2011

Increasing the Accuracy of Protein Three-Dimensional Structure Comparison

Mauricio J. Arriagada-Benitez

INCREASING THE ACCURACY OF PROTEIN THREE-DIMENSIONAL STRUCTURE COMPARISON

An Abstract of a Thesis

Submitted

In Partial Fulfillment

Of the Requirements for the Degree

Master of Science

Mauricio J. Arriagada-Benitez

University of Northern Iowa

July 2011

ABSTRACT

Structural alignment is an important tool for identifying structural and functional relationships between proteins. A typical protein structure alignment method is an iterative algorithm that computes an optimal residue-residue correspondence, once for each inspected spatial superposition of the input proteins.

The significance of this study allows us to better understand how important it is to explore in deep the analysis of proteins structure using a residue pair distance as the way to align two proteins. We assess the extent of improvements in the accuracy of the existing methods that can be made by exploring the search space in a more detailed manner.

As part of this study, we carried out two benchmarks. In the first benchmark, the improvement in the accuracies of three well known algorithms for protein structure comparison is assessed using a set of reference alignments generated by experts in the field. The second benchmark utilizes a set of commonly accepted measures of protein structure alignment quality.

INCREASING THE ACCURACY OF PROTEIN THREE-DIMENSIONAL STRUCTURE COMPARISON

A Thesis

Submitted

In Partial Fulfillment

Of the Requirements for the Degree

Master of Science

Mauricio J. Arriagada-Benitez

University of Northern Iowa

July 2011

This Study by: Mauricio J. Arriagada-Benitez

Entitled: INCREASING THE ACCURACY OF PROTEIN THREE-DIMENSIONAL STRUCTURE COMPARISON

Has been approved as meeting the thesis requirements for the

Degree of Master of Science

5/20/2011
Date — Dr. Aleksandar Poleksic, Chair, Thesis Committee

5/20/2011
Date — Dr. Mark A. Fienup, Thesis Committee Member

May 26, 2011
Date — Dr. Michael H. Walter, Thesis Committee Member

8/3/11
Date — Dr. Michael J. Licari, Dean, Graduate College

ACKNOWLEDGMENTS

At this stage of my life, I have not done this alone. Many people have participated somehow in this process and I would like to thank all of you for all your support.

First of all I would like to thank God for the opportunity of having achieved this degree far from my family. Thanks for giving me every day the strength and courage to continue moving forward.

I would like to thanks Dr. Poleksic, my advisor, who gave me all the knowledge to achieve my thesis and who believe in that I was able to fulfill this project. And also to my professors that guide me throughout my courses.

Finally, but not lees important, I would like to thank my mother who always believed in that I could successfully study abroad.

Finally, I appreciate to all my friends that helped me to finish my thesis and reviewing it. To my girlfriend, Gabriela, that has always been giving me all her support at distance.

TABLE OF CONTENTS

PAGE

LIST OF TABLES

TABLE PAGE

LIST OF FIGURES

FIGURE PAGE

CHAPTER 1

INTRODUCTION

Proteins are macromolecules that are present in all organisms. A protein is composed of amino acids, linked together by peptide bonds. Every protein has a specific role in an organism, depending on its chemical components and structure. There are four types of protein structure: primary, secondary, tertiary and quaternary. Primary structure is the sequence of amino acids. Secondary structure represents the local structural patterns, such as alpha helices and beta sheets. Tertiary structure is the shape of the protein, i.e. the way protein folds in the three dimensional space. Quaternary structure is the arrangement of multiple folds into a complex multiunit. It is well known that the function of a protein is largely determined by its three dimensional structure.

Every day the structures of new proteins are being discovered in biochemistry laboratories using different techniques such as nuclear magnetic resonance spectroscopy, NMR, (Wüthrich, 1990) and X-ray crystallography (Smyth & Martin, 2000). Often, an unknown three dimensional structure of a protein can also be determined, with high accuracy, from the primary sequence using computational methods, such as homology modeling or ab-initio techniques.

Once the atom coordinates of a protein have been determined, they are stored in databases such as the Proteins Data Bank, PDB, (Berman, 2008; Berman et al., 2000),

Protein Structure Alignment

Since the protein structure determines its function, protein three-dimensional structure comparison is one of the most important tasks in computational molecular biology. Figure 1 shows an example of a superposition of two proteins with the corresponding sequential alignment.

The alignment of protein structures is central to fields such as protein evolution, protein functional studies and drug design. Honig *et al.* particularly highlight the role of accurate methods for comparing the polypeptide folds in the *Structural Genomic Initiative* (Goldsmith-Fischman & Honig, 2003) – a project whose goal is to determine the structures of thousands of proteins from the protein universe.

Many different strategies have been proposed to search for an optimal alignment of a pair of proteins, including simulated annealing, genetic algorithms, and geometric hashing (Szustakowski & Weng, 2000). However, most state-of-art methods for pairwise alignment work by inspecting and scoring many different superpositions of the input proteins. Each superposition is obtained by rigidly transforming (rotating and translating) one of the proteins in the three-dimensional space, while the second protein is held fixed (Kolodny, Koehl, & Levitt, 2005). For each superposition of the input proteins, an optimal residue-residue correspondence is found using computational techniques such as dynamic programming or linear programming.

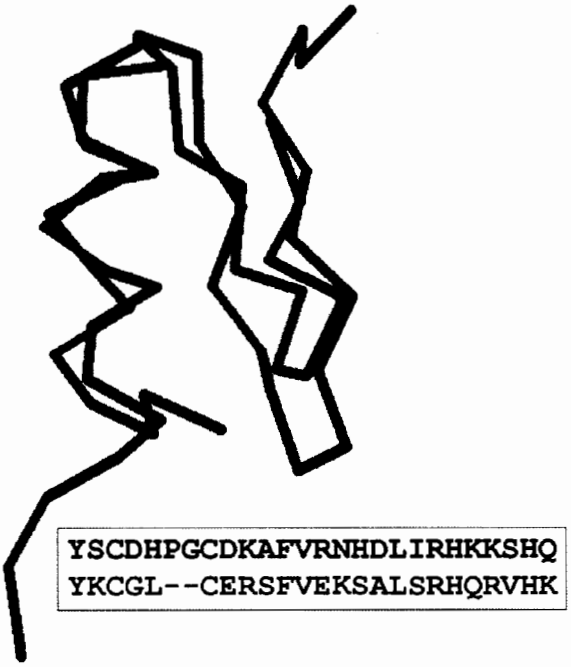

Figure 1. A toy example of a protein structural alignment and a corresponding sequence alignment. Note the small loop in the structure of the black protein corresponding to the insertion of the residues P and G in the alignment of primary structures.

One of the most important aspects of protein structure similarity is the choice of the alignment quality measure. Most alignment methods use the root mean square distance (RMSD), as the measure of pairwise structural homology. Low RMSD value indicates homology. However, this statistical measurement cannot be used in isolation, since it does not take into account the length of aligned regions. Hence, many alternative measures have been proposed to accurately assess the pairwise structural similarity (Structal, TM-align, LOCK, Dali, CE, etc). Perhaps one of the simplest such measures is the number of residues in the input proteins that can be fit under a specified distance from each other. In the rest of this thesis, we will denote this measure by MNumPairs(d),

where *d* represents the distance threshold in Angstroms. This measure, along with numerous other measures, can be efficiently optimized using *dynamic programming* – a technique in which an optimal solution to a problem is derived from optimal solutions to sub problems (Gerstein & Levitt, 1996). A variety of optimization tasks can be addressed using dynamic programming in diverse fields such as robotics (Sallaberger & D'eleuterio, 1995), economics (Grüne & Semmler, 2004), engineering (Pérez, Bossio, Moitre, & García, 2006), and so forth.

The main problem faced by methods for protein structure alignment is the infinite size of the space off all superpositions of the input proteins. Since only a finite subset of this space can be explored, the solution produced by current methods is almost never an optimal solution. The focus of this study is to estimate the size of gap between the solutions obtained by current methods and the optimal solutions.

Objectives

General Objectives

To estimate the difference in quality between optimal solution and heuristic solutions produced by current methodologies, we compare the accuracy of three popular alignments methods: Structal, TM-align, and LOCK2, before and after replacing each methods' superpositions by superpositions that optimize MNumPairs(*d*). The later superpositions are generated by the EPSILONOPTIMAL algorithm (Poleksic, 2009). EPSILONOPTIMAL is a computationally expensive procedure, but is capable of generating superposition that approximate MNumPairs(*d*) to any desired accuracy.

Specific Objectives

Specifics objectives are listed as follows:

- To study and implement a dynamic programming algorithms for local alignment used in Structal, TM-align and LOCK2
- To evaluate the performance of these methods on the superpositions that optimizes MNumPairs(d) in the *Sisyphus* benchmark for the alignment accuracy.
- To evaluate the performance of these methods on the superpositions that optimizes MNumPairs(d) in the *FSSP* benchmark for the alignment accuracy.
- To assess the increase in the precision of Structal, TM-align and LOCK2 in the benchmarks above.

Thesis Organization

After a brief literature review on protein structure, the Chapter 2 presents a theoretical framework of the structural protein alignment. Moreover, it gives an overview of dynamic programming and the Smith-Waterman algorithm as part of the procedures utilized for aligning protein structures.

The three methods studied here are described in Chapter 3 in details. The results and analysis are given in Chapter 4.

Finally, Chapter 5 presents the conclusions and the future work to be done in this study area.

CHAPTER 2

THEORETICAL FRAMEWORK OF THE STUDY

Methodology

We first give an overview of protein structure, the notion of a structural alignment and finally the description of the alignment methods analyzed as part of this study.

Protein Structure Framework

Proteins regulate a majority of processes in a living organism. Understanding the function of a protein helps understand the information carried by the corresponding gene (Griffiths, Wessler, Lewontin, & Carroll, 2008).

A protein, or polypeptide, is a chain composed of amino acids. There are 20 standard amino-acids. When two amino acids join, a water molecule is removed and a link between amino acids, called *peptide bond*, is formed.

There are four types of protein structure: primary, secondary, tertiary, and quaternary. The primary structure, shown in Figure 2, is simply the sequence of amino acids.

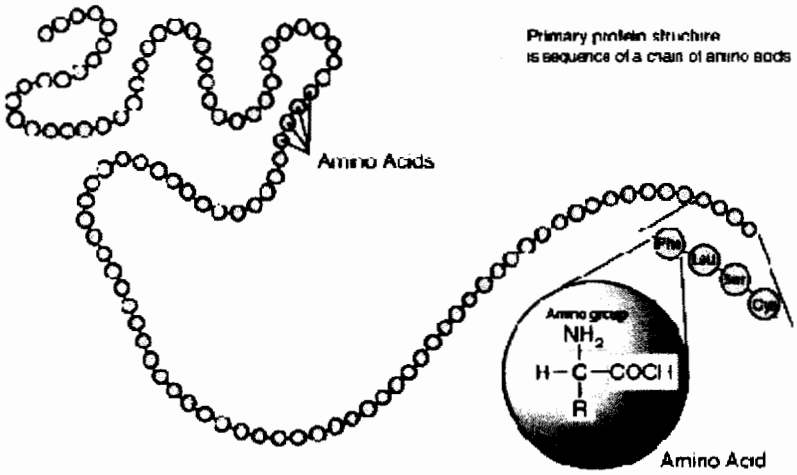

Figure 2. Primary structure of a protein. All amino acids share the same chemical composition except for the R group. Reprinted from Chemical Composition of Proteins: (Protein Structure), in The Biotechnology Project, n.d., Retrieved March 10, 2011, from http://biotech.matcmadison.edu/resources/proteins/labManual/chapter_2.htm

The secondary structure emerges as the fold of local regions from the polypeptide chain forms diverse shapes. Among the shapes, both β plated sheet and α-helix are the most common (Figure 3). Closer amino acids from the polypeptide chain bond together forming the secondary structure.

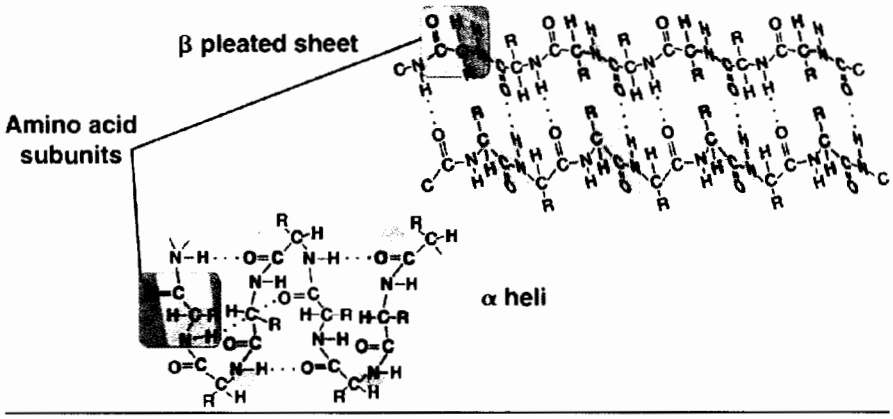

*Figure 3.*Secondary structure of a protein. The figure illustrates the β plated sheet and α-helix shapes. Reprinted from BIOMOLECULES, in Cyndie's Biology Homepage, n.d., Retrieved March 10, 2011, from http://faculty.cbu.ca/cglogowski/images/SecondaryStructure.jpg

The tertiary structure represents the way a protein folds in the three-dimensional space (Figure 4).

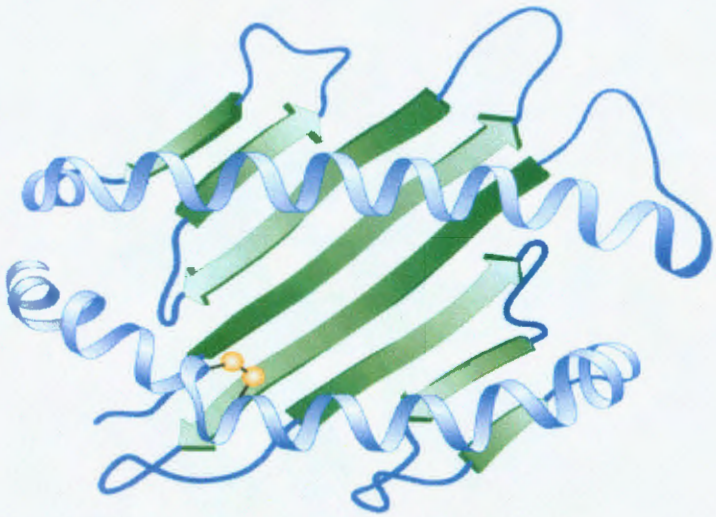

Figure 4. Tertiary structure of a protein. The β plated sheet is shown in green and α-helices in light blue. Adapted from Protein Structures: Primary, Secondary, Tertiary, Quaternary, in SCHOOLWORKHELPER-St. Rosemary Educational Institution, n.d., Retrieved March 10, 2011, http://schoolworkhelper.net/2010/11/protein-structures-primary-secondary-tertiary-quaternary/

The quaternary structure represents a complex of more than one polypeptide chain or protein subunit (Figure 5).

Figure 5. Quaternary structure of a protein. Each protein is shaded with different color. Reprinted from Quaternary Structure, in Department of Chemistry- The University of MAINE, n.d., Retrieved March 10, 2011, from http://chemistry.umeche.maine.edu/MAT500/Proteins12.html

The protein three-dimensional structure is typically determined through the experiments such as X-ray crystallography and NMR spectrometry. Current databases of protein structures, such as PDB, contain tens of thousands of entries, with this data exponentially increasing, largely due to the projects such as *Structural Genomics*.

The protein structural data is further organized by the degree of structural similarity. The Structural Classification of Proteins Database SCOP, classifies proteins depending on both their structure and evolutionary relationship into groups: Family, Superfamily, Common Fold, and Class (Murzin, Brenner, Hubbard, & Chothia, 1995). CATH, on the other hand, is an automatically generated database in which the polypeptide structures are classified into Class, Architecture, Topology, and Homologous

superfamily (Orengo et al., 1997). Thus, its name reflects for the four-level structural hierarchy.

The *FSSP* Database (Family of Structurally Similar Proteins) is similar in spirit to SCOP, except that the degree of protein structural homology in *FSSP* is determined using the DALI algorithm (Holm, Ouzounis, Sander, Tuparev, & Vriend, 1992).

Protein Structural Alignment

Because structure determines function, an alignment of protein structures plays a central role in many areas, from protein evolution to structure-based drug design. An optimal pairwise alignment is commonly found by exploring the space of all spatial superpositions of the input proteins. For each inspected superposition, an optimal residue-residue correspondence is computed using computational techniques, such as dynamic programming.

Dynamic Programming

Definition

Dynamic programming is a method that solves a problem by combining the solutions of sub problems. An example of dynamic programming is the Smith-Waterman algorithm for proteins structure.

Smith-Waterman

The Smith-Waterman algorithm (Smith & Waterman, 1981) is a widely used tool for aligning proteins sequences. Using dynamic programming, this procedure aligns local regions of protein or DNA sequences based on the alignment of subsequences. A similar

study conducted by (Needleman & Wunsch, 1970) presented an algorithm that generates a *global* sequence alignment.

The input to Smith-Waterman are two sequences $A = a_1a_2a_3 \dots a_n$ and $B = b_1b_2b_3 \dots b_m$. The algorithm proceeds in 3 steps: initialization (1), computing the entries of the dynamic programming matrix H (2) and the trace-back procedure for computing an optimal alignment (see also Figures 6 and 7).

$$H_{i,0} = H_{0,j} = 0 \quad \text{for} \quad 0 \leq i \leq n, 0 \leq j \leq m \tag{1}$$

$$H_{i,j} = \max \begin{cases} 0 & \\ H_{i-1,j-1} + w(a_i, b_j) & \text{Match/Mismatch} \\ H_{i-1,j} + w(a_i, -) & \textit{Deletion} \\ H_{i,j-1} + w(-, b_j) & \textit{Insertion} \end{cases} \tag{2}$$

$w(a_i, b_j)$ represents the substitution score for a_i and b_j, which is, in case of sequential alignments, specified in the mutation matrices, such as BLOSUM (Henikoff & Henikoff, 1992). For structural alignments, $w(a_i, b_j)$ is often a function of the Euclidean distance between the residues a_i and b_j i.e., between their representative atoms, such as alpha carbons C_α. $w(a_i, -)$ and $w(-, b_j)$ represent the penalties for residues insertions and deletions.

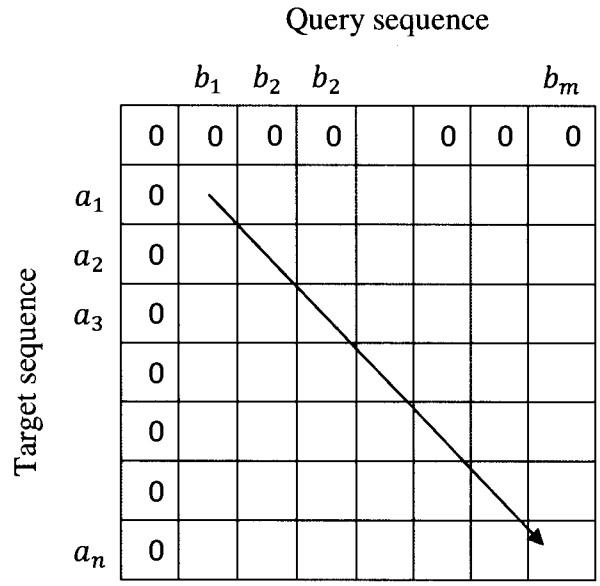

Figure 6. Dynamic programming matrix initialization. The dynamic programming matrix H is initialized with zeros as specified in (1). The remaining entries are computed either row-by-row or column-by-column.

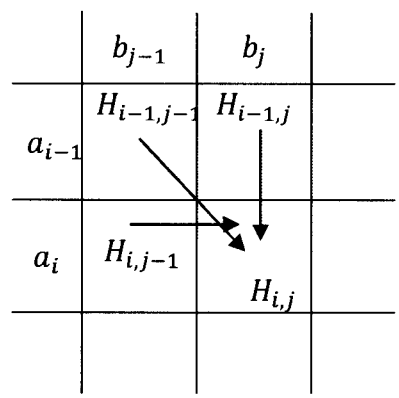

Figure 7. How to compute H matrix using dynamic programming. To compute $H_{i,j}$, we take the maximum score from the "previous" three cells. If the resulting score is negative, $H_{i,j}$ is set to 0.

The maximum entry of H represents the score of an optimal alignment.

To compute an optimal alignment (the pairing of residues) itself, the matrix H is traversed in the backward direction, starting from the cell with the highest score. The trace-back procedure determines the sequence of cells, in order, that gives rise to the cell with the highest score. Each cell in this sequence corresponds to a pair of aligned residues.

It should be noted that the algorithm by Smith and Waterman does not consider "affine gaps." An improvement of the Smith-Waterman algorithm, presented by (Gotoh, 1982) includes an affine gap model. In this model, the gap-opening penalty is assigned to the first gap and the gap-extension penalty is assigned to each consecutive gap. This idea is conceptually similar to the one described by (Smith & Waterman, 1981).

The initialization step is given by formulas (3) below (Gök & Yılmaz, 2006).

$$\begin{aligned} H_{i.0} &= \mathrm{I}_{i,0} = D_{i,0} = 0 \\ H_{0.j} &= \mathrm{I}_{0,\mathrm{j}} = D_{0,j} = 0 \end{aligned} \tag{3}$$

The dynamic programming matrices are filled out according to formulas (4)-(6). As before, $\mathrm{w}(a_i, b_j)$ denotes the match or mismatch cost for the residues a_i and b_j. The gap opening and extension costs are denoted by g_o and g_e, respectively.

$$H_{i,j} = \max \begin{cases} 0 \\ H_{i-1,j-1} + w(a_i, b_j) \\ I_{i,j} \\ D_{i,j} \end{cases} \tag{4}$$

$$I_{i,j} = \max\begin{cases} H_{i,j-1} - g_o \\ I_{i,j-1} - g_e \end{cases} \quad (5)$$

$$D_{i,j} = \max\begin{cases} H_{i-1,j} - g_o \\ D_{i-1,j} - g_e \end{cases} \quad (6)$$

Backtracking starts from the maximum scoring cell in the matrix and ends when 0 is reached. A couple of examples are presented in (Hochreiter, 2008).

The above model is implemented here in order to compute the methods' specific alignments based on the newly generated superpositions.

Protein Structure Alignment Methods Used In This Study

Structal

In 1996, Gerstein and Levitt presented a novel method that utilizes a variable position gap to improve a common pairwise alignment procedure (Levitt & Gerstein, 1998; Subbiah, Laurents, & Levitt, 1993). Considering two proteins $A = a_1, a_2, \dots, a_n$ and $B = b_1, b_2, \dots, b_m$ as sequences of points in the three dimensional space $\mathbb{R}^3$, this algorithm uses iterative dynamic programming to compute an alignment between subchains $(a_{1_k}, \dots, a_{i_k})$ and $(b_{1_k}, \dots, b_{i_k})$ of A and B that minimizes the $cRMS$ score:

$$cRMS = \sqrt{\frac{1}{k}\sum_{r=1}^{k} \left\| a_{i_r} - b_{i_r} \right\|^2} \quad (7)$$

In each iteration, the protein A is held fixed, while the protein B is rigidly transformed in space (Kabsch, 1976). The score matrix entries used in dynamic programming are computed according to the following formula:

$$S_{ij} = \frac{M}{1 + \left(\frac{d_{ij}}{d_0}\right)^2} \tag{8}$$

where $M = 20$, $d_0 = 5\text{Å}$ and d_{ij} represents the Euclidean distance between the residue i from the chain A and residue j from the chain B, $d_{i,j} = \|a_i - b_j\|$.

The Structal method aims to find a the subchains P and Q of A and B, respectively, along with the superposition of A and B that maximizes the $STRUCTURAL\_SCORE_{P,Q}$, given by

$$STRUCTURAL\_SCORE_{P,Q} = \max_{\hat{b}} \sum_{i=1}^{k} \frac{20}{1 + \frac{\|a_{p_i} - \hat{b}_{q_i}\|^2}{5}} - 10 \cdot G_{P,Q} \tag{9}$$

In the formula above, $\hat{b}_{q_i}$ represents the residue of the rigidly transformed sub chain Q and $G_{P,Q}$ is the total number of gaps in the alignment. This Structal program, along with the protein set samples, can be downloaded at http://csb.stanford.edu/levitt/Structal/

It is worthwhile to mention that it is possible to approximate the $STRUCTAL\_SCORE$ to any specified accuracy in polynomial time (Kolodny & Linial, 2004).

TM-Align

The TM-align method (Zhang & Skolnick, 2005) is widely used in protein structure analysis, in particular for the assessments of quality of protein models generated by comparative modeling or ab-inition techniques. It works by optimizing a specific measure of the alignment quality, called TM-score (Zhang & Skolnick, 2004). An improved version of the program, Fr-TM-align, has been recently published (Pandit & Skolnick, 2008)

The TM-score similarity matrix is given by:

$$S(i,j) = \frac{1}{1 + \frac{d_{ij}^2}{d_0(L_{min})^2}} \tag{10}$$

where d_{ij} represents the distance between the residue at position i in protein A and the residue at position j in protein B. The parameter d_0 depends on the length L_{min} of the smaller protein:

$$d_0(L_{min}) = 1.24\sqrt[3]{L_{min} - 15} - 1.8 \tag{11}$$

The gap opening and extension penalties, which are optimized for the best performance, are set to 0.6 and 0.0, respectively.

This program and installation manual are downloadable at http://zhanglab.ccmb.med.umich.edu/TM-align/

LOCK2

LOCK2 (Shapiro & Brutlag, 2004a) is an improved version of the original LOCK algorithm (Singh & Brutlag, 1997), which incorporates the secondary structure information into the alignment process. In contrast to Structal and TM-align, LOCK 2 uses an iterative procedure to minimize the *RMSD* between pair vectors of the secondary structure. The algorithm employs the threshold between a pair of residues of 3Å for the atomic superposition. Rigid transformations for *RMSD* minimization are realized using the methods of Horn (1987) and Horn, Hilden, and Negahdaripour (1988).

The LOCK2 web server is available at http://foldminer.stanford.edu/ (Shapiro & Brutlag, 2004b) and the software can be downloaded at http://lock2.stanford.edu/.

CHAPTER 3

TOOLS AND PROCEDURES

Tools

The algorithm development and benchmarking were carried out on a personal Dell laptop with Intel 64-bit architecture and 4GB RAM. The Ubuntu 9.10 Linux 32-bit operating system was installed, providing a suitable environment for running the third party software.

Various scripts for parsing the PDB files and running the algorithms were developed in PERL. We have chosen PERL over other scripting languages due to its simplicity in handling the data structures and hashes. Moreover, PERL is very good at text and string manipulation (Christiansen & Torkington, 2003).

Procedures

Our first goals were to install Structal, TM-align, and LOCK2 and to build a shell script (used in Linux to execute shell commands) for running these programs. Each method generates output in diverse formats, so the ability to parse text files is required.

One of the major technical problems encountered was the difference in the way these three algorithms process the input files. To circumvent this problem and to perform an unbiased analysis, we decided to develop our own alignment procedures implemented by the three methods (Structal, TM-align, and LOCK2) . Thus, the original software was run only to generate the output PDB files containing superimposed proteins.

A flowchart in Figure 8 shows the sequence of steps we followed to generate methods' specific alignments.

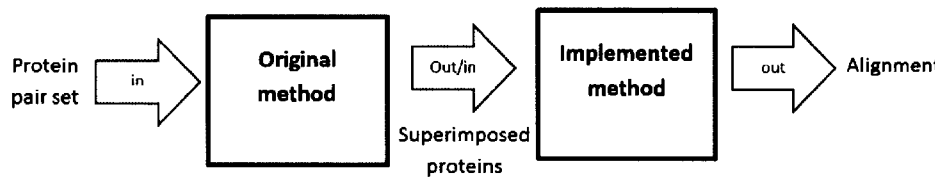

Figure 8. Implementation flow diagram. Original software for the three methods was used only to generate PDB files of superimposed proteins. These files were then given as input to our routines that generate method-specific alignments.

Both, the output PDB files containing pairs of proteins superimposed by Structal, TM-align, and LOCK2 and the PDB files containing the same pairs of proteins superimposed by EPSILONOPTIMAL are given as input to the replicated alignment procedures used in Structal, TM-align, and LOCK2 (Figure 9).

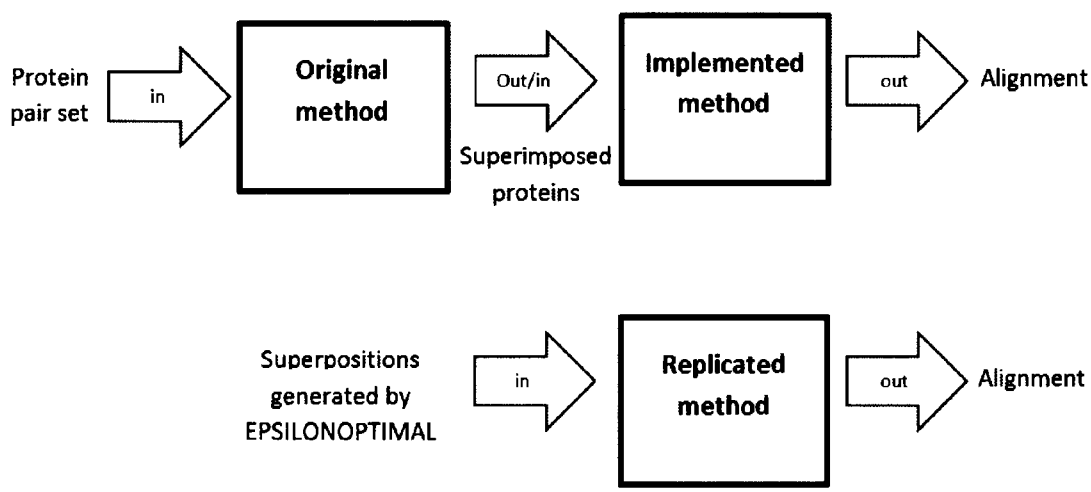

Figure 9. Specific implementation flow diagram. The first pipeline generates methods-s specific superpositions and alignments. The second pipeline generates alignments using the superpositions produced by EPSILONOPTIMAL.

Finally, the accuracy of the two sets of alignments is compared head-to-head, as described below.

Methods

Sisyphus Benchmark

The Sisyphus benchmark consists of 127 alignments of homologous proteins (Andreeva, Prlić, Hubbard, & Murzin, 2007). These alignments (from now on called the "reference alignments") were carefully constructed by human experts. It should be emphasized that *Sisyphus* is a difficult test in that the structural homology between the proteins from this set is difficult to detect using automated methods for protein structure matching. In order to directly compare the results of our benchmark with the results of some previous studies (Poleksic, 2009; Rocha, Segura, Wilson, & Dasgupta, 2009), we used only a subset of the Sisyphus set, consisting of 106 alignments between the single-chain proteins. The reference alignments are available for download at http://sisyphus.mrc-cpe.cam.ac.uk.

The alignment accuracy in the *Sisyphus* benchmark is defined as the agreement with the reference alignments. Aside from testing the absolute agreement, we also tested the accuracy allowing for the alignment shifts (errors). For the tolerance shift s, the agreement with reference alignment is defined as I_s/L_{ref} , where I_s is the number of aligned residues that are shifted by no more than s positions in the reference alignment and L_{ref} is the length of the reference alignment (Poleksic, 2009; Rocha et al., 2009).

An example is given in Figure 10.

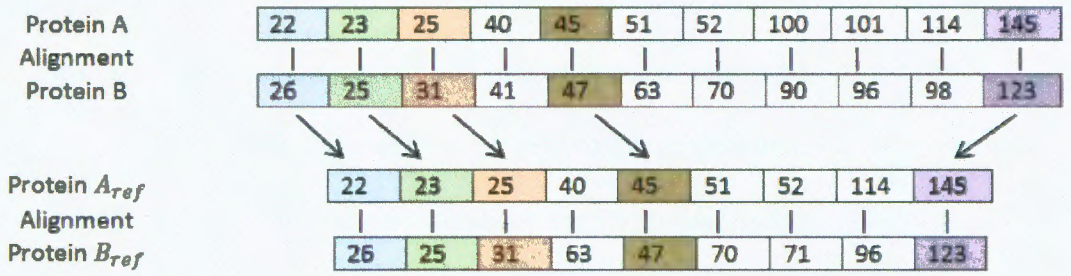

Figure 10. A 0-shift comparison of an alignment and the reference alignment. The two alignments agree only in 5 out of 9 positions.

Figure 11 shows an example of 1-shift.

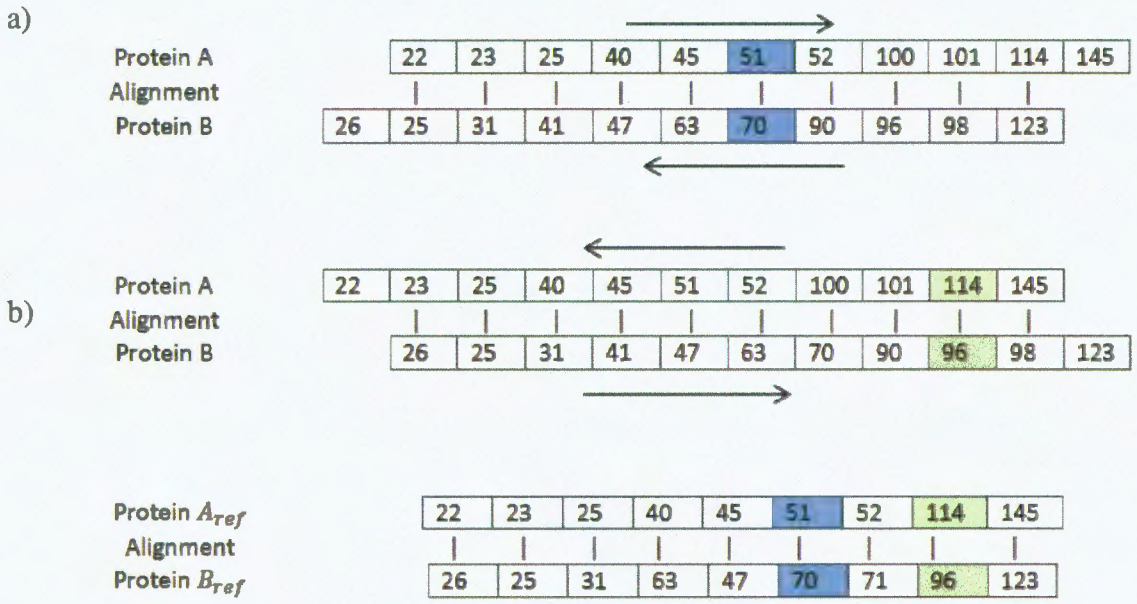

Figure 11. An example of shift-1 analysis. Shift is allowed in both directions (a), (b). Hence, $I_0/L_{ref} = 5/9$, $\frac{I_1}{L_{ref}} = (5+2)/9$.

FSSP Benchmark

Our second test set consists of 185 pair of proteins selected from the FSSP database (Holm et al., 1992). The protein pairs in this representative set are grouped according to the FSSP structural classification: family, superfamily, and fold, with 55, 68, and 60 pairs in each group, respectively. The FSSP test set is can be downloaded at http://bioinformatics.cs.uni.edu/fssp_183.html.

The *FSSP* benchmark utilizes some commonly used protein structure alignment quality measures. More specifically, the accuracy of the alignments is assessed by: The total number of aligned pairs N_{mat} (12), $cRMS$ score (13), the similarity index SI (14), and the percentage of structural similarity PSI (15).

$$N_{mat} = number\ of\ aligned\ pairs\ between\ a\ and\ b \tag{12}$$

$$cRMS = \sqrt{\frac{1}{N_{mat}} \sum_{r=1}^{N_{mat}} \left\| a_{i_r} - b_{i_r} \right\|^2} \tag{13}$$

where (a_{i_r}, b_{i_r}) are the aligned pairs.

$$SI = \frac{cRMS \times min\{L(a), L(b)\}}{N_{mat}} \tag{14}$$

where $L(a)$and $L(b)$ are the lengths of the proteins a and b respectively.

$$PSI\_d = \frac{MNumPairs(d)}{min\{L(a), L(b)\}} \tag{15}$$

where $MNumPairs(d)$ is the method specific number of aligned amino acids pairs (a_{i_r}, b_{i_r}) such that $\|a_{i_r} - b_{i_r}\| < d$.

CHAPTER 4

RESULTS AND DISCUSSION

Sisyphus Benchmark

Figure 12 shows the accuracy of Structal before and after replacing its own superpositions by the superpositions that optimize the number of pairs of residues that can be fit under 3Å. The blue line traces the performance of the original Structal method for different tolerance shifts. The red line traces the performance of Structal on superpositions computed by the EPSILONOPTIMAL algorithm. The Figures 13 and 14 show corresponding data for the other two methods. Tables 2 and 3 show the performance of TM-align and LOCK2 methods in all tolerance shifts.

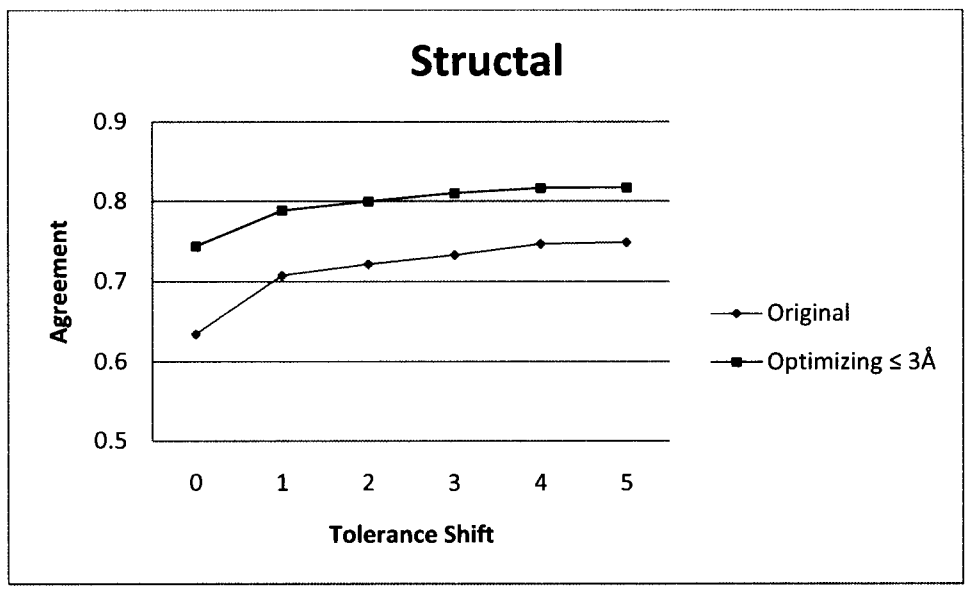

Figure 12. Agreement with the reference alignments obtained by Structal. Blue line shows the accuracy of the original method, whereas the red line shows the accuracy of the same method, but with more accurate superpositions given at input.

Table 1

Structal performance at all tolerance shifts.

	0	1	2	3	4	5
Original	0.63387	0.70728	0.72122	0.73265	0.74649	0.74840
Implemented	0.74368	0.78811	0.79954	0.80997	0.81629	0.81709
Difference (%)	11.0	8.1	7.8	7.7	7.0	6.9

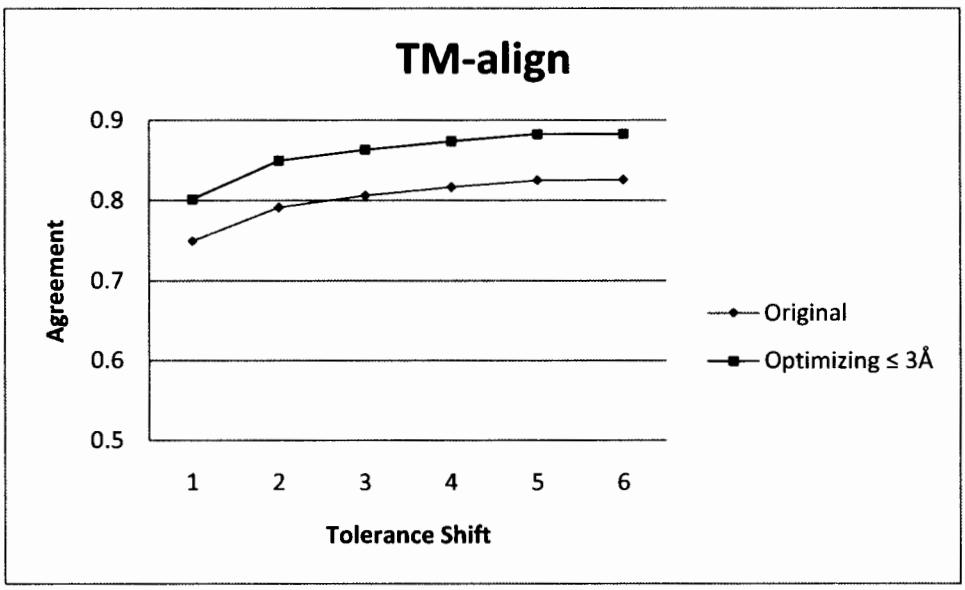

Figure 13. Agreement with the reference alignments obtained by TM-align.

Table 2

TM-align performance at all tolerance shifts.

	0	1	2	3	4	5
Original	0.74960	0.79122	0.80586	0.81629	0.82481	0.82561
Implemented	0.80154	0.84928	0.86292	0.87314	0.88227	0.88247
Difference (%)	5.19	5.81	5.71	5.69	5.75	5.69

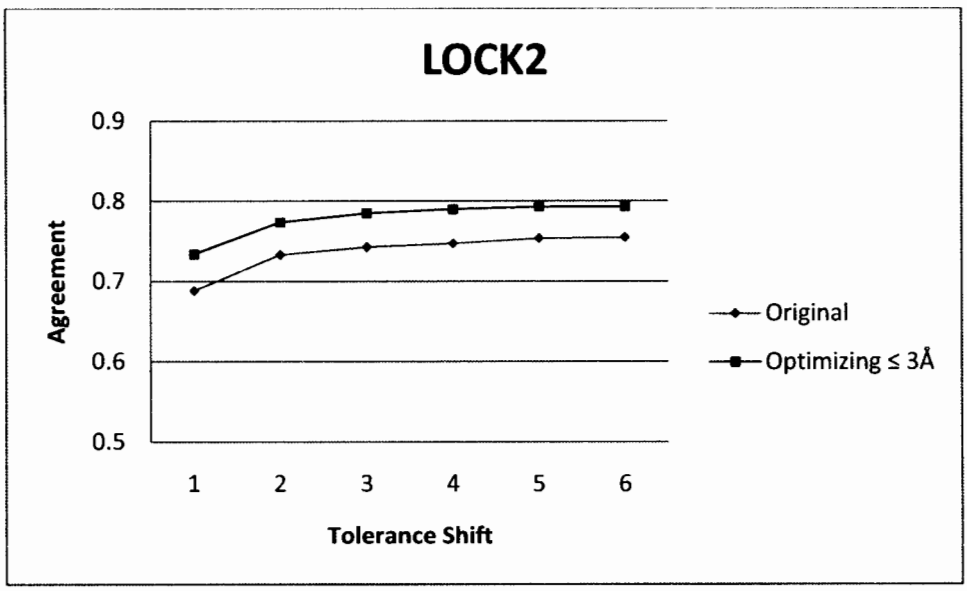

Figure 14. Agreement with the reference alignments obtained by LOCK2.

Table 3

Lock2 performance at all tolerance shifts.

	0	1	2	3	4	5
Original	0.68783	0.73255	0.74208	0.74659	0.75291	0.75421
Implemented	0.73335	0.77316	0.78460	0.78951	0.79262	0.79292
Difference (%)	4.55	4.06	4.25	4.29	3.97	3.87

FSSP Benchmark

The *FSSP* benchmarking results of the three methods in two different settings (original superpositions versus the superpositions optimizing $MNumProt(3)$) are shown in Table 4.

Table 4

FSSP benchmark of three protein structure alignment methods

	Nmat	MNumPairs (3)	MNumPairs (4)	PSI_3	PSI_4	SI	cRMS
Structal							
original	81.37	50.47	56.25	0.59	0.65	7.85	7.22
Optimizing ≤ 3Å	80.99	53.81	57.38	0.63	0.67	7.37	6.76
TM-align							
original	78.52	53.35	59.83	0.62	0.69	5.86	5.16
Optimizing ≤ 3Å	78.07	55.85	60.34	0.65	0.70	5.95	5.19
LOCK2							
original	66.74	51.75	54.16	0.60	0.63	8.35	5.35
Optimizing ≤ 3Å	69.15	58.46	59.38	0.68	0.69	5.69	4.24

The results in Table 4 further indicate that a "fine-tooth comb" search of the superposition space increases the accuracy of all methods, over a wide set of commonly used metrics of protein alignment quality. The only exception is $cRMS$ and SI scores for TM-align, which are slightly better when the original superpositions are used. $MNumPairs(d)$ and PSI_d scores are significantly better for all methods with new superpositions, in particular when the distance cutoff $d = 3$ is used.

Tables 5, 6 and 7 have been included to visualize the performance of the three proteins structure alignment methods on the three protein groups used in the *FSSP* benchmarking.

Table 5

Structal performance on the three protein groups.

		Nmat	cRMS	CA<4	CA<3	SI	PSI_4	PSI_3
Family	Original	96.91	4.46	80.55	74.60	4.76	0.79	0.74
	Optimizing ≤ 3Å	96.09	4.29	80.73	77.11	4.62	0.79	0.76
Superfamily	Original	80.94	7.94	53.90	47.71	8.41	0.64	0.58
	Optimizing ≤ 3Å	79.54	7.04	54.06	50.09	7.61	0.65	0.61
Fold	Original	67.62	8.95	36.63	31.48	10.07	0.54	0.47
	Optimizing ≤ 3Å	68.78	8.71	39.75	36.68	9.63	0.57	0.54

Table 6

TM-align performance on the three protein groups.

		Nmat	cRMS	CA<4	CA<3	SI	PSI_4	PSI_3
Family	Original	94.75	3.32	81.05	74.71	3.65	0.79	0.73
	Optimizing ≤ 3Å	94.69	3.45	81.65	77.49	3.79	0.80	0.76
Superfamily	Original	77.06	5.45	58.44	51.40	6.11	0.70	0.62
	Optimizing ≤ 3Å	76.35	5.35	57.79	52.71	6.10	0.69	0.64
Fold	Original	65.32	6.52	41.95	35.98	7.60	0.60	0.52
	Optimizing ≤ 3Å	64.78	6.61	43.70	39.58	7.76	0.62	0.57

Table 7

LOCK2 performance on the three protein groups.

		Nmat	cRMS	CA<4	CA<3	SI	PSI_4	PSI_3
Family	Original	87.04	3.30	76.75	74.09	3.98	0.76	0.73
	Optimizing ≤ 3Å	89.00	3.23	80.82	79.73	3.80	0.79	0.78
Superfamily	Original	64.28	5.28	51.24	48.63	8.17	0.62	0.59
	Optimizing ≤ 3Å	66.82	4.39	56.32	55.37	5.85	0.67	0.67
Fold	Original	50.92	7.31	36.77	34.82	12.56	0.52	0.50
	Optimizing ≤ 3Å	53.58	4.98	43.20	42.47	7.25	0.61	0.61

Figures 15-17 show several examples of the difference in quality between superpositions generated by existing alignment methods and those computed through a deeper search of the superposition space.

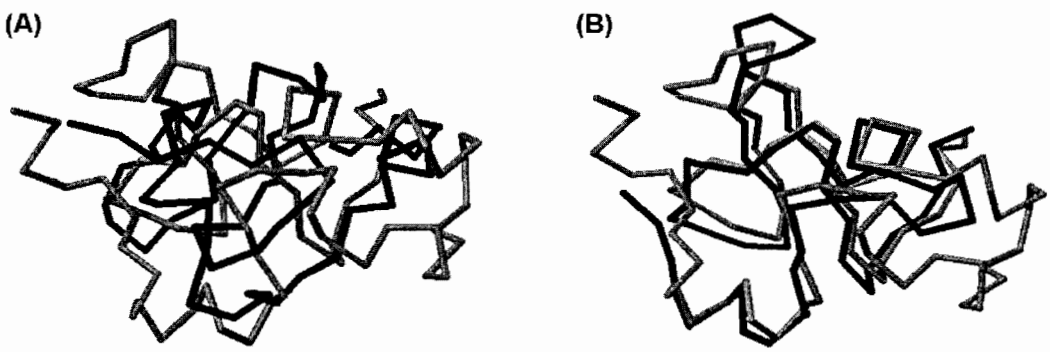

Figure 15. Structural alignments of two high-potential iron-sulfur proteins. The purple phototrophic bacterium Rhodocyclus Tenuis (black; PDB ID: 1isuA) and Chromatium Vinosum (gray; PDB ID: 1ckuA). The alignment in (A) is generated by an LOCK2. The alignment in (B) is obtained by optimizing the number of pairs of residues that can be fit under 3Å.

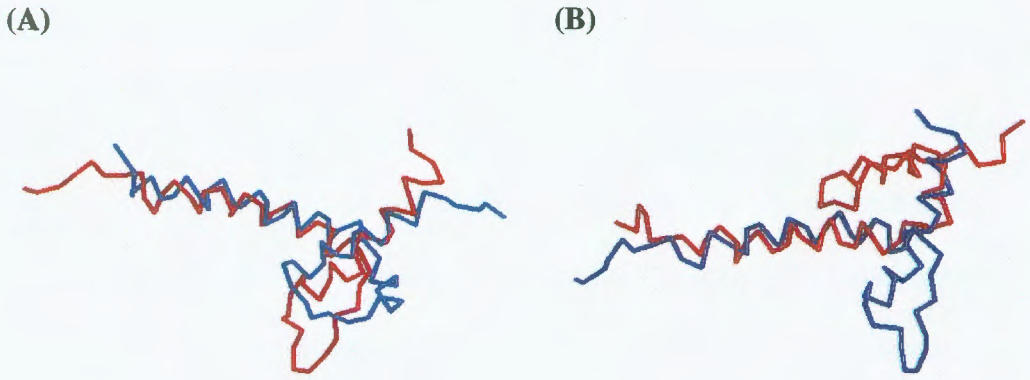

Figure 16. LOCK2 proteins superpositions. Structural superposition of 1an4_A and 1a0a_A by EPSILONOPTIMAL (A) and LOCK2 (B).

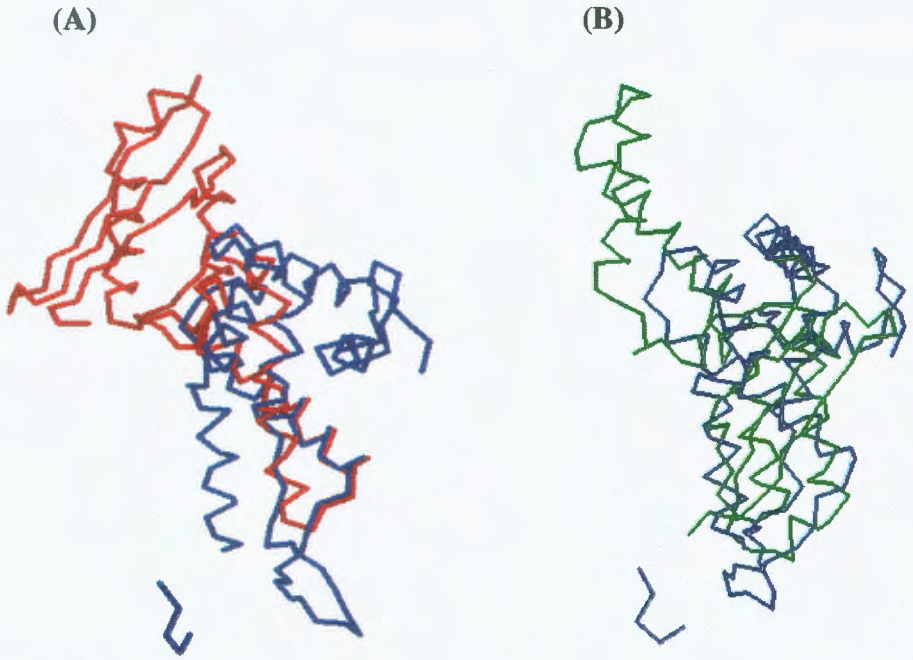

Figure 17. Structal proteins superpositions. The superpositions of 1rss_ and 1tfe_ generated by EPSILONOPTIMAL (A) and Structal (B).

CHAPTER 5

CONCLUSIONS

Protein structure alignment problem is notoriously complex, because of infinite number of ways to position two protein structures in the three-dimensional space. Because of this difficulty, the advances in the field are mostly made through the design of better objective functions. In order to produce reasonable solutions in a timely manner, current methods must trade accuracy for speed and explore only a finite, but representative, set of protein structural superpositions. Hence, the gap between the accuracy of the heuristic alignments generated by current methods and optimal alignment exists. However, to the best of our knowledge, the size of this gap has never been estimated or published.

The present study utilizes a recently developed algorithm for maximizing the number of residues in the input proteins that can be superimposed under a user-specified distance cutoff. Although slow, this algorithm, called EPSILONOPTIMAL, is able to approximate optimal alignments with any given accuracy. Hence, EPSILONOPTIMAL can be utilized to estimate the quality of heuristic alignments.

This study shows significant increase in the quality of the alignments generated by three widely used protein structure alignment methods, obtained through a more detailed search of the superposition space. The accuracy of the new alignments compares favorably with the accuracy of the original alignments in both of our benchmarks. In the *Sisyphus* benchmark, for instance, the new superpositions increase the accuracy of the alignments for Structal, TM-align and LOCK2 by 11%, 7% and 6% respectively.

Improvements of similar magnitude were also seen across different alignment quality measures in the *FSSP* benchmark.

The results of our study indicate the possibilities for further advances in the field and suggest that this area of research will remain attractive in the years to come.

REFERENCES

Andreeva, A., Prlić, A., Hubbard, T., & Murzin, A. (2007). SISYPHUS—structural alignments for proteins with non-trivial relationships. *Nucleic Acids Research , 35*, D253-D259. doi: 10.1093/nar/gkl746

Berman, H. (2008). The Protein Data Bank: a historical perspective. *Acta Cryst. , A64*, 88–95. doi:10.1107/S0108767307035623

Berman, H., Westbrook, J., Feng, Z., Gilliland, G., Bhat, T., Weissig, H., et al. (2000). The Protein Data Bank. *Nucleic Acid Research , 28* (1), 235-242.

Christiansen, T., & Torkington, N. (2003). *Perl Cookbook, Second Edition.* Sebastopol, United States: O'Reilly Media.

Gerstein, M., & Levitt, M. (1996). Using iterative dynamic programming to obtain accurate pairwise and multiple alignments of protein structures. *Proc Inc Conf Intell Syst Mol Biol, 4*, 59-67.

Gök, M., & Yılmaz, Ç. (2006). Hardware Designs for Local Alignment of Protein Sequences. *Computer and Information Sciences – ISCIS , 4263*, 277-285.

Goldsmith-Fischman, S., & Honig, B. (2003). Structural genomics: Computational methods for structure analysis. *Protein Science , 12*, 1813–1821. doi:10.1110/ps.0242903

Gotoh, O. (1982). An improved algorithm for matching biological sequences. *Journal of Molecular Biology , 162* (3), 705-708.

Griffiths, A., Wessler, S., Lewontin, R., & Carroll, S. (2008). Protein Structure. In J. Correa, S. Moran, A. Peltier, L. Lovier, T. Szczepanski, & M. L. Byrd (Eds.), *Introduction to Genetic Analysis* (9th ed., pp. 319-349). New York: W. H. Freeman and Company.

Grüne, L., & Semmler, W. (2004). Using dynamic programming with adaptive grid scheme for optimal control problems in economics. *Journal of Economic Dynamics and Control , 28* (12), 2427-2456. doi:10.1016/j.jedc.2003.11.002

Henikoff, S., & Henikoff, J. (1992). Amino acid substitution matrices from protein blocks. *Proc Natl Acad Sci U S A , 89* (22), 10915–10919.

Hochreiter, S. (2008). Retrieved March 12, 2011, from Institute of Bioinformatics, Johannes Kepler University Linz: http://www.master-bioinformatik.at/curriculum/BioInf_I_Notes.pdf

Holm, L., Ouzounis, C., Sander, C., Tuparev, G., & Vriend, G. (1992). A database of protein structure families with common folding motifs. *Protein Sci. , 1* (12), 1691–1698. doi:10.1002/pro.5560011217

Horn, B. (1987). Closed-form solution of absolute orientation using unit quaternions. *J. Opt. Soc. Am. , 4* (4), 629-642.

Horn, B., Hilden, H., & Negahdaripour, S. (1988). Closed-form solution of absolute orientation using orthonormal matrices. *J. Opt. Soc. Am. , 5* (7), 1127-1135.

Kabsch, W. (1976). A solution for the best rotation to relate two sets of vectors. *Acta Crystallographica , 32* (5), 922–923.

Kolodny, R., Koehl, P., & Levitt, M. (2005). Comprehensive Evaluation of Protein Structure Alignment Methods: Scoring by Geometric Measures. *J. Mol. Biol. , 346*, 1173–1188. doi:10.1016/j.jmb.2004.12.032

Kolodny, R., & Linial, N. (2004). Approximate protein structural alignment in polynomial time. *Proc Natl Acad Sci U S A , 101* (33), 12201–12206. doi:10.1073/pnas.0404383101

Levitt, M., & Gerstein, M. (1998). A unified statistical framework for sequence comparison and structure comparison. *Proc. Natl. Acad. Sci. USA , 95* (11), 5913–5920.

Murzin, A., Brenner, S., Hubbard, T., & Chothia, C. (1995). SCOP: A Structural Classification of Proteins Database for the Investigation of Sequences and Structures. *J. Mol. Biol. , 247* (4), 536–540.

Needleman, S. B., & Wunsch, C. D. (1970). A general method applicable to the search for similarities in the amino acis sequence of two proteins. *J. Mol. Biol. , 48*, 443-453.

Orengo, C., Michie, A., Jones, S., Jones, D., Swindells, M., & Thornton, J. (1997). CATH — a hierarchic classification of protein domain structures. *Structure , 5* (8), 1093–1108.

Pandit, S., & Skolnick, J. (2008). Fr-TM-align: a new protein structural alignment method based on fragment alignments and the TM-score. *BMC Bioinformatics , 9* (1), 531. doi:10.1186/1471-2105-9-531

Pérez, L., Bossio, G., Moitre, D., & García, G. (2006). Optimization of power management in an hybrid electric vehicle using dynamic programming. *Mathematics and Computers in Simulation , 73* (1-4), 244-254. doi:10.1016/j.matcom.2006.06.016

Poleksic, A. (2009). Algorithms for optimal protein structure alignment. *Bioinformatics , 25*, 2751-2756. doi:10.1093/bioinformatics/btp530.

Rocha, J., Segura, J., Wilson, R., & Dasgupta, S. (2009). Flexible structural protein alignment by a sequence of local transformations. *Structural bioinfomatics , 25* (13), 1625–1631. doi:10.1093/bioinformatics/btp296

Sallaberger, C., & D'eleuterio , G. (1995). Optimal robotic path planning using dynamic programming and randomization. *Acta Astronautica , 35* (2-3), 143-156.

Shapiro, J., & Brutlag, D. (2004a). FoldMiner: Structural motif discovery using an improved superposition algorithm. *Protein Science , 13*, 278-294. doi:10.1110/ps.03239404

Shapiro, J., & Brutlag, D. (2004b). FoldMiner and LOCK 2: protein structure comparison and motif discovery on the web. *Nucleic Acids Research , 32*, 536-541. doi:10.1093/nar/gkh389

Singh, A. P., & Brutlag, D. L. (1997). Hierarchical Protein Structure Superposition using both Secondary Structure and Atomic Representations. *Proc Int Conf Intell Syst Mol Biol , 5*, 284-293.

Smith, T. F., & Waterman, M. S. (1981). Identification of Common Molecular Subsequences. *Journal of Molecular Biology , 147* (1), 195-197.

Smyth, M., & Martin, J. (2000). x Ray crystallography. *Mol Path , 53* (1), 8-14. doi:10.1136/mp.53.1.8

Subbiah, S., Laurents, V. D., & Levitt, M. (1993). Structural similarity of DNA-binding domains of bacteriophage repressors and the globin core. *Current Biology , 3* (3), 141-148.

Szustakowski, J., & Weng, Z. (2000). Protein Structure Alignment Using a Genetic Algorithm. *PROTEINS: Structure, Function, and Genetics , 38* (428–440).

Wüthrich, K. (1990). Protein structure determination in solution by NMR spectroscopy. *The Journal of Biological Chemistry , 265* (36), 22059-22062.

Zhang, Y., & Skolnick, J. (2004). Scoring Function for Automated Assessment of Protein Structure Template Quality. *PROTEINS: Structure, Function, and Bioinformatics, 57*, 702-7010.

Zhang, Y., & Skolnick, J. (2005). TM-align: a protein structure alignment algorithm based on the TM-score. *Nucleic Acids Research , 33* (7), 2302-2309. doi:10.1093/nar/gki524