious steric clashes and sometimes deviant geometry — out of 101 S-motifs, all but 13 contain either steric clashes or bad geometry — making this dataset suitable for testing RNABC.

The S-motif test is done with current RNABC version. For the above 88 S-motifs, RNABC was run on the suites containing either steric clashes or bad geometry, specifying clash-free output with canonical parameters. For example, for the S-motif with primary-strand residues 76-79 in chain 9 of rr0082/1S72 (5S ribosomal RNA) which is shown in Figure 3.6, residues 76 and 77 contain steric clashes so I ran RNABC on suites 76-77 and 77-78, but not on suite 78-79. Table 3.4 summarizes the results. Although adjusting contiguous suites can help in difficult cases, I have confined in this test to running only the suites with clashes.

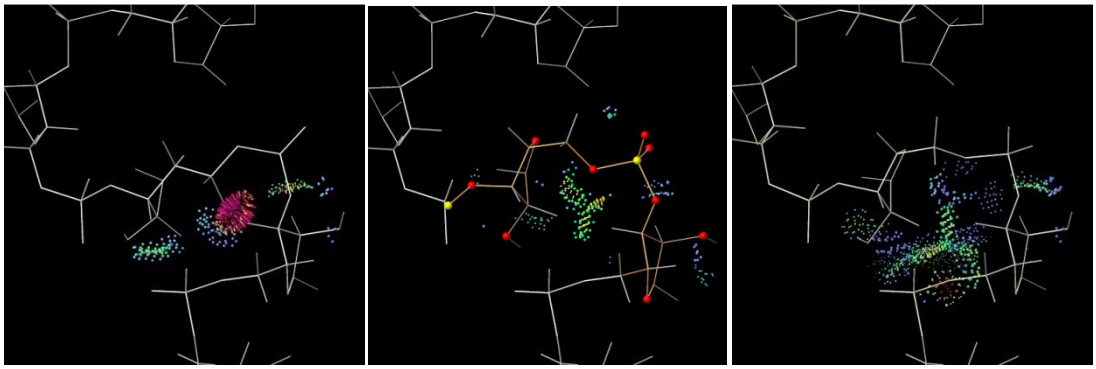
Table 3.4 Performance on removing steric clashes and bad geometry for the 101 S-motifs



For the 101 original S-motifs, 84 have at least one steric clash, and RNABC proposes at least one clash-free conformation for 72 of those (86%). In the 33 S-motifs with bad geometry, RNABC found conformations with good geometry for 31 of them (94%).

Electron density was available for 30 of the 42 structures (71 of the 101 S-motifs). The output conformations were checked for acceptable fit to the electron density where available (e.g. Figure 3.8), and two S-motif outputs were rejected at this stage. Combining both criteria, the overall success rate on this first test was 72 good new proposed conformations out of the 88 S-motifs originally having problems (82%). As an example of what can be accomplished, the RNABC refit shown in Figure 3.6c is very similar to the hand refit in Figure 3.6b, but

took significantly less time and expertise.



a. before reconstruction **b.** refit by hand **c.** refit by RNABC.

Figure 3.6 Suite 76-77 of chain 9, rr0082/1S72 before and after reconstruction

3.3.2.2 Conformations: improving many dissimilar problem suites

Having shown the consistent usefulness of RNABC in correcting a specific backbone motif, a second test was conducted to determine the program's ability to handle severe local problems in a variety of contexts. A set of 25 diverse structures were chosen from the RNA database of Murray, *et al.* [Murray03], with representatives ranging from simple duplex RNA to the ribosomal subunits and tRNAs. For each of these structures, MolProbity and KiNG identify suites with especially bad clashes and sugar-pucker outliers. The test was done with the preliminary version of RNABC and was conducted by Richardson Lab in Duke University. RNABC was run on those suites, as well as suites immediately before and after. If an RNABC run with default parameters failed to yield results, parameters were relaxed in a sequential manner, ensuring that new conformations were found whenever feasible. Table 3.5 gives sample command lines used at each level of trial and the number of suites in test two that first gave output conformations at each level.

Table 3.5 Command lines at successive trial levels for test two

Sample Command	New cases output
RNABC -CHAIN[x] -RESNUM[n] [input.pdb] > [outputfile]	21
RNABC -CHAIN[x] –RESNUM[n] -PARAMETER7 [input.pdb] > [outputfile]	15
RNABC -CHAIN[x] -RESNUM[n] -PARAMETER7 -SIG4 [input.pdb] > [outputfile]	21
RNABC -CHAIN[x] -RESNUM[n] -PARAMETER7 -SIG4 -PUCKER2-3† [input.pdb] > [outputfile]	15

†Note that pucker parameter can be -PUCKER3-3, 3-2, 2-2, or 2-3.

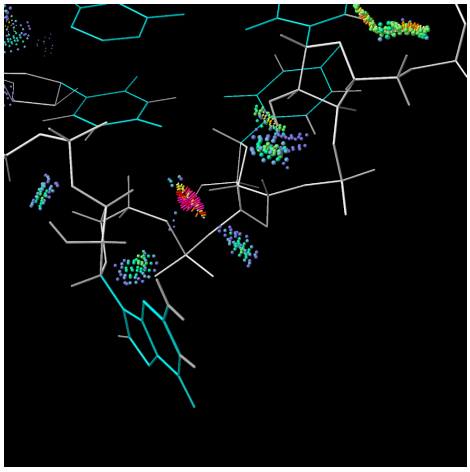
RNABC suggested new conformations for 72 of the 154 suites tested. However, 8 of these new suites were later rejected (see below), 3 due to remaining steric overlaps and/or sugar pucker outliers, 2 because of poor fit to the electron density, and 3 for both of those reasons. Thus, RNABC produced new clash-free conformations and/or better sugar puckers, with satisfactory geometry and density fit, for 64 of the 154 suites tested (42%); 19 of those successes were obtained with default parameters.

Table 3.6 shows the most common problems identified among the original 72 suites, along with how well RNABC improved them. A given suite may have multiple problems, which are categorized into steric clashes (separated by specific pairs of clashing atoms), pucker outliers, and unfavorable ε dihedral values. Pucker and ε dihedral problems often occur together since distortion of ε is often the result of fitting a ribose into the wrong pucker state. RNABC does best at correcting steric clashes, as these were its central design emphasis. It can usually improve and sometimes correct sugar puckers that are misfit as 3’ or 4’ when they should be 2’, as in the example of Figure 3.7. The “other” puckers are extreme distortions, which the program finds difficult to improve or correct. Each of the bad ε values

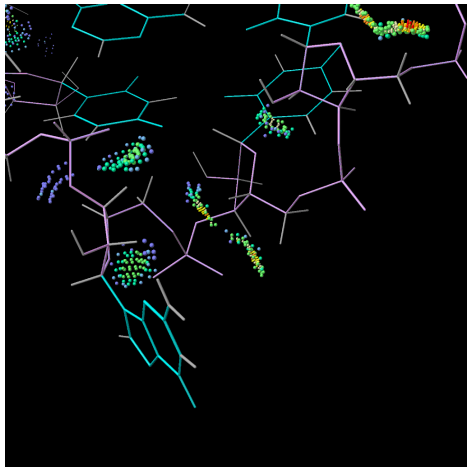
was related to a bad sugar pucker; RNABC corrects 5 of them; the 14 ε values that remain unfavorable correspond to 14 sugar puckers that are improved but are not corrected completely. For all but three suites, when RNABC aggravated a problem in one category, it greatly improved the other two categories.

Table 3.6 Corrections: Instances of three categories of problems in the original structures for 72 suites, and how many were fixed, improved, unchanged, or worsened by RNABC. Configurations are deemed unchanged unless there is a difference of either 5 clash spikes, 10° δ dihedral, 0.5Å perpendicular-line length, or 40° ε dihedral. Note that the total number of clashes is greater than 72 — many suites contained several clashes

Common problems	# of instances	# fixed completely	# improved	# unchanged	# worse	% fixed	% fixed or improved
Steric Clashes							
1H5'–O2'	29	17	6	3	3	59	79
2HO'–P	23	13	5	4	1	57	78
C5' or H5'–C2' or H2'	19	11	4	3	1	58	79
1H2'–O4'	16	10	2	2	2	63	75
Others	80	45	17	7	11	56	78
Pucker outliers							
C4'	12	2	8	2	0	17	83
C3'→ C2'	11	2	7	1	1	18	82
Others	11	1	0	4	6	9	9
Unfavorable ε dihedrals (–45° to +155°)							
Bad ε	19	5	0	14	2	26	26



a. before refit: clash of 1H5' with O2', and C4' puckers for both sugars



b. RNABC refit, in which puckers are improved (C2'-endo) and the clash has been removed

Figure 3.7 pr0032/1FFY suite 33-34 before and after refit by RNABC

The final filter was to determine for the 10 structures (42 of the 72 suites) that had structure factors available, how well RNABC's proposed new conformations fit into the electron density. Although RNABC currently incorporates no constraints for electron density, the fit improved in almost every case — dramatically for some suites, as depicted in Figure 3.8. Five suites were exceptions; three conformations already targeted for elimination by other geometric offenses and two new cases were found that lay significantly outside the density compared to the initial structure. Thus, 8 of the 72 outputs were rejected by these post filtering steps, with 89% of the suggested suite conformations deemed acceptable for future refinement. Overall, this test of RNABC on extreme structural deviations had a 42% success rate, with a fairly low rate of false positives.

The test closes with a look at how many different sets of conformations are output by RNABC, and how different these are from the original structure. In the 235 suites for which RNABC produced output conformations, the output dihedral angles differed from the original by 20°(±3°) RMSD across the 6-dihedral sets, with the extremes ranging from 2°

(tiny wiggles) to 100° (large backbone shifts). Often a single dihedral undergoes a relatively large change while the other dihedrals adjust slightly to accommodate; sometimes two dihedrals change 30°-50° (usually α and γ in the long-recognized "crankshaft" motion). Cases in which 3 or more dihedrals change more than 35° were rare. Moreover, 30% of the time RNABC yields two conformations that are different from each other as well (dihedral RMSD > 20°); a further 5% yield 3 or more different conformations. Thus, RNABC is capable of giving the user significantly new and sometimes varied options with which to replace the original local conformation.

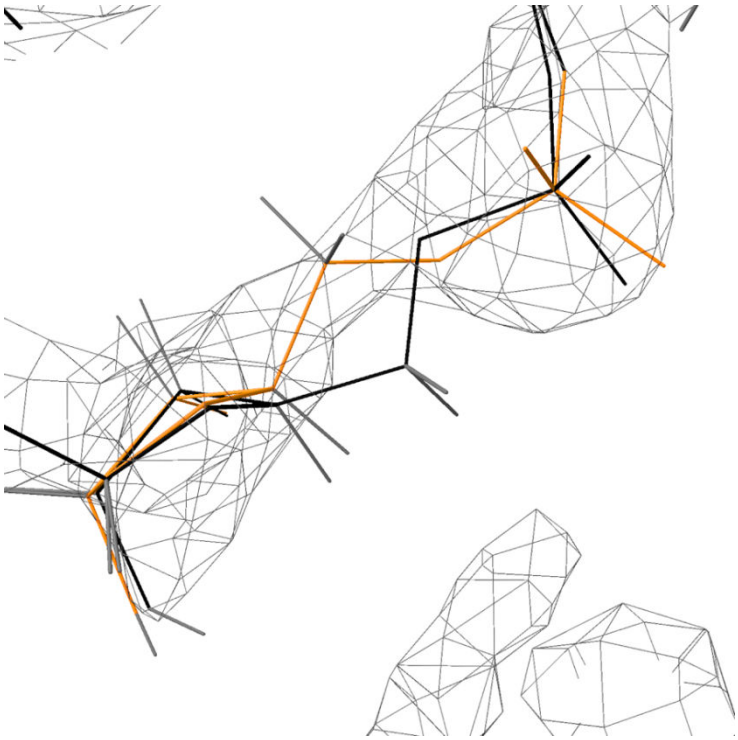


Figure 3.8 rr0082/1S72 suite 1941-1942 refit. The original is in black, and the refit in orange; RNABC's conformation, chosen to avoid bad geometry and clashes, also fits the density better

CHAPTER 4

OPTIMIZING MULTIPLE STRUCTURE ALIGNMENT

4.1 Introduction

Macromolecular structure alignment is an important topic in bioinformatics. RNA and proteins with similar 3D structures may have similar functions and are often evolved from common ancestors [Branden99]. While available sequences of RNA and protein outnumbered available structures by several magnitudes and sequence alignment methods have been widely used to determine protein families and find sequence homology, RNA and protein structure alignment has its importance in disclosing the extend of structure similarity. For example, structure alignment provides "golden standard" for sequence alignment, and conserved regions determined by structure alignment are good candidates for threading and homology modeling.

If RNA and protein structures are considered as rigid bodies, then the problem of structure alignment is to translate and rotate these structures to minimize a score function. Pairwise structure alignment commonly uses root mean squared deviation (RMSD) to measure the structural similarity between corresponding atoms in two structures, once a suitable correspondence has been chosen and the molecules have been translated and rotated to the best match [Horn87]. Pairwise RMSD can be extended to measure the goodness of multiple structure alignment in several ways. Examples from the literature include sum of all

pairwise squared distances [Lupyan05, Sutcliffe87], which I also use, or average RMSD per aligned position [Ochagavia04].

Multiple structure alignment introduces some interesting facts: As a first example, lower *B-factors* (values measure the mobility or uncertainty of given atoms' positions) may suggest that the positions of the atoms should be regarded as more precisely known and should count more toward an alignment or a consensus structure. As a second example, if the correspondence between atoms is derived by multiple sequence alignment, one would like to use conserved atoms in the alignment and omit, or at least reduce the influence of, the exceptions — in a family of structures, an outlier atom should not force the removal of all other atoms that were reliably determined at a certain position. In both examples, I want to be able to assign weights that indicate the confidence levels of atoms' positions. For structure alignment, weighting individual atoms allows a measure of local control in RMSD that is otherwise missing because RMSD is a global measure. Gapped alignment is a special case in which the weight of each atom is assigned either zero or one. In the next section, I show how to use the weights that are assigned to atoms to determine weights of pairs in RMSD and develop an algorithm for multiple structure alignment with weighted atoms.

Many algorithms for multiple structure alignment have been presented. Some first do pairwise structure alignments and then combine structures together. STRUCTAL [Gerstein98] chooses a structure that has minimum total RMSD to all other structures as the consensus structure and aligns other structures to it, MAMMOTH-mult [Lupyan05] chooses one structure at a time and minimizes total RMSD to all previously aligned structures until all structures are aligned, STAMP [Russell92] combines closest pairs and builds a tree for all structure to align them together, and MULTAL [Taylor94] progressively combines the most

similar sequences into a consensus.

Other algorithms align all the structures together instead of combining aligned pairs. Sutcliffe *et al.* [Sutcliffe87], Verboon and Gabriel [Verboon95], and Pennec [Pennec96] iteratively align all the structures to their average structure and achieve minimum RMSD by optimizing rotations for each structure: Algorithm 4.1 is a refinement of theirs, whereas I correctly handles the weights for atoms and optimizes both translations and rotations. CE [Guda01] uses Monte Carlo optimization to achieve a tradeoff between the average atom distance and the aligned columns. MUSTA [Leibowitz01] and MASS [Dror03] use geometric hashing for Cα atom and secondary structures respectively, and combine them into a consensus structure. MultiProt [Shatsky04] and MALECON [Ochagavia04] iteratively use each structure as a consensus, align other structures to it and determine the largest core as the consensus. CBA [Ebert06] and MUSTANG [Konagurthu06] progressively group similar structures, recalculate atom correspondences and optimize the alignment.

4.2 Methods

I define the average structure and weighted RMSD for multiple structures, and then establish properties of the wRMSD.

4.2.1 Weighted Root Mean Square Deviation

Assume there are n structures each having m points (atoms), so that structure S_i for ($1 \le i \le n$) has points $p_{i1}, p_{i2}, \ldots, p_{im}$. For a fixed position k, the n points p_{ik} for ($1 \le i \le n$) are assumed to correspond. I assign a weight $w_{ik} \ge 0$ to point p_{ik} and assign zero weights to gaps, where the coordinates of points in the gaps do not matter. For structure S_i, I define the

weighted centroid as $\sum_{k=1}^{m} w_{ik} p_{ik} \Big/ \sum_{k=1}^{m} w_{ik}$. I assume that there is at least one nonzero weight at each aligned position and define the weight normalized by position as $\hat{w}_{ik} = n w_{ik} \Big/ \sum_{l=1}^{n} w_{lk}$ (Note $\sum_{i=1}^{n} \hat{w}_{ik} = n$). I define the weighted average structure $\overline{S}$ to have points $\overline{p}_k = \sum_{i=1}^{n} w_{ik} p_{ik} \Big/ \sum_{l=1}^{n} w_{lk} = \frac{1}{n} \sum_{l=1}^{n} \hat{w}_{lk} p_{lk}$ for $(1 \le k \le m)$.

Given n structures, I define *weighted RMSD* (wRMSD) as the square root of the weighted average of all squared pairwise distances. Note there are $n(n-1)/2$ structure pairs and each structure pair has m distances. Thus, if $w_{ijk} = \hat{w}_{ik} w_{jk} = w_{ik} \hat{w}_{jk}$ is the weight for point pair (p_{ik}, p_{jk}), then I define

$$\text{wRMSD} = \sqrt{\frac{2}{mn(n-1)} \sum_{i=2}^{n} \sum_{j=1}^{i-1} \sum_{k=1}^{m} w_{ijk} \left\| p_{ik} - p_{jk} \right\|^2}$$

There are many ways to define a combined weight w_{ijk}; I choose to multiply the weights w_{ik} and w_{jk} to capture the confidence in aligning atoms p_{ik} or p_{jk} from structure i and structure j at position k. If either w_{ik} or w_{jk} is zero, then the combination w_{ijk} is zero; if both atoms at a position have equal confidence, then they both factor equally into the combination. This choice is compatible with unweighted RMSD, and captures gapped alignment as a special case. As can be seen in the mathematics, with this choice I can align structures to an average structure and speed up computation. Alternate ways to define w_{ijk} may not work: For example, if I define $w_{ijk} = (w_{ik} + w_{jk}) / 2$, then when one of w_{ik} or w_{jk} is zero and the other is nonzero, the wRMSD value will be influenced by an atom position in which I have no confidence.

Since m and n are fixed, I can equivalently minimize the weighted sum of all squared

pairwise distances $\sum_{i=2}^{n}\sum_{j=1}^{i-1}\sum_{k=1}^{m} w_{ijk}\left\|p_{ik}-p_{jk}\right\|^2$ instead of the wRMSD. The following lemma on weighted sums of squares allows us to make several observations about the average structure $\overline{S}$ under the wRMSD.

Lemma 4.1 For any aligned position k, the sum of weighted squared distances from p_{1k}, p_{2k}, …, p_{nk} to any point q_k equals the sum to the average point $\overline{p}_k$ plus the sum from $\overline{p}_k$ to q_k: $\sum_{i=1}^{n} w_{ik}\left\|p_{ik}-q_k\right\|^2 = \sum_{i=1}^{n} w_{ik}\left\|p_{ik}-\overline{p}_k\right\|^2 + \sum_{i=1}^{n} w_{ik}\left\|q_k-\overline{p}_k\right\|^2$

Proof: To establish the Lemma, I subtract the second term from both sides, expand the difference of squares, and apply the definition of $\overline{p}_k$ in the penultimate step.

$$\sum_{i=1}^{n} w_{ik}\left(\left\|p_{ik}-q_k\right\|^2-\left\|p_{ik}-\overline{p}_k\right\|^2\right)=\sum_{i=1}^{n} w_{ik}\left(p_{ik}-q_k+p_{ik}-\overline{p}_k\right)\left(p_{ik}-q_k-p_{ik}+\overline{p}_k\right)$$

$$=\left(\overline{p}_k-q_k\right)\sum_{i=1}^{n} w_{ik}\left(2p_{ik}-\overline{p}_k-q_k\right)=\left(\overline{p}_k-q_k\right)\sum_{i=1}^{n} w_{ik}\left(\overline{p}_k-q_k\right)=\sum_{i=1}^{n} w_{ik}\left\|q_k-\overline{p}_k\right\|^2 \quad \square$$

I list three theorems relating the weighted sum of all squared pairwise distances to the average structure. Theorem 4.1 says that if the wRMSD is used to compare multiple structures, then what really happens is that all structures are being compared to the average structure — that the average structure $\overline{S}$ is a consensus. By comparing to the average structure, I reduce the number of pairs of structures that must be compared from $n(n-1)/2$ to n.

Theorem 4.1 The weighted sum of squared distances for all pairs equals the weighted sum of squared distances from all structures to the average structure $\overline{S}$:

$$\sum_{i=2}^{n}\sum_{j=1}^{i-1}\sum_{k=1}^{m} w_{ijk}\left\|p_{ik}-p_{jk}\right\|^2 = n\sum_{i=1}^{n}\sum_{k=1}^{m} w_{ik}\left\|p_{ik}-\overline{p}_k\right\|^2$$

Proof: In Lemma 4.1, I replace q_k by p_{jk}, multiply by the weight $\hat{w}_{jk}$, and sum over all j and k to obtain:

$$\sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{k=1}^{m} w_{ijk}\left\|p_{ik}-p_{jk}\right\|^2 = 2\sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{k=1}^{m} w_{ijk}\left\|p_{ik}-\overline{p}_k\right\|^2$$

Re-arrange the order of summation on the left and notice that the terms with $i = j$ are canceled and every other term appears twice:

$$\sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{k=1}^{m} w_{ijk}\left\|p_{ik}-p_{jk}\right\|^2 = 2\sum_{i=2}^{n}\sum_{j=1}^{i-1}\sum_{k=1}^{m} w_{ijk}\left\|p_{ik}-p_{jk}\right\|^2$$

The resulting equation gives the desired result after dividing out the extra factor of two:

$$2\sum_{i=2}^{n}\sum_{j=1}^{i-1}\sum_{k=1}^{m} w_{ijk}\left\|p_{ik}-p_{jk}\right\|^2 = 2\sum_{i=1}^{n}\sum_{k=1}^{m}\left(w_{ik}\left\|p_{ik}-\overline{p}_k\right\|^2\sum_{j=1}^{n}\hat{w}_{jk}\right) = 2n\sum_{i=1}^{n}\sum_{k=1}^{m}\left(w_{ik}\left\|p_{ik}-\overline{p}_k\right\|^2\right) \quad \square$$

Theorems 4.2 and 4.3 suggest how to choose the structure closest to a given set of structures. If you can choose any structure, then chose the average $\overline{S}$; if you must choose from a limited set, then choose the structure closest to the average $\overline{S}$.

Theorem 4.2 The average structure $\overline{S}$ minimizes the weighted sum of squared distances from all structures, *i.e.* for any structure Q with points q_1, q_2, …, q_m, $\sum_{i=1}^{n}\sum_{k=1}^{m} w_{ik}\left\|p_{ik}-q_k\right\|^2 \geq \sum_{i=1}^{n}\sum_{k=1}^{m} w_{ik}\left\|p_{ik}-\overline{p}_k\right\|^2$ and equality holds if and only if $q_k = \overline{p}_k$ or $w_{ik} = 0$ for all points.

Proof: This follows immediately from Lemma 1 since $\sum_{i=1}^{n} w_{ik}\left\|q_k-\overline{p}_k\right\|^2 \geq 0$ with equality if and only if $q_k = \overline{p}_k$ or $w_{ik} = 0$ for all points. □

Theorem 4.3 The structure from Q_1, Q_2, …, Q_l with minimum wRMSD to $\overline{S}$ minimizes the weighted sum of squared distances to all structures S_i for ($1 \leq i \leq n$).

Proof: In Lemma 4.1, $\sum_{i=1}^{n}\sum_{k=1}^{m} w_{ik}\left\|p_{ik}-\overline{p}_k\right\|^2$ is fixed by the set of structures, so it is both necessary and sufficient to minimize $\sum_{i=1}^{n}\sum_{k=1}^{m} w_{ik}\left\|q_k-\overline{p}_k\right\|^2$. □

4.2.2 Rotation and Translation to Minimize wRMSD

In structure alignment, structures are translated and rotated in 3D space to minimize the wRMSD. I define R_i as a 3×3 rotation matrix and T_i as a 3×1 translation vector for structure S_i. I aim to find optimal T_i and R_i for each structure to minimize wRMSD. The target function is $\arg\min_{R,T}\left(\sum_{i=2}^{n}\sum_{j=1}^{i-1}\sum_{k=1}^{m} w_{ijk}\left\|R_i p_{ik}-T_i-R_j p_{jk}+T_j\right\|^2\right)$.

Let $p_{ik}' = R_i p_{ik} - T_i$ and apply Theorem 1 to the target function, so

$$\sum_{i=2}^{n}\sum_{j=1}^{i-1}\sum_{k=1}^{m} w_{ijk}\left\|R_i p_{ik}-T_i-R_j p_{jk}+T_j\right\|^2 = \sum_{i=2}^{n}\sum_{j=1}^{i-1}\sum_{k=1}^{m} w_{ijk}\left\|p_{ik}'-p_{jk}'\right\|^2$$

$$= n\sum_{i=1}^{n}\sum_{k=1}^{m} w_{ik}\left\|R_i p_{ik}-T_i-\overline{Rp}_k+\overline{T}_k\right\|^2$$

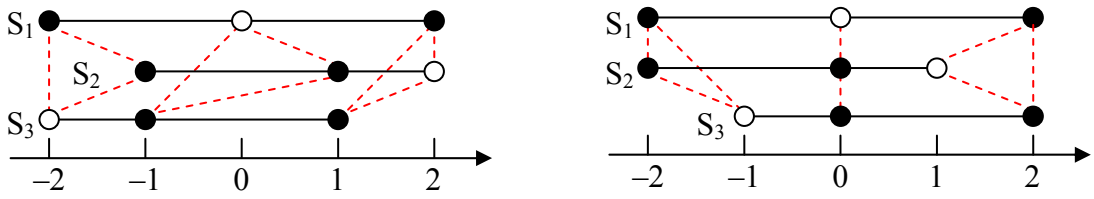
where $\overline{Rp}_k = \sum_{l=1}^{n} R_l \hat{w}_{lk} p_{lk} \Big/ \sum_{l=1}^{n} \hat{w}_{lk} p_{lk}$ and $\overline{T}_k = \frac{1}{n}\sum_{l=1}^{n} \hat{w}_{lk} T_l$.

In this way, I change the minimization of the wRMSD for all pairs to the minimization of the wRMSD from all structures to the average structure.

4.2.2.1 Optimum translation and rotation

Horn [Horn87] shows that to align a pair of structures to minimize the wRMSD, one can first translate both structures so their centroids coincident (say, at the origin), then solve for the optimum rotation. For weighted multiple structure alignment, however, this is no longer

true. Consider the example of Figure 4.1 with 3 structures S_1, S_2, and S_3, each containing three weighted atoms in correspondence from left to right. Black dots denote weights equal to 1 and white dots denote the weights equal to 0, i.e. the gaps. The alignment in Figure 4.1a moves the weighted centroids to the origin, and obtains wRMSD $\sqrt{6}$; moving unweighted centroids to the origin would give wRMSD $\sqrt{2/3}$. The alignment in Figure 4.1b achieves the optimum wRMSD 0 by translating S_2 by –1 and S_3 by 1 from Figure 4.1a. The difference arises because the centroid is defined for each structure independently, but the contribution of each structure to the alignment score depends also on the weights assigned to the structures that are being compared to.



a. Alignment by moving centroids to the origin **b.** Alignment achieves optimum RMSD

Figure 4.1 Example of aligning three structures with gaps. Dashed lines denote the correspondence of points, black dots denote weights equal to 1, and white dots denote weights equal to 0, i.e. the gaps

Verboon and Gabriel [Verboon95] and Pennec [Pennec96] present iterative algorithms to minimize RMSD for multiple structure alignment by translating the centroids of all structures to the origin and optimizing rotations, but the example in Figure 4.1 shows that their algorithms may not find optimum RMSD in weighted structure alignment. It turns out that the optimum translations cannot be found easily. Theorem 4.4 (see Appendix I for proof) shows the relation of the optimum translations and rotations. In general, the translations and rotations cannot be separated for minimizing wRMSD in multiple structure alignment.

Theorem 4.4 The optimum translation T_i and the optimum rotation R_i for structure S_i ($1 \leq$

$i \le n$) satisfy the following n linear equations, of which n–1 are independent:

$$\sum_{k=1}^{m} w_{ik}\left(R_i p_{ik} - T_i\right) = \frac{1}{n}\sum_{k=1}^{m} w_{ik}\left(\sum_{l=1}^{n} \hat{w}_{lk}\left(R_l p_{lk} - T_l\right)\right)$$

Given all optimal rotations R_i for ($1 \le i \le n$) and one translation T_j ($1 \le j \le n$), the remaining n–1 optimal translations T_i for ($1 \le i \le n$, $i \ne j$) can be obtained by

$$T_i = T_j - \left(\sum_{l=1}^{n} R_l\left(\frac{1}{n}\sum_{k=1}^{m} p_{lk}\left(w_{ilk} - w_{jlk}\right)\right) - R_i\sum_{k=1}^{m} w_{ik} p_{ik} + R_j\sum_{k=1}^{m} w_{jk} p_{jk}\right) \Bigg/ \sum_{k=1}^{m} w_{ik} \ .$$

Note that if I use weighted RMSD at aligned positions, i.e. $w_{ik} = w_{jk} = w_k$ for ($1 \le i, j \le n$, $1 \le k \le m$), then the translations and rotations can be separated and the optimal translations can be obtained by translating the weighted centroids of all structures to the origin. It is because I have $\hat{w}_{ik} = 1$ and $w_{ijk} = w_k$, and the optimal translation T_i in Theorem 4.4 becomes:

$$T_i = T_j - \left(R_j\sum_{k=1}^{m} w_k p_{jk} - R_i\sum_{k=1}^{m} w_k p_{ik}\right) \Bigg/ \sum_{k=1}^{m} w_k$$

If I translate the centroid C_i of structure S_i for ($1 \le i \le n$) to the origin before optimizing the rotation, i.e. $p'_{ik} = p_{ik} - C_i$, then I have $\sum_{k=1}^{m} w_k p_{ik} \Big/ \sum_{k=1}^{m} w_k = 0$. The above equation becomes $T_i = T_j$ and I can simply choose $T_j = 0$, so the optimal translation is achieved by translating the centroid of each structure to the origin.

4.2.2.1 Algorithm for minimizing wRMSD

Finding optimal translations and rotations for multiple structures is harder than for a pair because the minimization problem no longer reduces to a linear equation. Instead of directly finding the optimal translations and rotations, I use the fact that the average is the best consensus from Theorem 4.1, and present an iterative algorithm to converge to the minimum

wRMSD. I align each structure to the average structure separately in each iteration. Because translating and rotating structures also change the average structure, I repeatedly calculate the average structure and align each structure to it until the algorithm converges to a local minimum of wRMSD.

Algorithm 4.1. Given n structures with m points (atoms) each and weights $w_{ik} \geq 0$ for each point, minimize wRMSD to within a chosen ε, e.g. $\varepsilon = 1.0 \times 10^{-5}$.

1. Calculate the average structure $\overline{S}$ with points $\overline{p}_k = \frac{1}{n}\sum_{i=1}^{n} \hat{w}_{ik} p_{ik}$, and the weighted sum of squared distances to $\overline{S}$: $SD = \sum_{i=1}^{n}\sum_{k=1}^{m} w_{ik} \left\| p_{ik} - \overline{p}_k \right\|^2$.

2. For each i, translate S_i and $\overline{S}$ so their centroids, using weighs for S_i, are at the origin. (Let $\overline{C}_i = \sum_{k=1}^{m} w_{ik}\overline{p}_k \Big/ \sum_{k=1}^{m} w_{ik}$ denote the centroid of $\overline{S}$ using the weights of S_i). Use Horn's method [Horn87] to optimally rotate each S_i into alignment with $\overline{S}$. Translate S_i and $\overline{S}$ by $-\overline{C}_i$.

3. Calculate new average $\overline{S}^{\text{new}}$ and $SD^{\text{new}} = \sum_{i=1}^{n}\sum_{k=1}^{m} w_{ik} \left\| p_{ik}{}^{\text{new}} - \overline{p}_k{}^{\text{new}} \right\|^2$.

4. If $(SD - SD^{\text{new}}) / SD < \varepsilon$, then the algorithm terminates;
   otherwise, set $SD = SD^{\text{new}}$ and $\overline{S} = \overline{S}^{\text{new}}$ and go to step 2.

The translation of $\overline{S}$ by $-\overline{C}_i$ in step 2 keeps $\overline{S}$ untouched (no rotation involved with $\overline{S}$). The translation of S_i by $-\overline{C}_i$ keeps the weighted sum of squared distances for S_i and $\overline{S}$ the same, after optimized by Horn's method.

Horn's method and the above theorems imply that the deviation SD decreases monotonically in each iteration. From Theorem 4.1, I know that minimizing the deviation SD

to the average minimizes the global wRMSD. From Horn [Horn87], in step 2 I have

$$\sum_{i=1}^{n}\sum_{k=1}^{m} w_{ik} \left\| p_{ik}{}^{\text{new}} - \overline{p}_k \right\|^2 \le \sum_{i=1}^{n}\sum_{k=1}^{m} w_{ik} \left\| p_{ik} - \overline{p}_k \right\|^2 = SD \,.$$

From Theorem 4.2, in step 3 I have

$$SD^{\text{new}} = \sum_{i=1}^{n}\sum_{k=1}^{m} w_{ik} \left\| p_{ik}{}^{\text{new}} - \overline{p}_k{}^{\text{new}} \right\|^2 \le \sum_{i=1}^{n}\sum_{k=1}^{m} w_{ik} \left\| p_{ik}{}^{\text{new}} - \overline{p}_k \right\|^2 .$$

So $SD^{\text{new}} \le SD$ and SD decreases in each iteration. The algorithm stops when the decrease is less than a threshold ε and achieves a local minimum of SD.

Horn's method calculates the optimal rotation matrix for two m-atom structures in O(m) operations and the translations in step 2 and 3 take O($n\ m$) in total, so initialization and each iteration take O($n\ m$) operations.

If I use weighted RMSD at aligned positions or unweighted RMSD, I can simplify Algorithm 4.1 by translating the centroids of all structures to the origin before step 2, and remove all the translation operations in step 2. It is easy to verify that the convergence and the time complexity of the algorithm are the same.

4.3 Results and Discussion

4.3.1 Performance

I test Algorithm 4.1 by minimizing wRMSD for 23 protein families from HOMSTRAD database [Mizuguchi98] with more than 10 structures and total aligned length longer than 100 (each aligned position contains more than two Cα atoms). I assign weights 1 to aligned Cα atoms and weights 0 to gaps. I run the algorithm 10,000 times for each protein family. Each time I randomly translate (within 100Å) and rotate each structure in 3D space, then

minimize wRMSD. The results are shown in Table 4.1.

Table 4.1 Performance of the algorithm on different protein families from HOMSTRAD. I report *n*, the number of proteins, *m*, the number of atoms aligned, the wRMSD from HOMSTRAD Alignment (HA), the wRMSD of the optimal alignment from Algorithm 4.1, statistics on iterations and time (milliseconds) for 10,000 runs of each alignment

Protein family	*n*	*m*	#gaps	wRMSD HA(Å)	optim. wRMSD	% rel. diff	Iterations avg,med,max	Time (ms) avg,median,max
α-amylase	23	616	415	6.14	6.01	2.11	15.5, 16, 19	382, 391, 471
αamylase_NC	23	741	517	6.24	6.09	2.40	14.5, 15, 20	407, 411, 551
asp	13	346	49	2.20	2.15	2.46	9.4, 9, 12	100, 100, 130
cys	13	242	52	1.74	1.71	1.83	13.2, 13, 16	110, 110, 140
fabp	17	137	15	1.89	1.89	0.26	6.9, 7, 8	44, 40, 60
ghf22	12	129	10	1.42	1.40	1.50	6.0, 6, 7	29, 30, 40
glob	41	168	59	2.07	2.01	2.57	10.5, 11, 12	148, 150, 170
gluts	14	230	30	2.84	2.76	2.77	8.0, 8, 10	62, 60, 80
grs	11	498	236	4.18	3.64	14.69	8.4, 8, 9	110, 110, 140
igvar-h	21	134	27	2.25	2.14	5.42	8.3, 8, 10	60, 60, 70
kinase	15	421	216	7.69	7.39	4.04	16.0, 16, 21	212, 210, 280
ldh	14	352	86	2.64	2.60	1.41	11.6, 12, 14	127, 130, 160
lipocalin	15	190	72	3.97	3.88	2.34	11.6, 12, 14	87, 90, 110
ltn	12	246	44	1.51	1.49	1.39	7.6, 8, 9	60, 60, 80
p450	12	481	186	4.08	4.04	1.18	10.0, 10, 13	132, 130, 160
phc	12	177	29	3.20	2.93	9.24	9.1, 9, 12	55, 50, 80
phoslip	18	130	19	1.51	1.49	1.53	10.5, 11, 12	66, 70, 150
proteasome	17	283	135	6.86	6.10	12.50	13.2, 13, 16	137, 140, 160
sdr	13	297	120	4.03	3.73	8.00	9.5, 10, 12	89, 90, 110
sermam	27	275	94	2.10	2.06	2.18	9.4, 9, 12	134, 130, 170
subt	11	309	87	2.82	2.78	1.53	15.6, 16, 18	148, 150, 180
tim	10	254	12	1.47	1.46	0.87	7.3, 7, 9	51, 50, 70
uce	13	162	48	2.57	2.50	3.14	9.6, 10, 11	57, 60, 70

For all minimized RMSD values in each protein family's 10,000 tests, the difference between maximum and minimum RMSD is less than 1.0×10^{-5}, so they converge to the same local minimum, which is likely the global minimum.

Figure 4.2 shows that for all 23 families, each iteration decreases RMSD rapidly, in 5-6 iterations, whereas the maximum number of iterations for $\varepsilon = 1.0\times10^{-5}$ is 21. The experiment was run on 1.8 GHz Pentium M laptop with 768M memory. The code is written in MATLAB and is downloadable at http://www.cs.unc.edu/~xwang. Figure 4.3 indicates that the observed average running time is linear in the number of atoms in the structures, so Algorithm 4.1 approaches the lower bound of multiple structure alignment $\Theta(n\,m)$.

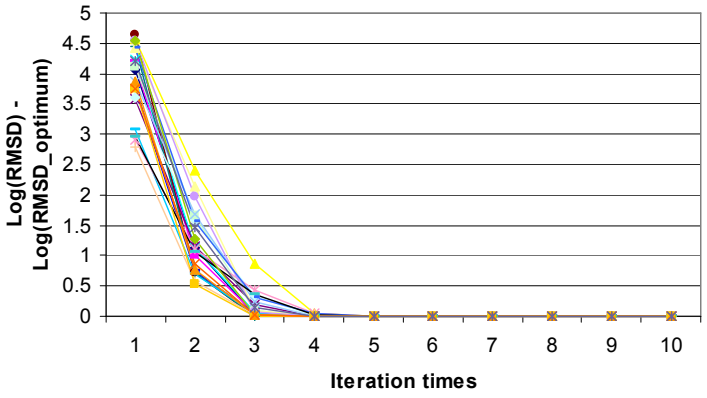


Figure 4.2 Convergence of wRMSD for 23 protein families. Each structure starts with a random translation and rotation

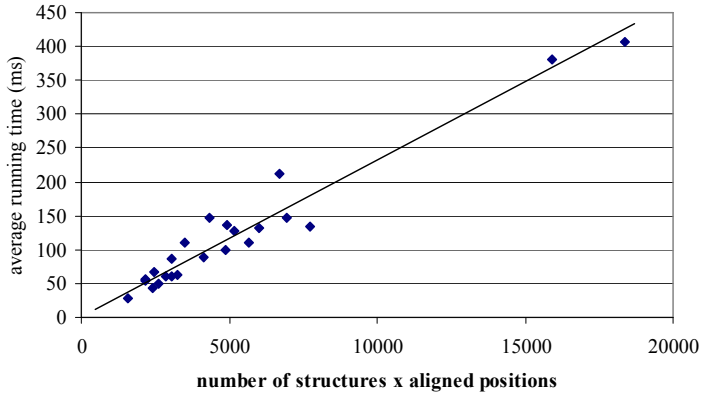


Figure 4.3 Average running time vs. number of atoms for 23 protein families

4.3.2 Optimizing aligned multiple structures in other programs

Existing multiple structure alignment algorithms optimize the aligned multiple structures once the correspondence is determined. I run Algorithm 4.1 on their aligned structures to see how much their results can be improved. There are five web servers for multiple structure alignment available: CE-MC [Guda01], MAMMOTH-multi [Lupyan05], MultiProt [Shatsky04], POSA [Ye05], and Superpose [Maiti04]. MultiProt and POSA provide ungapped alignment results only, so I compare Algorithm 4.1 to the web servers CE-MC, MAMMOTH-multi and Superpose.

I first run the 22 protein families (α-amylase_NC and α-amylase families use the same protein structures and we choose α-amylase_NC with longer aligned sequences) from Section III.A on each the three web servers using the defaulted setting. The results are shown in Table II and the best RMSDs are shown in bold.

From all three programs, algorithm1 reduces RMSDs for all the aligned structures. The aligned structures of CE-MC [Guda01] are optimized by Monte Carlo optimization and their RMSDs are very close to the minima. Some aligned structures from MAMMOTH-multi [Lupyan05] have large room to be minimized for RMSDs. It is interesting to see that overall the alignments by CE-MC are best. Most alignments by Superpose stuck at certain local minima and failed to achieve the better alignments as CE-MC and MAMMOTH-multi did.

Table 4.2 Comparison of RMSD of aligned protein families from CE-MC, MAMMOTH-mult, and Superpose programs and optimized wRMSD from algorithm 4.1 (best RMSD is in bold). We report the number of chains (*n*), the number of atoms aligned for each program (*m*), the RMSD from original alignment (CE, MAM, Sup), the wRMSD of the optimal alignment from our algorithm (wRMSD), and the improvement of RMSD (%diff)

Protein family	*n*	CE-MC vs. wRMSD	MAMMOTH-mult vs. wRMSD	Superpose vs. wRMSD
		m, CE, wRMSD, %diff	*m*, CE, wRMSD, %diff	*m*, CE, wRMSD, %diff
α-amylase_NC	23	PDB read failure*	932, 4.65, **4.58**, 7.0	No output•
asp	13	Broken chain failure#	376, 2.11, **2.10**, 0.5	366, 4.23, **4.13**, 2.4
cys	13	Broken chain failure	279, 1.61, **1.59**, 1.3	No output
fabp	17	PDB read failure	146, 1.87, **1.86**, 0.5	No output
ghf22	12	PDB read failure	131, 1.41, **1.40**, 0.7	No output
glob	41	Too many structures+	172, 2.06, **2.04**, 1.0	No output
gluts	14	PDB read failure	233, 2.83, **2.78**, 1.8	No output
grs	11	PDB read failure	552, 4.04, **3.75**, 7.7	No output
igvar-h	21	282, 3.17, **3.16**, 0.3	137, 2.13, **2.05**, 3.9	No output
kinase	15	PDB read failure	476, 7.45, **7.23**, 3.0	No output
ldh	14	PDB read failure	372, 2.55, **2.52**, 1.2	No output
lipocalin	15	Broken chain failure	204, 3.66, **3.58**, 2.2	No output
ltn	12	Broken chain failure	280, **1.42**, **1.42**, 0.0	281, 2.61, **2.58**, 1.2
p450	12	PDB read failure	504, 4.14, **4.05**, 2.2	568, 9.72, **9.43**, 3.1
phc	12	195, 1.66, **1.65**, 0.6	180, 3.59, **2.96**, 21.3	181, 5.33, **5.04**, 5.8
phoslip	18	PDB read failure	134, 1.55, **1.54**, 0.7	No output
proteasome	17	301, 2.32, **2.31**, 0.4	306, 4.70, **4.23**, 11.1	No output
sdr	13	PDB read failure	360, 3.98, **3.74**, 6.4	No output
sermam	27	Broken chain failure	252, 2.09, **2.07**, 1.0	No output
subt	11	PDB read failure	396, 6.40, **6.18**, 3.6	477, 5.93, **5.87**, 1.0
tim	10	PDB read failure	260, **1.44**, **1.44**, 0.0	262, 1.84, **1.82**, 1.1
uce	13	181, 1.65, **1.64**, 0.6	176, 3.42, **3.37**, 1.5	189, 3.67, **3.53**, 4.0

*At least one PDB file could not be read
#At least one protein sequence is broken into multiple chains, which is not supported by CE-MC
+CE-MC allows at most 25 sequences
•No output from Superpose for unknown reason

4.3.3 Finding Structural Conserved Regions

For RNA and protein molecules, structural conserved regions have great importance for classifying molecules, determining active site and functions, and applying homology modeling. RMSD has an inherent drawback that outliers have strong effects and cannot be used to determine the conserved regions. Many different measurements have been developed to determine the structural conserved regions [Altman94, Chew02]. Here I show that heuristic methods based on wRMSD can be developed to find conserved regions — overcoming the inherent drawback of RMSD. By modeling B-factors and deviations from the average positions as the weights, I demonstrate one heuristic to find well-aligned positions that determine the structural conserved regions.

Given n structures with m points (atoms) each and weights $w_{ik} \geq 0$ for each point, I use the following iterative steps to adjust weights:

1. Align the protein structures using the algorithm of Section 4.2.2.1 by setting $w_{ik} = e^{-bik/10}$, where b_{ik} is the B-factor for atom k in structure S_i for $(1 \leq i \leq n)$.
2. For each aligned position k, calculate the number of aligned atoms l, distance $d_{ik} = \|p_{ik} - \overline{p}_k\|$ for the l aligned structures, and the average squared distance $a_k = \sum_l d_{lk}^2 \Big/ l$. Then calculate the mean $\overline{a}$ and standard deviation σ of a_k.
3. If all $a_k \leq \overline{a} + 3\sigma$, then return;

   Otherwise set weights $w_k = e^{-bik/10}(l/n)/(0.1 + a_k)$ if $a_k \leq \overline{a} + 3\sigma$ for $(1 \leq k \leq m)$ and other weights to 0, align structures by wRMSD, and go to step 2.

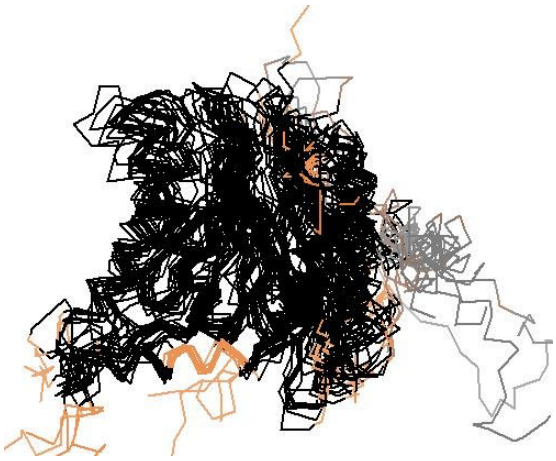
B-factor measures the mobility or uncertainty of a given atom position. In general, lower B-factor suggests that the atom position should be regarded as more precisely known. Thus,

in step 1, the term $e^{-bik/10}$ gives higher initial weights for those atoms whose positions are more accurate. In step 3, the weights are adjusted by two terms: The term $1/(0.1+a_k)$ encourages alignment in the positions where the average squared deviation, a_k, is small, and the term l/n encourages the positions with more aligned atoms. By combining these factors, I reduce the weights of outliers and enhance the weights of atoms in structural conserved regions.

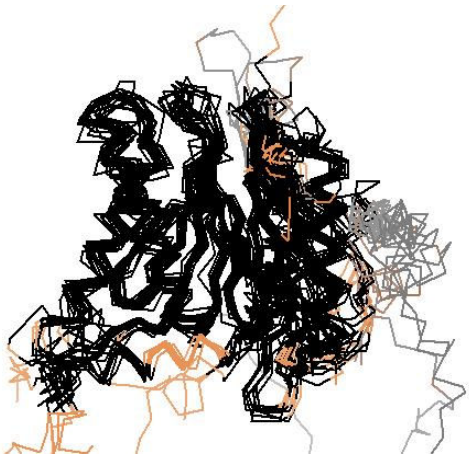
Figure 4.4 shows multiple structure alignments of the short-chain dehydrogenases/reductases (sdr) and proteasome families before (a,c) and after (b,d) optimizing the structural conserved regions. The alignments before optimizing the structural conserved regions are done with gapped alignment by Algorithm 4.1. From the figure, it can be seen clearly that the above iterative algorithm significantly improved the alignment of the structural conserved regions. The distributions of a_k for sdr and proteasome families are shown in Figure 4.5. For each structure, about 75% of the aligned positions in the optimized alignments have smaller a_k values than the unoptimized (gapped) alignments. The changes of wRMSD for regions $a_k \le \overline{a}$, $a_k \le \overline{a} + \sigma$, $a_k \le \overline{a} + 2\sigma$, and all a_k are shown in Table 4.3. For each structure, the wRMSD for the whole structure increases, but the wRMSDs for the first three regions decrease and the overall alignment is improved by achieving better alignments for the conserved regions.

a. Alignment of sdr family before optimizing the conserved region

b. Alignment of sdr family after optimizing the conserved region



c. Alignment of proteasome family before optimizing the structural conserved regions

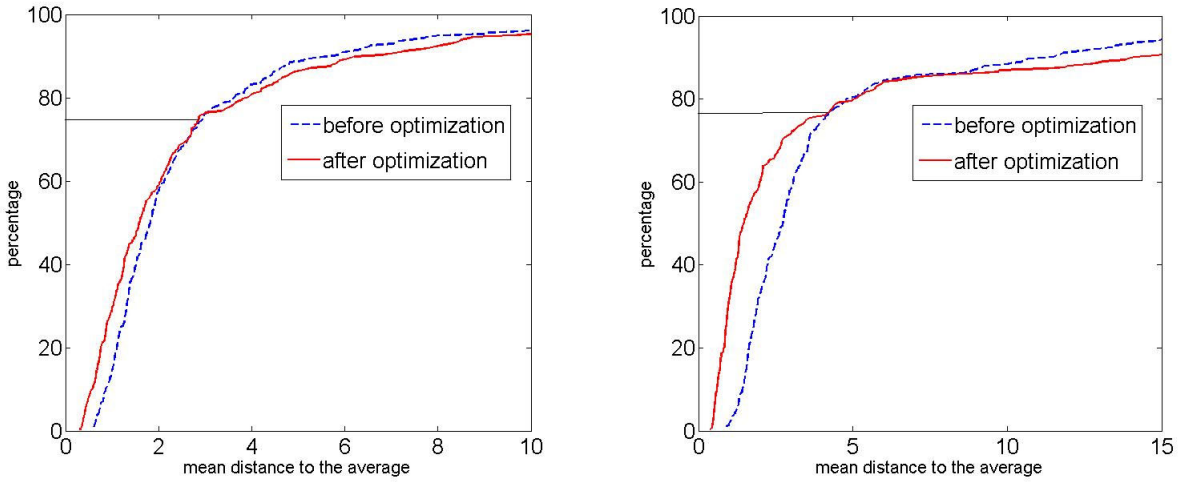


d. Alignment of proteasome family optimizing the structural conserved regions

Figure 4.4 Alignment of short-chain dehydrogenases/reductases (sdr) and proteasome families before and after optimizing the structural conserved regions. Positions are colored by number of standard deviations from average with black $a_k \leq \overline{a}$, peach $\overline{a} \leq a_k \leq \overline{a} + \sigma$, brown $\overline{a} + \sigma \leq a_k \leq \overline{a} + 2\sigma$, and gray $a_k > \overline{a} + 2\sigma$

Table 4.3 wRMSD before and after optimizing conserved regions for sdr and proteasome families

Region	$a_k \leq \overline{a}$	$a_k \leq \overline{a} + \sigma$	$a_k \leq \overline{a} + 2\sigma$	all
Sdr (before/after)	2.20, 1.84	2.60, 2.40	3.09, 3.12	3.80, 4.11
Proteosome (before/after)	3.46, 2.31	3.83, 3.01	4.18, 3.69	6.17, 6.94



a. sdr family **b.** proteasome family

Figure 4.5 The distribution of a_k

CHAPTER 5

MINING RNA TERTIARY MOTIFS

5.1 Introduction

An *RNA motif* is a short fragment of RNA (continuous or non-continuous) that appears repeatedly in a variety of RNA molecules and reflects specific local sequential or structural arrangement of RNA molecules. Identifying RNA motifs is an important step for understanding RNA structures and their functions [Leontis03], because natural selection in molecular evolution suggests that motifs with an important role are biased to appear.

RNA motifs have been classified into three types: A *sequence motif* is a common fragment of RNA sequences. A *secondary motif* is a common pattern of RNA base pairing relations, which form the scaffold of RNA structures and serve important biological roles like regulating cellular processes. A *tertiary motif* [Tamura02, Batey99, Hermann99] is a common pattern of spatial interactions between nucleotides that is related to biological functions such as stabilizing tertiary structures or binding metal ions. Although tertiary motifs are important for RNA folding and function, current RNA motif identification algorithms focus on finding sequence and secondary motifs, not tertiary motifs.

In this chapter, which includes joint work with Jun Huan and others, I apply graph database mining method to identify RNA tertiary motifs. The goal is automated motif discovery by (1) modeling RNA structures as graphs and (2) mining a graph database to

identify common subgraphs (tertiary motifs) from RNA. I represent the 3D structures of RNA molecules as a database of structure graphs, discover common subgraphs with a subgraph mining algorithm, and build consensus motifs (representatives of subgraphs in same groups) by geometric algorithms.

Each RNA structural graph includes three types of edges: *backbone edges* that encode connectivity along the primary sequence of an RNA molecule, *base pair edges* that encode base pair interaction of nucleotides, and *contact edges* that encode non-local contacts from the tertiary structure of the molecule. Thus I capture aspects of RNA primary, secondary and tertiary structures in the graph.

The frequent subgraph mining algorithm by Huan *et al.* [Huan03] is used to identify the frequently occurring subgraphs in a collection of RNA structure graphs. For each group of subgraphs, I derive consensus motifs by applying a multiple structure alignment algorithm that classifies mirror symmetric subgraphs as right or left handed and iteratively finds local optimal solution (consensus motif). With the alignment algorithm, I show that the aligned tertiary motifs fit well with a 3D Gaussian distribution model.

I demonstrate the overall utility of the algorithm on transfer RNA (tRNA) and ribosome RNA (rRNA). tRNA and rRNA are selected because of their abundance in known RNA structures and the extensive manual study in the SCOR database [Tamura04]. SCOR is a comprehensive database for recording RNA secondary and tertiary motifs that classifies RNA information into structural classification, functional classification, and tertiary interaction. By comparing the mined RNA tertiary motifs to the collections of motifs in tertiary interactions in SCOR, I show that this graph mining method can find known tertiary motifs, plus novel ones.

5.2 Related Work

Many RNA motif identification algorithms have been developed with various assumptions. Below, I review some major algorithms, which are classified into four groups.

The first group of motif identification algorithms involves manual processing to identify tertiary motifs. Klosterman *et al*. [Klosterman04] described examples of newly found RNA tertiary motifs, including extruded helical single strand, internal loop triples, and U-turns in internal loops. All these tertiary motifs are observed manually, but not discovered automatically by tools.

The second group of motif identification algorithms finds sequence motifs only. For example, Morgante *et al*. [Morgante05] use a graph representation of sequence and find common non-consecutive motifs for two or more sequences. Rajasekaran *et al*. [Rajasekaran05] find common sequence of length l with Hamming distance of d in t sequences of length n. Zhao *et al*. [Zhao05] find the similar DNA motifs based on a permuted Markov model.

The third group of motif identification algorithms uses simplified representations of RNA structures to find common structural motifs. COMPADRES [Wadley04] reduces RNA 3D structure to a sequence of contiguous P and C4' atoms and calculates and clusters the pseudo-dihedrals defined by P and C4' atoms. Huang *et al*. [Huang05] cut an RNA sequence into 6-nt fragments, compare their RMSD values, and cluster into a hierarchical structure by the unweighted pair group method with arithmetic mean (UPGMA). The structural motifs discovered by COMPARES and Huang *et al*. are limited to short consecutive sequences since they use no knowledge of secondary and tertiary interactions.

The fourth group of motif identification algorithms uses structure alignment to derive

tertiary motifs. ARTS [Dror05], which stands for alignment of RNA tertiary structures, compares two RNA sub-structures with sizes from two to thousands of nucleotides. It uses a set of base pairs as seed, compares their minimum RMSD every two consecutive base pairs, extends to the whole structures, and scores the matching. RAG [Gan04] represents RNA secondary structure as tree and dual-graph motifs, enumerates all possible motifs, and clusters based on topological characteristics. These methods have difficulty finding tertiary motifs because they do not consider spatial interactions.

In previous study, graph modeling and graph mining have been successfully applied to analyze 3D protein structures [Huan04]. Adapting the same technique to RNA analysis is non-trivial because of the following reasons: (1) Modeling RNA structure is different from that of protein structure: RNA structures are much larger and less stable than protein structures. (2) RNA is composed of 4 residues rather than 20 in proteins, which means that RNA graph mining has smaller set of node labels.

5.3 Algorithms for Mining RNA Tertiary Motifs

First, I define labeled graphs, which serve as the formal base of the graph representation of RNA molecules, and the data structure used by the frequent subgraph mining algorithm. Second, I discuss constructing graph representations for RNA molecules. Finally, I introduce the novel structure alignment algorithm for building consensus motifs.

5.3.1 Labeled Graphs and Frequent Subgraph Mining Algorithms

A labeled graph G is a quadruple $G = (V, E, \Sigma, \lambda)$. V is a set of nodes, $E \subseteq V \times V$ is a set of undirected edges joining distinct nodes, Σ is a set of node labels and edge labels, and the

labeling function λ defining the mappings from nodes and edges to their labels: $V \cup E \rightarrow \Sigma$. The *size* of a graph G is the cardinality of its node set V. A *graph database GD* is simply a group of labeled graphs. Figure 5.1 shows a graph database with three labeled graphs. The labels of nodes and edges are specified within the nodes and along the edges for each graph.

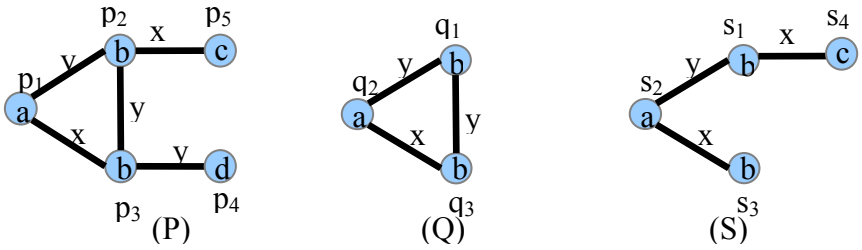


Figure 5.1 A database *GD* of three labeled graphs

From the graph theory, I formalize the search for tertiary motifs as the search for commonly occurring subgraphs in a group of graphs. A fundamental part of the frequent subgraph mining algorithm is to decide whether a subgraph G occurs in another graph G_o. To make this more precise, I define that a graph $G = (V, E, \Sigma, \lambda)$ is *subgraph isomorphic* to $G_o = (V_o, E_o, \Sigma_o, \lambda_o)$ if there exists a one-one mapping $f: V \rightarrow V'$ such that:

$\forall\ u \in V, \lambda(u) = \lambda_o(f(u)),$

$\forall\ u, v \in V, (u, v) \in E \Rightarrow (f(u), f(v)) \in E_o,$

$\forall\ (u, v) \in E,\ \lambda(u, v) = \lambda_o(f(u), f(v)).$

The one-one mapping f is defined as a subgraph isomorphism from G to G_o. Figure 5.1 shows a subgraph isomorphism f: $q_1 \rightarrow p_2$, $q_2 \rightarrow p_1$, and $q_3 \rightarrow p_3$ from graph Q to P, hence graph Q occurs in P through the subgraph isomorphism f. Huan *et al*. [Huan04] show an example of using labeled graphs in protein structures.

Given a graph database *GD*, which contains a set of graphs, the *support* of a subgraph G is the fraction of graphs in *GD* in which G occurs. Given a threshold $0 \leq \sigma \leq 1$, I define G to

be *frequent* if its support is at least σ. The goal of frequent subgraph mining is to identify all frequent subgraphs from a graph database *GD* with support threshold σ. Figure 5.2 shows all six frequent connected subgraphs with σ=1 from the three graphs of Figure 5.1.

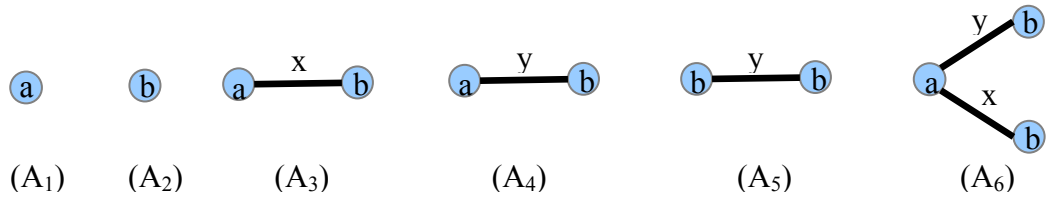


Figure 5.2 All frequent connected subgraphs from *G* in Figure 5.1 with support threshold σ = 100%

I use the Fast Frequent Subgraph Mining algorithm (FFSM), which is competitive or outperforms other state-of-art subgraph mining algorithms [Huan03].

5.3.2 Graph Modeling of RNA Molecules

In my graph representation of an RNA molecule, each node represents one nucleotide and each edge represents the connection for two nucleotides. I generate RNA graphs from RNA structures in the following way:

RNA molecules consist of four different nucleotides with the same backbone but different bases — A, C, G, and U. In the graph representation, each node corresponds to a nucleotide and is labeled either with purine (A and G) or pyrimidine (C and U). I reduce the alphabet to two because these nucleotides do not have significant structural differences, and it is common that mutated and wild-type RNAs have the same motif with different nucleotides [Leontis03]. I have tried the alphabet of all four symbols, but then I find very few tertiary motifs.

I generate three types of edges to represent RNA primary, secondary, and tertiary

structures in the following priority order:

a *backbone edge* connects two contiguous nucleotides,

a *base pair edge* connects nucleotides recorded as base paired in NDB [Berman92],

a *contact edge* connects spatial neighboring nucleotides within 8Å.

Backbone and base pair edges are labeled by their types. For each nucleotide pair, contact edges are labeled by discretized distances in the following way: Each nucleotide is abstracted as two points, its phosphorus atom and the geometric center of its sugar ring (since most tertiary interactions involve the phosphate and sugar groups. I define the distance between two nucleotides as the shortest distance between their abstracted points, and discretize this into distance bins, detailed in section 5.4.2.

I create one graph for each RNA structure, collect all the graphs into a graph database, and use the FFSM algorithm [Huan03] to mine frequent subgraphs.

5.3.3 Constructing Consensus Motifs with Computational Geometry

The graph representation in Section 5.3.2 abstracts away some of the precise geometry of motifs. After obtaining frequent subgraphs, I construct the corresponding tertiary motifs by the atom coordinates in 3D structure, and develop a novel multiple structure alignment algorithm that classifies mirror symmetric motifs as right or left handed and finds the optimal alignment by minimizing the sum of root mean squared distance (RMSD), which is widely used in measuring structure similarity in bioinformatics.

Given n motifs, G_1, G_2, …, G_n, each with m points in correspondence, e.g. p_{i1}, p_{i2}, …, p_{im} for motif G_i, I define the average motif $\overline{G}$ with points $\overline{p}_k = \frac{1}{n}\sum_{i=1}^{n} p_{ik}$ for $1 \le k \le m$, and define RMSD as the square root of the average of all squared pairwise distances between

motifs, $\sqrt{\frac{2}{mn(n-1)}\sum_{i=2}^{n}\sum_{j=1}^{i-1}\sum_{k=1}^{m}\left\|p_{ik}-p_{jk}\right\|^2}$, where $n(n–1)/2$ is the total number of motif pairs and m is the number of points in each motif. Since n and m are fixed, I can look for rigid transformations that minimize the summation. As mentioned in Chapter 4, I observe that the sum of all squared pairwise distances between n motifs equals n times the sum of squared distances to the average motif $\overline{G}$, i.e. $\sum_{i=2}^{n}\sum_{j=1}^{i-1}\sum_{k=1}^{m}\left\|p_{ik}-p_{jk}\right\|^2 = n\sum_{i=1}^{n}\sum_{k=1}^{m}\left\|p_{ik}-\overline{p}\right\|^2$.

To minimize RMSD, I translate and rotate/reflect motifs in 3D space to minimize the target function $\arg\min_{R,T}\left(\sum_{i=2}^{n}\sum_{j=1}^{i-1}\sum_{k=1}^{m}\left\|R_i p_{ik}-T_i-R_j p_{jk}+T_j\right\|^2\right)$, where R_i is a 3×3 rotation/reflection matrix and T_i as a 3×1 translation vector for motif G_i. Matrix R_i can be either a rotation (determinant = 1) or reflection (determinant = –1). After minimization, I classify all motifs into two handedness groups depending on whether reflection matrix gives better RMSD. The following algorithm iteratively aligns all motifs G_i for ($1 \le i \le n$) to $\overline{G}$, classifies mirror symmetric motifs, and updates the coordinates of $\overline{G}$ to minimize RMSD.

Algorithm 5.1. Given n motifs with m points each, classify and align motifs by performing the following steps:

1. Move the centroids of all G_i for ($1 \le i \le n$) to the origin.
2. Calculate the average motif $\overline{G}$ and $SD = \sum_{i=1}^{n}\sum_{k=1}^{m}\left\|p_{ik}-\overline{p}_k\right\|^2$.
3. Align G_i for ($1 \le i \le n$) to $\overline{G}$ by the optimal rotation or reflection matrix R_i, calculated by using the singular value decomposition (SVD) to determine the maximum eigenvalue of the covariance matrix N.

4. Calculate $SD^{new} = \sum_{i=1}^{n}\sum_{k=1}^{m}\left\|R_i p_{ik} - \overline{p}_k\right\|^2$.

5. If $SD - SD^{\text{new}} > \varepsilon$ (1.0×10^{-5} in tests), update $p_{ik} = R_i p_{ik}$ and $SD = SD^{\text{new}}$, calculate the average motif $\overline{G}$, and go to step 3; otherwise, go to step 6.

6. Set R_i = the product of all the rotation or reflection matrices for G, and classify G_i as right or left handed by the determinants of R_i (either 1 or –1).

This algorithm extends the algorithm presented in Chapter 4, which finds optimal alignment in nearly linear time but does not classify the motifs into right and left handed.

In each iteration, steps 1-5 need O(nm) each and step 6 needs O(n). The proof of convergence in Chapter 4 also applies to this algorithm. In experiments reported below, the number of iterations is small and the values reached are stable.

5.4 Experiments

5.4.1 Data Sets

A list of selected tRNAs and rRNAs used in this paper is shown in Table 5.1. In total I have 20 tRNAs, 3 5s rRNAs, 2 16s rRNAs, and 4 23s rRNAs. There are many examples of same RNA from same species binding to different proteins in NDB [Berman92]. I manually cleansed the data set with the following criteria to remove redundant ones:

A. From NDB with cutoff date December 22nd, 2005, I choose RNA with more than 90% nucleotides present.

B. For duplicated structures (from same species with same function), I keep the most recent one. If the time is the same, I keep the one with highest resolution.

C. For two structures with more than 70% of sequence similarity, I keep the more recent

one. If the time is the same, I keep the one with higher resolution.

D. I keep wild-type RNA and remove mutated RNA and synthesized RNA.

The tRNAs and rRNAs in Table 5.1 are the only available RNA molecules in NDB. Each RNA molecule is represented by a four-letter string, known as the protein databank identifiers (PDB ID). The fact that I have relatively few structures of rRNAs, some of which are large (especially the 23s rRNA), is a potential problem. FFSM determines frequent subgraphs by the number of graphs (structures) that have a subgraph, rather than the number of times a subgraph is found. This makes sense for identifying common structures in large families of related molecules, but I plan in future work to try to modify FFSM to count frequency by number of subgraphs for RNA.

Table 5.1 List of selected tRNAs and rRNAs (before December 22nd, 2005)

Type		Pdb Name
tRNA		1ehz, 1yfg, 1fir, 1qf6, 1qu2, 1eiy, 1f7u, 1il2, 1h4s, 2fmt, 1ivs, 1n78, 1j1u, 1j2b, 1u0b, 1wz2, 1zjw, 1h3e, 2csx, 1ser
rRNA	5s	1nkw (chain 9), 1s72 (chain 9), 1yl3 (chain B)
	16s	1fjg, 1pns
	23s	1nkw (chain 0), 1pnu (chain 0), 1s72 (chain 0), 1yl3 (chain A)

5.4.2 Identifying Tertiary Motifs

I identify motifs for tRNAs and rRNAs (5s, 16s and 23s) in two separate groups. For each group, I generate three different graphs using different bin sizes for contact edges (3, 4, or 5Å), with cutoff distance 8Å. This cutoff distance is large enough to capture the edges of most known tertiary motifs; I have tried larger cutoff distances but found too many contact edges, causing "noisy" occurrences of motifs. Lists of all the mined motifs can be found at

http://www.cs.unc.edu/~xwang/RNAGraph/.

Most of the mined motifs contain 4 nucleotides. RNA molecules are quite flexible and large frequent motifs are less likely to be found in the same topology. In trying larger cutoff distance (e.g. 18Å) for rRNAs, I find the largest mined motifs contain 8 nucleotides.

I compare the results to SCOR [Tamura04], which is a comprehensive database of RNA motifs identified by manual. As mentioned in section 5.3.2, the focus is to identify motifs that are involved in backbone interactions within a single chain, which fall into the tertiary motifs category in SCOR. Because I use the phosphorus, which is between two nucleotides, as one of the two points representing a nucleotide, I allow a shift of one nucleotide when comparing mined motifs to those of SCOR.

Note that all the motifs discussed in this paper involve backbone interactions only. I do not consider the backbone-base interactions. The contact distances are longer in the backbone-base interactions than the backbone-backbone interactions, and the number of contact edges and the noise in the data (motifs without biological meaning) significantly increase.

For rRNA, I choose a support threshold σ of 70% — that is, motifs must occur in 7 of the 9 graphs in the family to be considered frequent. The threshold is high because the 16s and 23s rRNAs are large and have many motifs. For example, for bin sizes of 3, 4, and 5Å I find 75, 260 and 152 distinct subgraphs in the 23s rRNA 1s72, respectively.

The ribose zipper is a tertiary motif formed by hydrogen bonds among the 2'–OH groups of sugars at two anti-parallel backbone strands. I identify 37 of the 43 ribose zippers recorded in SCOR (86%) for 23s rRNA 1s72. The number of found ribose zippers using different bin sizes is shown in Table 5.2. Note that all 37 identified ribose zippers are found with bin size

4Å, which occupies 14% of 260 total distinct subgraphs.

Table 5.2 Ribose zippers found in 23s rRNA 1s72

Bin size	3Å	4Å	5Å
Number of identified ribose zippers	12	37	8
Total found distinct subgraphs	75	260	152

There are five subcategories of ribose zippers (canonical, single, reverse single, naked and Cis) in 1s72 and I identify instances of each of them. Figure 5.3 shows a canonical ribose zipper (nucleotides 1078-1079 and 2077-2078, 23s rRNA 1s72).

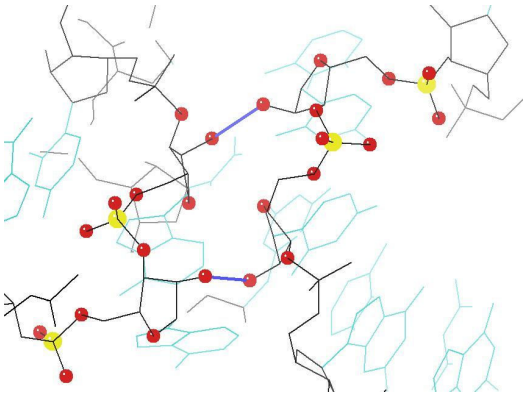


Figure 5.3 Canonical ribose zipper (nucleotides 1078-1079 and 2077-2078, 23s rRNA 1s72). Yellow ball is phosphorus, red ball is oxygen, and blue line is hydrogen bond

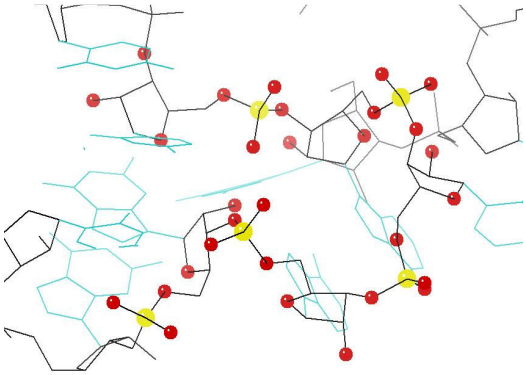


Figure 5.4 U turn motifs form by 5 continuous nucleotides (nucleotides 394-398, 23s rRNA 1s72), found by bin size = 4Å. Yellow ball is phosphorus and red ball is oxygen

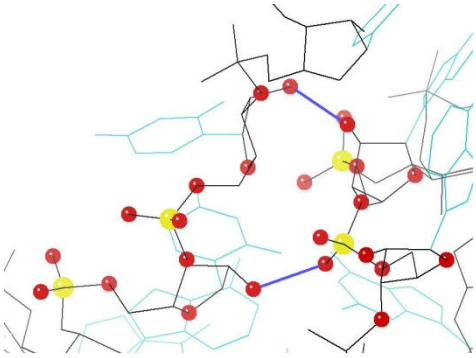


Figure 5.5 Tertiary interaction formed by a hydrogen bond (blue line) between two sugars and a hydrogen bond (blue line) between sugar and phosphorus (nucleotides 66-67 and 107-108, 23s rRNA 1s72), found by bin size = 3Å

I have also identified motifs classified as secondary motifs in SCOR. For example, Figure 5.4 shows a U-turn motif formed by five contiguous nucleotides (394-398); this method identifies four of them (nucleotides 394-395 and 397-398, 23s rRNA 1s72).

By carefully checking the mined motifs that do not match any existing motifs in SCOR, I find some interesting structures that could be good candidates for tertiary motifs. For example, Figure 5.5 shows a tertiary motif with one hydrogen bond between two sugars and another hydrogen bond between sugar and phosphorus (nucleotides 66-67 and 107-108, 23s rRNA 1s72).

For tRNA, I choose a support threshold σ of 20%, that is, motifs must occur in 4 of the 20 graphs in the family to be considered frequent. The threshold is much lower because tRNA is quite flexible and is much smaller than the large rRNA. I find several good candidates for tertiary motifs, available at http://www.cs.unc.edu/~xwang/RNAGraph/.

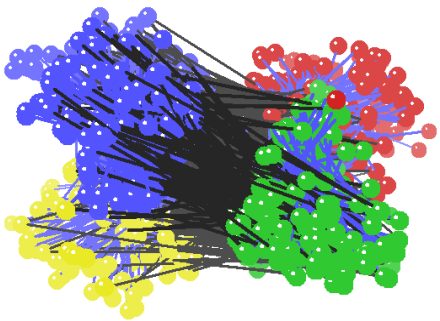
For the 20 tRNAs I choose, SCOR records only 5 tertiary motifs in 3 tRNA: 1ehz, 1yfg and 1fir. All the tertiary motifs are large (the smallest having 7 nucleotides), and no two tertiary motifs share the same topology. So for tRNA, I cannot compare the mined tRNA motifs with SCOR, because the support threshold of tertiary motifs of tRNAs in SCOR is too

low ($\sigma < 5\%$).

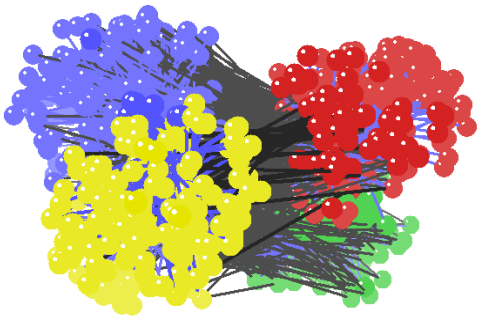
5.4.3 Consensus Motifs

I apply the multiple structure alignment algorithm to classify the structures of found tertiary motifs and generate consensus motifs. The alignment is done on a laptop with Pentium M 1.8GHz CPU and 784M memory. Table 5.3 shows the performance of aligning 12 motif groups by bin size = 4Å. The running time is collected from 1,000 tests on each motif group. I can see that when I classify mirror symmetric motifs, the RMSD is significantly decreased, along with the number of iterations and running time.

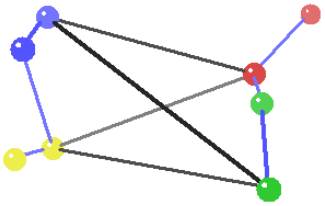
All the motif groups contain mirror symmetric motifs and better alignment is achieved when using the algorithm to classify and separate motifs by handedness, as shown in Figure 5.6. For the identified frequent motif groups, I did not find strong relationship for the handedness with the functions of motifs and the type of motifs. For example, all five types of ribose zippers can occur in both right and left handedness. It is an interesting problem whether all the tertiary motifs are independent of handedness and it is possible that the handedness is important for certain motifs.



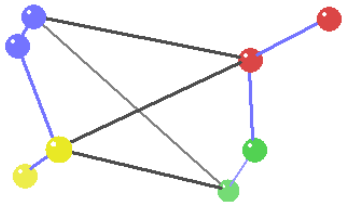
a. Aligning right hand occurrences of motif #12



b. Aligning left hand occurrences of motif #12



c. Consensus motif for right handed occurrences



d. Consensus motif for left handed occurrences

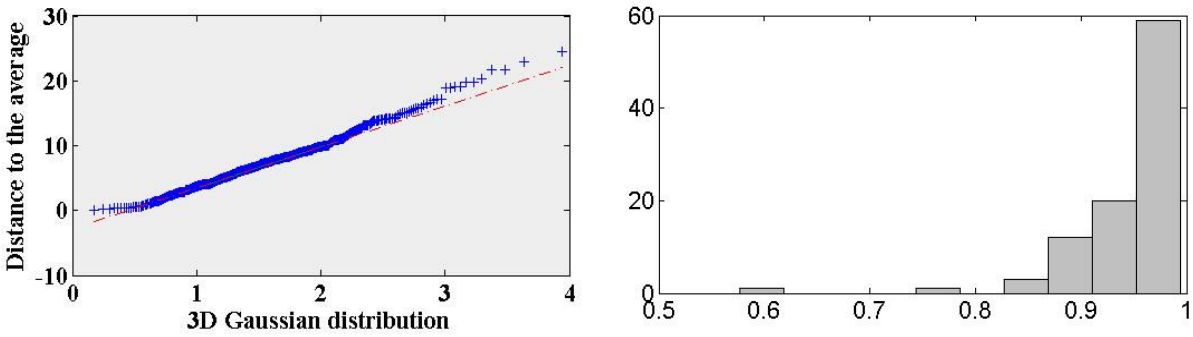
Figure 5.6 Example of aligning instances of motif #12. Two points in each of the four nucleotides are colored as yellow, red, green and blue. Blue line is backbone edges and black line is contact edges

Table 5.3 Performance for 12 mined motifs by bin size = 4Å

Motif ID	# of subgraphs all, 1s72, zipper	Motif RMSD	Motifs with reflection RMSD, iterations, time(s)	Right handed #	Right handed RMSD	Left handed #	Left handed RMSD
1	160, 19, 0	4.11	3.52, 6, 0.095 ± 0.005	81	3.44	79	3.47
2	202, 45, 7	3.93	3.53, 8, 0.158 ± 0.004	106	3.38	96	3.44
3	38, 8, 0	4.40	3.64, 4, 0.016 ± 0.005	20	3.56	18	3.72
4	10, 1, 0	3.54	3.35, 5, 0.005 ± 0.005	6	2.95	4	2.78
5	79, 21, 0	4.50	3.91, 5, 0.039 ± 0.003	41	3.74	38	3.93
6	53, 10, 0	3.81	3.60, 8, 0.041 ± 0.003	28	3.49	25	3.59
7	27, 7, 0	3.73	3.36, 7, 0.018 ± 0.004	15	3.29	12	3.04
8	396, 116, 5	4.32	3.80, 7, 0.288 ± 0.013	219	3.76	177	3.79
9	28, 7, 0	4.40	3.94, 4, 0.011 ± 0.004	15	3.83	13	3.92
10	16, 5, 0	3.94	3.85, 9, 0.014 ± 0.005	10	3.88	6	3.50
11	353, 76, 11	3.89	3.76, 16, 0.950 ± 0.576	192	3.54	161	3.72
12	361, 86, 24	4.05	3.56, 8, 0.382 ± 0.230	218	3.51	143	3.54

5.4.4 Statistical Analysis of Consensus Motifs

Deriving the statistical description of the aligned motifs is an intriguing question that has significant theoretical and practical implications. I test the null hypothesis that the distances of n atoms at a fixed position k to the average $\bar{p}_k$ are consistent with the distances from a 3D Gaussian distribution. The Gaussian is most used distribution function due to the central limit theorem of statistics, and previous studies hint that Gaussian is the best model to describe the aligned structures [Alexandrov04]. I adopt the Quantile-Quantile Plot (QQ plot) procedure [Evans00] to test the fitness of the aligned data to the 3D Gaussian model. Figure 5.7a shows QQ plot for phosphorus of first node in motif #12. The y-axis is the distance from each motif to the average for a fixed position and the x-axis is the quantile data from 3D Gaussian. The correlation coefficient R^2 = 0.993, which suggests that the data fits a 3D Gaussian model reasonably well. I carried out the same experiments for all the positions and the collected histogram of the correlation coefficient R^2 is shown in figure 5.7b. I identify that more than 88% of the positions I check have $R^2 > 0.9$.



a. QQ plot for phosphorus of first node in motif #12

b. Histogram of R^2 for all aligned positions

Figure 5.7 3D Gaussian distribution analysis of the distances from each point to average motif

CHAPTER 6

CONCLUSION

In this dissertation, I present three works that solve different problems for RNA and protein structures by exploiting different aspects of RNA and protein geometric properties. I show that when the proper geometric properties are extracted for corresponding structural problems, geometric algorithms solve the problems efficiently.

In the first work, I present the RNABC program that produces new clash-free conformations with acceptable geometry for a large fraction of RNA suites with local backbone problems. To my knowledge, RNABC is the first piece of software that aims to correct identified local problems in the backbone conformation of RNA structures. RNABC is freely available on multiple platforms, straightforward to run, executes quickly, and is suitable for routine crystallographic use.

While I have performed tests on correcting errors in completed structures, I believe that the best way to use RNABC is to incorporate it into the process of crystallographic refinement. By improving the geometry of RNA backbone earlier in the process of refinement and rebuilding, one can hope to improve the phases and map clarity at the next iteration, as has been done very successfully for protein backbone and sidechains [Arendall05].

Although RNABC does not guarantee to output the optimally correct answer every time, it seems probable that on-line diagnosis in the MolProbity validation site followed by

RNABC calculations and then re-refinement could significantly improve backbone conformation in almost any RNA crystal structure. These changes are often sufficiently large, and in sufficiently critical positions, that they would affect structure/function conclusions about biologically important RNA molecules.

In the second work, I analyze the problem of minimizing the multiple structure alignment using weighted RMSD. I show that the wRMSD for all pairs is the same as the wRMSD to the average structure. I also show that in general, translations and rotations cannot be decoupled when minimizing weighted RMSD, which makes the problem hard. To my knowledge, it is the first to achieve the optimum RMSD for both rotations and translations in weighted multiple structure alignment; previous works [Sutclife87, Verboon95] focus on optimizing rotations only.

Based on the property of the average structure, I create an efficient iterative algorithm to achieve optimum translations and rotations in minimizing wRMSD and prove its convergence. The 10,000 tests on each of 23 protein families from HOMSTRAD show that Algorithm 4.1 reaches the same local minimum regardless of the starting positions of structures, so the local minimum is most probably the global minimum. I further discuss the effects of outliers in the alignment using RMSD and present an iterative algorithm to find structural conserved region by iteratively assigning higher weights (by modeling the B-factors and deviations from the average positions) to better aligned positions until reaching convergence.

In the third work, I present an automated method of mining graph database to identify tertiary motifs in RNA structures. In this method, I defined a graph representation of RNA molecules and applied frequent subgraph mining algorithm for mining tertiary motifs. In

post-processing of the tertiary motifs, I develop a multiple structure alignment algorithm for classifying mirror symmetric motifs and finding consensus motifs, and show that the aligned motifs follow 3D Gaussian distribution model. The results show that the automated method can discover tertiary motifs in RNA molecules, despite limitations on the number of available RNA structures, and the fact that I included RNA only, but not the proteins that rRNA, in particular, interacts with.

6.1 Future Work

All three works in this dissertation present opportunities for extension in the future. For the first work, I plan to improve the RNABC program by analyzing the spatial arrangements of phosphate and base positions and allowing small movements of the anchored atoms to improve the ability to find alternative conformations from badly deviant starting conformations. The current RNABC program corrects one dinucleotide at a time; I plan to build a software pipeline to automatically correct longer strands of RNA backbone. Furthermore, I plan to build similar programs to find alternative conformations for some complex protein sidechains.

For the second work, the extension from RMSD to wRMSD and the property that average structure is the consensus lay a solid foundation for structure similarity analysis and provide new hints on many current questions. I plan to develop new algorithms to solve problems like query databases for similar structures, perform all-to-all structure comparison, detect dissimilar structure, determine structural conserved region, and calculate structure-based phylogenetic trees for RNA and protein families.

For the third work, I plan to use graph database mining to find fingerprint (i.e. distinct

motif) candidates for RNA families, find RNA and protein interface motifs, and investigate evolutionary relations among the tRNAs. Statistical analysis of the aligned RNA subgraphs is intriguing and I plan to investigate how Gaussian distribution model may help cluster RNA tertiary motifs.

APPENDIX I:

Theorem 4.4 The optimum translation T_i and the optimum rotation R_i for structure S_i ($1 \le i \le n$) satisfy the following n linear equations, of which n–1 are independent:

$$\sum_{k=1}^{m} w_{ik}\left(R_i p_{ik} - T_i\right) = \frac{1}{n}\sum_{k=1}^{m} w_{ik}\left(\sum_{l=1}^{n} \hat{w}_{lk}\left(R_l p_{lk} - T_l\right)\right)$$

Given all optimal rotations R_i for ($1 \le i \le n$) and one translation T_j ($1 \le j \le n$), the remaining n–1 optimal translations T_i for ($1 \le i \le n$, $i \ne j$) can be obtained by

$$T_i = T_j - \left(\sum_{l=1}^{n} R_l\left(\frac{1}{n}\sum_{k=1}^{m} p_{lk}\left(w_{ilk} - w_{jlk}\right)\right) - R_i\sum_{k=1}^{m} w_{ik}p_{ik} + R_j\sum_{k=1}^{m} w_{jk}p_{jk}\right)\Bigg/\sum_{k=1}^{m} w_{ik}\ .$$

Proof: I aim to find optimal rotations R_i and translations T_i to minimize the target function: $\arg\min_{R,T}\left(\sum_{i=1}^{n}\sum_{k=1}^{m} w_{ik}\left\|R_i p_{ik} - T_i - \overline{Rp}_k\,\overline{p}_k + \overline{T}_k\right\|^2\right)$, where $\overline{Rp}_k = \sum_{l=1}^{n} R_l \hat{w}_{lk} p_{lk} \Big/ \sum_{l=1}^{n} \hat{w}_{lk} p_{lk}$ and $\overline{T}_k = \frac{1}{n}\sum_{l=1}^{n} \hat{w}_{lk} T_l$.

Assume that I know the optimal rotations R_i for each structure S_i ($1 \le i \le n$) and I need to find optimal translations T_i.

Move each structure S_i by a vector A_i, where A_i satisfies equation:

$$\sum_{k=1}^{m} w_{ik}R_i\left(p_{ik} - A_i\right) = \frac{1}{n}\sum_{k=1}^{m} w_{ik}\left(\sum_{l=1}^{n} \hat{w}_{lk}R_l\left(p_{lk} - A_l\right)\right)$$

Let $q_{ik} = p_{ik} - A_i$, so $\sum_{k=1}^{m} w_{ik}R_i q_{ik} = \sum_{k=1}^{m} w_{ik}\,\overline{Rq}_k\,\overline{q}_k$, where $\overline{Rq}_k = \sum_{l=1}^{n} R_l \hat{w}_{lk} q_{lk} \Big/ \sum_{l=1}^{n} \hat{w}_{lk} q_{lk}$

The new average structure $\overline{S}$ from q_{ik} ($1 \le i \le n$, $1 \le k \le m$) has points:

$$\overline{q}_k = \frac{1}{n}\sum_{l=1}^{n}\hat{w}_{lk}q_{lk} = \frac{1}{n}\sum_{l=1}^{n}\hat{w}_{lk}\left(p_{lk} - A_l\right) = \overline{p}_k - \frac{1}{n}\sum_{l=1}^{n}\hat{w}_{lk}A_l = \overline{p}_k - \overline{A}_k$$

Note I have the equality:

$$\overline{Rp}_k\,\overline{p}_k = \left(\sum_{l=1}^{n}R_l\hat{w}_{lk}p_{lk}\Big/\sum_{l=1}^{n}\hat{w}_{lk}p_{lk}\right)\left(\frac{1}{n}\sum_{l=1}^{n}\hat{w}_{lk}p_{lk}\right) = \frac{1}{n}\sum_{l=1}^{n}R_l\hat{w}_{lk}p_{lk} = \frac{1}{n}\sum_{l=1}^{n}R_l\hat{w}_{lk}\left(q_{lk} + A_l\right)$$

$$= \frac{1}{n}\sum_{l=1}^{n}R_l\hat{w}_{lk}q_{lk} + \frac{1}{n}\sum_{l=1}^{n}R_l\hat{w}_{lk}A_l = \overline{Rq}_k\,\overline{q}_k + \overline{RA}_k\,\overline{A}_k, \text{ where } \overline{RA}_k = \sum_{l=1}^{n}R_l\hat{w}_{lk}A_l\Big/\sum_{l=1}^{n}\hat{w}_{lk}A_l\,.$$

So the target function after translation becomes:

$$\arg\min_{R,T}\left(\sum_{i=1}^{n}\sum_{k=1}^{m}w_{ik}\left\|R_iq_{ik} + R_iA_i - T_i - \overline{Rq}_k\,\overline{q}_k - \overline{RA}_k\,\overline{A}_k + \overline{T}_k\right\|^2\right)$$

Let $r_{ik} = R_iA_i - T_i - \overline{RA}_k\,\overline{A}_k + \overline{T}_k$, expand the target function to get:

$$\sum_{i=1}^{n}\sum_{k=1}^{m}w_{ik}\left\|R_iq_{ik} + R_iA_i - T_i - \overline{Rq}_k\,\overline{q}_k - \overline{RA}_k\,\overline{A}_k + \overline{T}_k\right\|^2 = \sum_{i=1}^{n}\sum_{k=1}^{m}w_{ik}\left\|R_iq_{ik} - \overline{Rq}_k\,\overline{q}_k + r_{ik}\right\|^2$$

$$= \sum_{i=1}^{n}\sum_{k=1}^{m}w_{ik}\left\|R_iq_{ik} - \overline{Rq}_k\,\overline{q}_k\right\|^2 + 2\sum_{i=1}^{n}\sum_{k=1}^{m}w_{ik}\left(R_iq_{ik} - \overline{Rq}_k\,\overline{q}_k\right)\cdot r_{ik} + \sum_{i=1}^{n}\sum_{k=1}^{m}w_{ik}\left\|r_{ik}\right\|^2$$

From the definition of A_i, I have $\sum_{k=1}^{m}w_{ik}R_iq_{ik} = \sum_{k=1}^{m}w_{ik}\,\overline{Rq}_k\,\overline{q}_k$ for ($1 \le i \le n$), so the second term is zero and I am left with the first and third terms. The first term does not depend on T_i for ($1 \le i \le n$) and $w_{ik} \ge 0$ for ($1 \le i \le n$, $1 \le k \le m$), so the target function is minimized by setting $r_{ik} = 0$.

Expand r_{ik} and re-arrange, so:

$$R_iA_i - T_i = \overline{R}_{kA}\,\overline{A}_k - \overline{T}_k = \frac{1}{n}\sum_{l=1}^{n}R_l\hat{w}_{lk}A_l - \frac{1}{n}\sum_{l=1}^{n}\hat{w}_{lk}T_l = \frac{1}{n}\sum_{l=1}^{n}\hat{w}_{lk}\left(R_lA_l - T_l\right)$$

So the optimum translation is achieved when $T_i = R_iA_i$, i.e. T_i satisfies the following n linear equations:

$$\sum_{k=1}^{m} w_{ik}\left(R_i p_{ik} - T_i\right) = \frac{1}{n}\sum_{k=1}^{m} w_{ik}\left(\sum_{l=1}^{n} \hat{w}_{lk}\left(R_l p_{lk} - T_l\right)\right)$$

Next I show that at most n–1 equations for T_i ($1 \le i \le n$) are independent. If I sum the right side of the n equations, re-arrange the order of the summation, reduce the term $\sum_{i=1}^{n} w_{ik}$, and re-arrange the order of the summation, I have:

$$\frac{1}{n}\sum_{i=1}^{n}\sum_{k=1}^{m} w_{ik}\left(\sum_{l=1}^{n} \hat{w}_{lk}\left(R_l p_{lk} - T_l\right)\right) = \frac{1}{n}\sum_{k=1}^{m}\left[\left(\sum_{l=1}^{n} \hat{w}_{lk}\left(R_l p_{lk} - T_l\right)\right)\left(\sum_{i=1}^{n} w_{ik}\right)\right]$$

$$= \sum_{k=1}^{m}\left[\left(\sum_{l=1}^{n} w_{lk}\left(R_l p_{lk} - T_l\right)\Big/\sum_{l=1}^{n} w_{lk}\right)\left(\sum_{i=1}^{n} w_{ik}\right)\right] = \sum_{k=1}^{m}\sum_{i=1}^{n} w_{ik}\left(R_i p_{ik} - T_i\right) = \sum_{i=1}^{n}\sum_{k=1}^{m} w_{ik}\left(R_i p_{ik} - T_i\right)$$

The summation of the left right side of n equations equals to the sum of the right side of n equations, so at most n–1 equations are independent.

Last I solve T_i ($1 \le i \le n$) from the n equations. Divide the ith equation by $\sum_{k=1}^{m} w_{ik}$ for ($1 \le i \le n$) and rearrange, I have:

$$T_i - \frac{1}{n}\sum_{l=1}^{n} \hat{w}_{lk} T_l = \left(\sum_{k=1}^{m} w_{ik} R_i p_{ik} - \frac{1}{n}\sum_{k=1}^{m}\sum_{l=1}^{n} w_{ilk} R_l p_{lk}\right)\Big/\sum_{k=1}^{m} w_{ik}$$

Let $a_l = \frac{1}{n}\hat{w}_{lk}$ and $b_i = \left(\sum_{k=1}^{m} w_{ik} R_i p_{ik} - \frac{1}{n}\sum_{k=1}^{m}\sum_{l=1}^{n} w_{ilk} R_l p_{lk}\right)\Big/\sum_{k=1}^{m} w_{ik}$, the n equations become:

$$T_i - \sum_{l=1}^{n} a_l T_l = b_i$$

Rewrite n equations in determinant form, negate equation 1 and add to equations 2,…, n, I have:

$$\begin{pmatrix} 1-a_1 & -a_2 & \dots & -a_n \\ -a_1 & 1-a_2 & \dots & -a_n \\ \dots & \dots & \dots & \dots \\ -a_1 & -a_2 & \dots & 1-a_n \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ \dots \\ b_n \end{pmatrix}$$

By fixing one translation T_j ($1 \le j \le n$), the remaining n–1 translations are:

$$T_i = T_j + b_i - b_j$$

$$= T_j + \left(\sum_{k=1}^{m} w_{ik} R_i p_{ik} - \frac{1}{n} \sum_{k=1}^{m} \sum_{l=1}^{n} w_{ilk} R_l p_{lk} \right) \Big/ \sum_{k=1}^{m} w_{ik} - \left(\sum_{k=1}^{m} w_{jk} R_j p_{jk} - \frac{1}{n} \sum_{k=1}^{m} \sum_{l=1}^{n} w_{jlk} R_l p_{lk} \right) \Big/ \sum_{k=1}^{m} w_{jk}$$

BIBLIOGRAPHY

[Adams02] Adams, P.D., Grosse-Kunstleve, R.W., Hung, L.W., Ioerger, T.R., McCoy, A.J., Moriarty, N.W., Read, R.J., Sacchettini, J.C., Sauter, N.K., Terwilliger, T.C., "PHENIX: building new software for automated crystallographic structure determination", *Acta Cryst. D.* **58**: 1948–1954, 2002.

[Adams04] Adams, P.L., Stahley, M.R., Kosek, A.B., Wang, J., Strobel, S.A., "Crystal structure of a self-splicing group I intron with both exons", *Nature.* **430**(6995): 45–50, 2004.

[Alexandrov04] Alexandrov, V., Gerstein, M., "Using 3D hidden Markov models that explicitly represent spatial coordinates to model and compare protein structures", *BMC Bioinformatics.* **5**:2, 2004.

[Altman94] Altman, R.B., Gerstein, M., "Finding an average core structure: application to the globins", *Proc. 2nd Int. Conf. Intell. Syst. Mol. Biol.* 19–27, 1994.

[Arendall05] Arendall, W.B. III, Tempel, W., Richardson, J.S., Zhou, W., Wang, S., Davis, I.W., Liu, Z.J., Rose, J.P., Carson, W.M., Luo, M., Richardson, D.C., Wang, B.C., "A test of enhancing model accuracy in high-throughput crystallography", *J. Struct. Funct. Genomics.* **6**(1), 1–11, 2005.

[Ban00] Ban, N., Nissen, P., Hansen, J., Moore, P.B., Steitz, T.A., "The complete atomic structure of the large ribosomal subunit at 2.4Å resolution", *Science.* **289**(5481): 905–920, 2000.

[Bates98] Bates, S., Phillips, A.C., Clark, P.A., Stott, F., Peters, G., Ludwig, R.L., Vousden, K.H., "p14ARF links the tumour suppressors RB and p53", *Nature*, **395**(6698): 124–125, 1998

[Batey99] Batey, R.T., Rambo, R.P., Doudna, J.A., "Tertiary motifs in RNA structure and folding", *Angew Chem Int Ed*. **38**(16): 2326–2343, 1999.

[Batey04] Batey, R.T., Gilbert, S.D., Montange, R.K., "Structure of a natural guanine-responsive riboswitch complexed with the metabolite hypoxanthine", *Nature.* **432**(7015): 411–415, 2004.

[Berman92] Berman, H.M., Olson, W.K., Beveridge, D.L., Westbrook, J., Gelbin, A., Demeny, T., Hsieh, S.H., Srinivasan, A.R., Schneider, B., "The nucleic acid database. A comprehensive relational database of three-dimensional structures of nucleic acids", *Biophys. J.*, **63**(3):751–759, 1992.

[Berman00] Berman, H.M., Westbrook, J., Feng, Z., Gilliland, G., Bhat, T.N., Weissig, H., Shindyalov, I.N., Bourne, P.E., "The Protein Data Bank", *Nucleic Acids Res.* **28**(1): 235–242, 2000.

[Bonneau01] Bonneau, R., Tsai, J., Ruczinski, I., Chivian, D., Rohl, C., Strauss, C.E., Baker, D., "Rosetta in CASP4: progress in ab initio protein structure prediction", *Proteins*. **45**(S5): 119–126, 2001.

[Branden99] Branden, C., Tooze, J., *Introduction to protein structure*, 2nd ed. Garland Publishing, Inc., New York, 1999.

[Brunger92] Brunger, A.T., "Free R value: a novel statistical quantity for assessing the accuracy of crystal structures", *Nature.* **355**: 472–475, 1992.

[Brunger98] Brunger, A.T., Adams, P.D., Clore, G.M., DeLano, W.L., Gros, P., Grosse-Kunstleve, R.W., Jiang, J.S., Kuszewski, J., Nilges, M., Pannu, N.S., Read, R.J., Rice, L.M., Simonson, T., Warren, G.L., "Crystallography & NMR system: A new software suite for macromolecular structure determination", *Acta Cryst. D.* **54**: 905–921, 1998.

[Chapman95] Chapman, M.S., "Restrained real-space macromolecular atomic refinement using a new resolution-dependent electron-density function", *Acta Cryst*. **A51**(1): 69–80, 1995.

[Chen04] Chen, J.L., Greider, C.W., "Telomerase RNA structure and function: implications for dyskeratosis congenita", *Trends Biochem Sci.* **29**(4): 183–192, 2004.

[Chew02] Chew, L.P., Kedem, K., "Finding the consensus shape for a protein family", *Algorithmica*, **38**(1): 115–129, 2002.

[Claverie05] Claverie, J.M., "Fewer genes, more non-coding RNA", *Science.* **309**(5740): 1529–1530, 2005.

[Conners07] Conners, R., Konarev, A.V., Forsyth, J., Lovegrove, A., Marsh, J., Joseph-Horne, T., Shewry, P., Brady, R.L., "An unusual helix-turn-helix protease inhibitory motif in a novel trypsin inhibitor from seeds of veronica (*Veronica hederifolia L.*)", *J. Biol. Chem.*, **282**(38): 27760–27768, 2007.

[Correll03] Correll, C.C., Beneken, J., Plantinga, M.J., Lubbers, M., Chan, Y.L., "The common and distinctive features of the bulged-G motif based on a 1.04Å resolution RNA structure", *Nucleic Acids Res.* **31**(23): 6806–6818, 2003.

[Crick70] Crick, F., "Central dogma of molecular biology" *Nature.* **227**(5258): 561–563, 1970.

[Dame05] Dame, R.T., "The role of nucleoid-associated proteins in the organization and compaction of bacterial chromatin", *Mol. Microbiol.* **56**(4): 858–870, 2005.

[Davis04] Davis, I.W., Murray, L.W., Richardson, J.S., Richardson, D.C., "MolProbity: structure validation and all-atom contact analysis for nucleic acids and their complexes", *Nucleic Acids Res.* **32**: W615–W619, 2004.

[Davis07] Davis, I.W., Leaver-Fay, A., Chen, V.B., Block, J.N., Kapral, G.J., Wang, X., Murray, L.W., Arendall, W.B. III, Snoeyink, J., Richardson, J.S., Richardson, D.C., "MolProbity: all-atom contacts and structure validation for proteins and nucleic acids" Nucleic Acids Res. **35**: W375–W383, 2007.

[Diamond71] Diamond, R., "A real-space refinement procedure for proteins", *Acta Cryst*. **A27**(5): 436–452, 1971.

[Diaz02] Diaz, M., Casali, P., "Somatic immunoglobulin hypermutation", *Curr Opin Immunol* **14**(2): 235–240, 2002.

[Doudna02] Doudna, J.A., Cech, T.R., "The chemical repertoire of natural ribozymes", *Nature.* **418**(6894): 222–228, 2002.

[Dror03] Dror, O., Benyamini, H., Nussinov, R., Wolfson, H.J., "Multiple structural alignment by secondary structures: Algorithm and applications", *Protein Science*, **12**: 2492–2507, 2003.

[Dror05] Dror, O., Nussinov, R., Wolfson, H., "ARTS: alignment of RNA tertiary structures", *Bioinformatics*, **21** Suppl 2: ii47–ii53, 2005.

[Duarte03] Duarte, C.M., Wadley, L.M., Pyle, A.M., "RNA structure comparison, motif search and discovery using a reduced representation of RNA conformational space", *Nucleic Acids Res.* **31**(16): 4755–4761, 2003.

[Dunbrack97] Dunbrack, R.L., Jr., Cohen, F.E., "Bayesian statistical analysis of protein sidechain rotamer preferences", *Protein Science*, **6**: 1661–1681, 1997.

[Ebert06] Ebert, J., Brutlag, D., "Development and validation of a consistency based multiple structure alignment algorithm", *Bioinformatics*, **22**(9): 1080–1087, 2006.

[Emsley04] Emsley, P., Cowtan, K., "Coot: model-building tools for molecular graphics", *Acta Crystallogr D.* **60**: 2126–2132, 2004.

[Evans00] Evans, M., Hastings, N., Peacock, B. *Statistical Distributions.* 3rd ed. New York, Wiley, 2000.

[Faux96] Faux, M.C., Scott, J.D., "Molecular glue: kinase anchoring and scaffold proteins", *Cell*, **85**: 9–12, 1996.

[Ferre-D'Amare98] Ferre-D'Amare, A.R., Zhou, K., Doudna, J.A., "Crystal structure of a hepatitis delta virus ribozyme", *Nature.* **395**(6702): 567–574, 1998.

[Fiser03] Fiser A., Sali A., "Modeller: generation and refinement of homology-based protein structure models", *Methods Enzymol.* **374**: 461–491, 2003.

[Frank06] Frank, J., *Three-Dimensional Electron Microscopy of Macromolecular Assemblies: Visualization of Biological Molecules in Their Native State*, Oxford University Press, New York, 2006.

[Gan04] Gan, H.H., Fera, D., Zorn, J., Shiffeldrim, N., Tang, M., Laserson, U., Kim, N., Schlick, T., "RAG: RNA-As-Graphs database--concepts, analysis, and features", *Bioinformatics*, **20**(8): 1285–1291, 2004.

[Gerstein98] Gerstein, M., Levitt, M., "Comprehensive assessment of automatic structural alignment against a manual standard, the SCOP classification of proteins", *Protein Science*, 7:445–456, 1998.

[Ginalski03] Ginalski, K., Elofsson, A., Fischer, D., Rychlewski, L., "3D-Jury: a simple approach to improve protein structure predictions", *Bioinformatics*, **19**(8): 1015–1018, 2003.

[Golden05] Golden, B.L., Kim, H., Chase, E., "Crystal structure of a phage Twort group I ribozyme-product complex", *Nature Struct. Mol. Biol.* **12**(1): 82–89, 2005.

[Guda01] Guda, C., Scheeff, E.D., Bourne, P.E., Shindyalov, I.N., "A new algorithm for the alignment of multiple protein structures using Monte Carlo optimization", *Proceedings of Pacific Symposium on Biocomputing*, 275–286, 2001.

[Hammes-Schiffer06] Hammes-Schiffer, S., Benkoviv, S.J., "Relating protein motion to catalysis", *Annual Review of Biochemistry*, **75**: 519–541, 2006.

[Hansen03] Hansen, J.L., Moore, P.B., Steitz, T.A., "Structures of five antibiotics bound at the peptidyl transferase center of the large ribosomal subunit", *J. Mol. Biol.* **330**(5): 1061–1075, 2003.

[Hermann99] Hermann, T., Patel, D.J., "Stitching together RNA tertiary architectures", *J Mol Biol*. **294**(4): 829–849, 1999.

[Holm96] Holm, L., Sander, C., "Mapping the protein universe", *Science*, **273**: 595–603, 1996.

[Horn87] Horn, B.K.P., "Closed-form solution of absolute orientation using unit quaternions", *Journal of the Optical Society of America A*, **4**(4): 629–642, 1987.

[Huan03] Huan, J., Wang, W., Prins, J., "Efficient mining of frequent subgraph in the presence of isomorphism", *in Proceedings of the 3rd IEEE International Conference on Data Mining (ICDM)*, 549–552, 2003.

[Huan04] Huan, J., Wang, W., Bandyopadhyay, D., Snoeyink, J., Prins, J., Tropsha, A., "Mining family specific residue packing patterns from protein structure graphs", *Proceedings of the 8th Annual International Conference on Research in Computational Molecular Biology (RECOMB)*, 308–315, 2004.

[Huang03] Huang, D.B., Vu, D., Cassiday, L.A., Zimmerman, J.M., Maher, L.J. III, Ghosh, G., "Crystal structure of NF-kappaB (p50)2 complexed to a high-affinity RNA aptamer", *Proc. Natl. Acad. Sci. USA*. **100**(16): 9268–9273, 2003.

[Huang05] Huang, H.C., Nagaswamy, U., Fox, G.E., "The application of cluster analysis in the intercomparison of loop structures in RNA", *RNA*. **11**(4): 412–423, 2005.

[Johnson74] Johnson, G.B., "Enzyme polymorphism and metabolism", *Science*, **184**: 28–37, 1974.

[Jones91] Jones, T.A., Zou, J.Y., Cowan, S.W., Kjeldgaard, M., "Improved methods for building protein models in electron-density maps and the location of errors in these models", *Acta Crystallogr A.* **47**: 110-119, 1991.

[Jovine00] Jovine, L., Djordjevic, S., Rhodes, D., "The crystal structure of yeast phenylalanine tRNA at 2.0Å resolution: cleavage by Mg(2+) in 15-year-old crystals", *J. Mol. Biol.* **301**(2): 401–414, 2000.

[Keeler05] Keeler, J., *Understanding NMR Spectroscopy*, Wiley, 2005.

[Kelley00] Kelley, L.A., MacCallum, R.M., Sternberg, M.J.E., "Enhanced genome annotation using structural profiles in the program 3D-PSSM", *J. Mol. Biol.* **299**(2): 501–522, 2000.

[Klein01] Klein, D.J., Schmeing, T.M., Moore, P.B., Steitz, T.A., "The kink-turn: a new RNA secondary structure motif", *EMBO J.* **20**(15): 4214–4221, 2001.

[Klein04] Klein, D.J., Moore, P.B., Steitz, T.A., "The roles of ribosomal proteins in the structure assembly, and evolution of the large ribosomal subunit", *J. Mol. Biol.* **340**(1): 141–177, 2004.

[Klein06] Klein, D.J., Ferre-D'Amare, A. R., "Structural basis of glmS ribozyme activation by glucosamine-6-phosphate", *Science.* **313**(5794): 1752–1756, 2006.

[Kleywegt04] Kleywegt, G.J., Harris, M.R., Zou, J.Y., Taylor, T.C., Wahlby, A., Jones, T.A., "The Uppsala electron-density server", *Acta Cryst.* D**60**: 2240–2249, 2004.

[Klosterman04] Klosterman, P.S., Hendrix, D.K., Tamura, M., Holbrook, S.R., Brenner, S.E., "Three-dimensional motifs from the SCOR: structural classification of RNA database: extruded strands, base triples, tetraloops and U-turn", *Nucleic Acids Res*. **32**(8): 2342–2352, 2004.

[Kolk98] Kolk, M.H., van der Graaf, M., Wijmenga, S.S., Pleij, C.W., Heus, H.A., Hilbers, C.W., "NMR structure of a classical pseudoknot: interplay of single- and double-stranded RNA", *Science.* **280**(5362): 434–438, 1998.

[Konagurthu06] Konagurthu, A.S., Whisstock, J.C., Stuckey, P.J., Lesk, A.M., "MUSTANG: a multiple structure alignment algorithm", *PROTEINS: Structure, Function, and Bioinformatics*, **64**(3): 559–574, 2006.

[Latchman97] Latchman, D.S., "Transcription factors: an overview", *Int. J. Biochem. Cell Biol.* **29**(12): 1305–1312, 1997.

[Leibowitz01] Leibowitz, N., Nussinov, R., Wolfson, H.J., "MUSTA--a general, efficient, automated method for multiple structure alignment and detection of common motifs: application to proteins", *Journal of Computational Biology*, **8**(2): 93–121, 2001.

[Leontis03] Leontis, N.B. Westhof, E., "Analysis of RNA motifs", *Curr Opin Struct Biol*. **13**(3): 300–308, 2003.

[Leontis06] Leontis, N.B., Altman, R.B., Berman, H.M., Brenner, S.E., Brown, J.W., Engelke, D.R., Harvey, S.C., Holbrook, S.R., Jossinet, F., Lewis, S.E., Major, F., Mathews, D.H., Richardson, J.S., Williamson, J.R., Westhof, E., "The RNA ontology consortium: an open invitation to the RNA community", *RNA*. **12**(4): 533–541, 2006.

[Lilley05] Lilley, D.M., "Structure, folding and mechanisms of ribozymes", *Curr Opin Struct Biol.* **15**(3): 313–323, 2005.

[Lin04] Lin, J.C., Duell, K., Konopka, J.B., "A microdomain formed by the extracellular ends of the transmembrane domains promotes activation of the G protein-coupled alpha-factor receptor", *Molecular Cell Biology*, **24**: 2041–2051, 2004.

[Lolle05] Lolle, S.J., Victor, J.L., Young, J.M., Pruitt, R.E., "Genome-wide non-mendelian inheritance of extra-genomic information in Arabidopsis", *Nature*, **434**(7032): 505–509, 2005.

[Long05] Long, S.B., Campbell, E.B., Mackinnon, R., "Crystal structure of a mammalian voltage-dependent Shaker family K+ channel", *Science* **309**(5736): 897–903, 2005.

[Lovell00] Lovell, S.C., Word, J.M., Richardson, J.S., Richardson, D.C., "The penultimate rotamer library" *Proteins: Struct Function and Genetics*, **40**: 389–408, 2000.

[Lovell03] Lovell, S.C., Davis, I.W., Arendall, W.B. III, de Bakker, P.I.W., Word, J.M., Prisant, M.G., Richardson, J.S., Richardson, D.C., "Structure validation by Cα geometry: φ, ψ and Cβ deviation", *Proteins: Structure, Function and Genetics.* **50**: 437–450, 2003.

[Lukavsky03] Lukavsky, P.J., Kim, I., Otto, G.A., Puglisi, J.D., "Structure of HCV IRES domain II determined by NMR", *Nature Struct. Biol.* **10**(12): 1033–1038, 2003.

[Lupyan05] Lupyan, D., Leo-Macias, A., Ortiz, A.R. "A new progressive-iterative algorithm for multiple structure alignment", *Bioinformatics*, **21**(15): 3255–3263, 2005.

[Maiti04] Maiti, R., Domselaar, G.H.V., Zhang, H., Wishart, D.S., "SuperPose: a simple server for sophisticated structural superposition," *Nucleic Acids Research*, **32**: W590–W594, 2004.

[Martick06] Martick, M., Scott, W.G., "Tertiary contacts distant from the active site prime a ribozyme for catalysis", *Cell.* **126**(2): 309–320, 2006.

[Mattick01] Mattick, J.S., "Non-coding RNAs: the architects of eukaryotic complexity", *EMBO Rep.* **2**: 986–991, 2001.

[McCarthy90] McCarthy, J.M., *Introduction to theoretical kinematics*, MIT Press, Cambridge, MA. 1990.

[McRee99] McRee, D.E., "XtalView/Xfit - a versatile program for manipulating atomic coordinates and electron density", *J Struct Biol.* **125**: 156–165, 1999.

[Mizuguchi98] Mizuguchi, K., Deane, C.M., Blundell, T.L., Overington, J.P., "HOMSTRAD: a database of protein structure alignments for homologous families", *Protein Science*, **7**: 2469–2471, 1998.

[Mohamed05] Mohamed, O.A., Jonnaert, M., Labelle-Dumais, C., Kuroda, K., Clarke, H.J., Dufort, D., "Uterine Wnt/beta-catenin signaling is required for implantation", *Proc Natl Acad Sci.* **102**(24): 8579–8584, 2005.

[Morgante05] Morgante, M., Policriti, A., Vitacolonna, N., Zuccolo, A., "Structured motifs search", *J Comput Biol*. **12**(8): 1065–1082, 2005.

[Morris92] Morris, A.L., MacArthur, M.W., Hutchinson, E.G., Thornton, J.M., "Stereochemical quality of protein structure coordinates", *Proteins.* **12**: 345–364, 1992.

[Murray99] Murray, H.L., Jarrell, K.A., "Flipping the switch to an active spliceosome", *Cell.* **96**: 599–602, 1999.

[Murray03] Murray, L.J., Arendall, W.B. III, Richardson, D.C., Richardson, J.S., "RNA backbone is rotameric", *PNAS.* **100**: 13904–13909, 2003.

[Murthy99] Murthy, V.L., Srinivasan, R., Draper, D.E., and Rose, G.D., "A complete conformational map for RNA", *J Mol Biol.* **291**(2): 313–327, 1999.

[Nielson05] Nielson, H., Westhof, E., Johansen, S., "An mRNA is capped by a 2', 5' lariat catalyzed by a Group I-like ribozyme", *Science.* **309**(5740): 1584–1587, 2005.

[Nigg95] Nigg, E.A., "Cyclin-dependent protein kinases: key regulators of the eukaryotic cell cycle", *Bioessays.* **17**(6): 471–480, 1995.

[Nilsen94] Nilsen, T.W., "RNA-RNA interactions in the spliceosome: unraveling the ties that bind", *Cell.* **78**: 1–4, 1994.

[Nissen00] Nissen, P., Hansen, J., Ban, N., Moore, P.B., Steitz, T.A., "The structural basis of ribosome activity in peptide bond synthesis", *Science.* **289**(5481): 920–930, 2000.

[Oberstrass06] Oberstrass, F.C., Lee, A., Stefl, R., Janis, M., Chanfreau, G., Allain, F.H., "Shape-specific recognition in the structure of the Vts1p SAM domain with RNA", *Nat Struct Mol Biol.* **13**(2): 160–167, 2006.

[Ochagavia04] Ochagavia, M.E., Wodak, S., "Progressive combinatorial algorithm for multiple structural alignments: application to distantly related proteins", *Proteins*, **55**(2): 436–454, 2004.

[Ortiz02] Ortiz, A.R., Strauss, C.E.M., Olmea, O., "Mammoth (matching molecular models obtained from theory): an automated method for model comparison", *Protein Sci.* **11**(11): 2606–2621, 2002.

[Parkinson96] Parkinson, G., Vojtechovsky, J., Clowney, L., Brünger, A.T., Berman, H.M., "New parameters for the refinement of nucleic acid containing structures", *Acta Crystallogr D Biol Crystallogr*. **52**: 57–64, 1996.

[Pennec96] Pennec, X., "Multiple registration and mean rigid shapes: Application to the 3D case", *Proceedings of the 16th Leeds Annual Statistical Workship*, 178–185, 1996.

[Perrakis99] Perrakis, A., Morris, R., Lamzin, V.S., "Automated protein model building combined with iterative structure refinement", *Nature Struct. Biol.* **6**(5): 458–463, 1999.

[Rajasekaran05] Rajasekaran, S., Balla, S., Huang, C.H., "Exact algorithms for planted motif problems", *J Comput Biol*. **12**(8): 1117–1128, 2005.

[Rhodes06] Rhodes, G., *Crystallography Made Crystal Clear: A Guide for Users of Macromolecular Models*, 3rd ed., Academic Press, 2006.

[Richardson01] Richardson, J.S., Richardson, D.C., "MAGE, PROBE, and Kinemages, chapter 25.2.8", In Rossmann,M.G. and Arnold,E. (eds), *International Tables for Crystallography*. Kluwer Academic Publishers, Dordrecht, the Netherlands. Vol. **F**: 727–730, 2001.

[Richardson08] Richardson, J.S., Schneider, B., Murray, L.W., Kapral, G.J., Immormino, R.M., Headd, J.J., Richardson, D.C., Ham, D., Hershkovits, E., Williams, L.D., Keating, K.S., Pyle, A.M., Micallef, D., Westbrook, J., Berman, H.M., "RNA backbone: consensus all-angle conformers and modular string nomenclature (an RNA Ontology Consortium contribution)", *RNA*, **14**: 465–481, 2008.

[Roux99] Roux, K., "Immunoglobulin structure and function as revealed by electron microscopy", *Int Arch Allergy Immunol* **120**(2): 85–99, 1999.

[Russell92] Russell, R.B., Barton, G.J., "Multiple protein sequence alignment from tertiary structure comparison: assignment of global and residue confidence levels", *PROTEINS: Structure, Function, and Genetics*, **14**: 309–323, 1992.

[Salehi-Ashtiani06] Salehi-Ashtiani, K., Luptak, A., Litovchick, A., Szostak, J.W., "A genomewide search for ribozymes reveals an HDV-like sequence in the human CPEB3 gene", *Science.* **313**(5794): 1788–1792, 2006.

[Sasisekharan69] Sasisekharan, V., and Lakshminarayanan, A.V., "Stereochemistry of nucleic acids and polynucleotides. VI. minimum energy conformations of dimethyl phosphate", *Biopolymers.* **8**: 505–514, 1969.

[Schluenzen00] Schluenzen, F., Tocilj, A., Zarivach, R., Harms, J., Gluehmann, M., Janell, D., Bashan, A., Bartels, H., Agmon, I., Franceschi, F., Yonath, A., "Structure of functionally activated small ribosomal subunit at 3.3 angstroms resolution", *Cell.* **102**(5): 615–623, 2000.

[Schneider04] Schneider, B., Moravek, Z., Berman, H.M., "RNA conformational classes", *Nucleic Acids Res.* **32**(5): 1666–1677, 2004.

[Schwede03] Schwede T., Kopp J., Guex N., Peitsch M.C., "SWISS-MODEL: an automated protein homology-modeling server", *Nucleic Acids Research*, **31**: 3381–3385, 2003.

[Serganov06] Serganov, A., Polonskaia, A., Phan, A.T., Breaker, R.R., Patel, D.J., "Structural basis for gene regulation by a thiamine pyrophosphate-sensing riboswitch", *Nature.* **441**(7097): 1167–1171, 2006.

[Shatsky04] Shatsky, M., Nussinov, R., Wolfson, H.J., "A method for simultaneous alignment of multiple protein structures", *Proteins*, **56**(1): 143–156, 2004.

[Shewchuk94] Shewchuk, J.R., *An Introduction to the Conjugate Gradient Method Without the Agonizing Pain*, 1994.

[Shih06] Shih, Y.L., Rothfield, L., "The bacterial cytoskeleton", *Microbiol. Mol. Biol. Rev.* **70**(3): 729–754, 2006.

[Soukup04] Soukup, J.K., Soukup, G.A., "Riboswitches exert genetic control through metabolite-induced conformational change", *Curr. Opin. Struct. Biol.* **14**: 344–349, 2004.

[Stahley05] Stahley, M.R., Strobel, S.A., "Structural evidence for a two-metal-ion mechanism of group I intron splicing", *Science.* **309**(5740): 1587–1590, 2005.

[Sussman76] Sussman, J.L., Kim, S., "Three-dimensional structure of a transfer RNA in two crystal forms", *Science.* **192**(4242): 853–858, 1976.

[Sutcliffe87] Sutcliffe, M.J., Haneef, I., Carney, D., Blundell, T.L., "Knowledge based modeling of homologous proteins, Part I: Three-dimensional frameworks derived from the simultaneous superposition of multiple structures", *Protein Engineering*, **1**(5): 377–384, 1987.

[Tamura02] Tamura, M., Holbrook, S.R., "Sequence and structural conservation in RNA ribose zippers", *J Mol Biol*. **320**(3): 455–474, 2002.

[Tamura04] Tamura, M., Hendrix, D.K., Klosterman, P.S., Schimmelman, N.R.B., Brenner, S.E., Holbrook, S.R., "SCOR: structural classification of RNA, version 2.0", *Nucleic Acid Res*. **32**: D182–184, 2004.

[Taylor94] Taylor, W.R., Flores, T.P., Orengo, C.A., "Multiple protein structure alignment", *Protein Science*, **3**: 1858–1870, 1994.

[Terwilliger02] Terwilliger, T.C., "Automated structure solution, density modification and model building", *Acta Cryst. D.* **58:** 1937–1940, 2002.

[Tomari05] Tomari, Y., Zamore, P.D., "Perspective: machines for RNAi", *Genes Dev.* **19**(5): 517–529, 2005.

[Torres-Larios05] Torres-Larios, A., Swinger, K.K., Krasilnikov, A.S., Pan, T., Mondragon, A., "Crystal structure of the RNA component of bacterial ribonuclease P", *Nature.* **437**(7058): 584–587, 2005.

[Verboon95] Verboon, P., Gabriel, K.R., "Generalized Procrustes analysis with iterative weighting to achieve resistance", *Br. J. Math. Statist. Psychol*, **48**: 57–74, 1995.

[Voet01] Voet, D., Voet, J.G., Pratt, C.W., *Fundamentals of Biochemistry*, John Wiley and Sons, New York, 2001.

[Wadley04] Wadley, L.M., Pyle, A.M., "The identification of novel RNA structure motifs using COMPADRES: an automated approach to structural discovery", *Nucleic Acids Res*. **32**(22): 6650–6659, 2004.

[Wang06] Wang, X., Snoeyink, J. "Multiple structure alignment by optimal RMSD implies that the average structure is a consensus", *Proceedings on LSS Computational Systems Bioinformatics Conference (CSB)*, 79–87, 2006.

[Wang07a] Wang, X., Huan, J., Snoeyink, J., Wang, W., "Mining RNA tertiary motifs with structure graphs", *19th International Conference on Scientific and Statistical Database Management* (SSDBM), 31, 2007.

[Wang07b] Wang, X., Snoeyink, J., "Defining and computing optimum RMSD for gapped multiple structure alignment", *The Workshop on Algorithms in Bioinformatics (WABI)*, 196–207, 2007.

[Wang08a] Wang, X., Kapral, K., Murray, L., Richardson, D., Richardson, J., Snoeyink, J., "RNABC: forward kinematics to reduce all-atom steric clashes in RNA backbone", *Journal of Mathematical Biology*, **56**: 253–278, 2008.

[Wang08b] Wang, X., Snoeyink, J., "Defining and computing optimum RMSD for gapped and weighted multiple structure alignment", *ACM/IEEE Transactions in Bioinformatics and Computational Biology*, submitted (invited paper), 2008.

[Weber01] Weber, R.E., Vinogradov, S.N., "Nonvertebrate hemoglobins: functions and molecular adaptations", *Physiol. Rev.* **81**(2): 569–628, 2001.

[White97] White, J.M., Hoffman, L.R., Arevalo, J.H., Wilson, I.A., "Attachment and entry of influenza virus into host cells. Pivotal roles of hemagglutinin", *Structural Biology of Viruses*, Oxford University Press, New York, 80–104, 1997.

[Wilson01] Wilson, P.D., "Polycystin: new aspects of structure, function, and regulation", *J. Am. Soc. Nephrol.* **12**(4): 834–45, 2001.

[Wimberly00] Wimberly, B.T., Brodersen, D.E., Clemons, W.M. Jr, Morgan-Warren, R.J., Carter, A.P., Vonrhein, C., Hartsch, T., Ramakrishnan, V., "Structure of the 30S ribosomal subunit", *Nature.* **407**(6802): 327–339, 2000.

[Winkler02] Winkler, W., Nahvi, A., Breaker, R.R., "Thiamine derivatives bind messenger RNAs directly to regulate bacterial gene expression", *Nature.* **419**(6910): 952–956, 2002.

[Word99a] Word, J.M., Lovell, S.C., LaBean, T.H., Taylor, H.C., Zalis, M.E., Presley, B.K., Richardson, J.S., Richardson, D.C., "Visualizing and quantifying molecular goodness-of-fit: small-probe contact dots with explicit hydrogen atoms", *J Mol Biol.* **285**(4): 1711–1733, 1999.

[Word99b] Word, J.M., Lovell, S.C., Richardson, J.S., Richardson, D.C., "Asparagine and glutamine: using hydrogen atom contacts in the choice of side-chain amide orientation", *J Mol Biol.* **285**(4): 1735–1947, 1999.

[Word00] Word, J.M., "All-atom small-probe contact surface analysis: An information-rich description of molecular goodness-of-fit", Ph.D thesis, Duke University, Durham, NC, 2000.

[Ye05] Ye, Y., Godzik, A., "Multiple flexible structure alignment using partial order graphs," *Bioinformatics*, **21**(10): 2362–2369, 2005.

[Zhang03] Zhang, Y., Kolinski, A., Skolnick, J., " TOUCHSTONE II: a new approach to ab initio protein structure prediction", *Biophysical Journal*, **85**: 1145–1164, 2003.

[Zhao05] Zhao, X., Huang, H., Speed, T.P., "Finding short DNA motifs using permuted Markov models", *Journal of Computational Biology*, **12**(6): 894–906, 2005.