

A Novel Approach To Structural Comparison of Proteins

by
Shantanu Sharma

under the guidance of
Dr. Somenath Biswas

*A Report Submitted in Partial Fulfillment for the Degree of
Bachelor of Technology*

Department of Computer Science and Engineering
Indian Institute of Technology, Kanpur

INDIA
April, 2004

A Novel Approach To Structural Comparison of Proteins

Shantanu Sharma

Submitted for the degree of Bachelor of Technology
April, 2004

Abstract

With the rapid discovery of protein structures, structural comparison of proteins has become a central task in bioinformatics research. Identifying structural similarities can provide significant insights into the relation between structure and function of proteins. Reliable and efficient structural matching plays a key role in computer-aided rational drug design and in assessing the quality of structure prediction methods. However no single structural comparison technique has proven to be efficient and robust over a range of application. In this project, we design and implement an efficient mechanism for structural comparison of proteins: utilizing as much bio-physical information of proteins as possible.

Declaration

The work in this dissertation is based on research carried out at the Department of Computer Science and Engineering, Indian Institute of Technology, Kanpur. No part of this thesis has been submitted elsewhere for any other degree or qualification and the whole of it is my own work unless referenced to the contrary in the text.

Acknowledgements

I would like to express my sincere gratitude to Prof. Somenath Biswas, my project supervisor, for his constant support, inspiration and continuous encouragement during my endeavors. The success of this project is largely due to the invaluable suggestions made by him during project discussions.

I am grateful for his support that he extended when I wanted to try some of the most challenging and novel things during my research. I must also express my gratitude for Prof. R. Sankararamakrishnan, Dr. Jeffrey Roach and Dr. Charles Carter, Professor of Biochemistry at University of North Carolina - Chapel Hill who helped me through their vast experience, especially when I was lost in the enormous world of *Protein Structures*.

I thank all my friends, especially Gaurav, Santosh and Nimit without whom my work would never have been as progressive as it was. I would like to dedicate this work of mine to the grace of the almighty god, and my parents because of whom I am capable of doing this work.

Contents

Abstract	ii
Declaration	iii
Acknowledgements	iv
1 Overview of Protein Structure Comparison	1
1.1 Introduction	1
1.2 Existing Approaches	1
1.2.1 DALI: Distance Alignment of Residues	1
1.2.2 LGA: Local-Global Alignment	2
1.2.3 SCOP: Structural Classification of Proteins	2
1.3 Motivation for a Novel Approach	2
2 An Overview of Our Approach	4
2.1 Algorithm of Our Approach	4
2.2 Brief Description	4
2.3 Using Forward and Backward Edges	5
2.4 Compressed structural representation of protein	5
3 Computing the Structural Representation of Proteins	7
3.1 An Introduction to Delaunay Tessellation	7
3.2 Empty Circle Property	8
3.3 Algorithm for computing the structural representation	8
4 Signatures of Secondary Structure Elements	10
4.1 Introduction	10
4.2 Identifying α helices	10
4.3 Identifying β strands	11
4.4 Identifying turns	12
5 Pairwise Comparison of Proteins	14
5.1 Introduction	14
5.2 Residue by Residue Comparison	14
5.3 Algorithm for computing similarity of two residues	14
5.4 Scoring Function	15
5.5 Pairwise comparison of proteins	15

6	Concluding Remarks	16
6.1	Distance Matrix and Clusters	16
6.2	Conclusion	17

List of Figures

2.1	A portion of compressed structural representation of the protein Cytidine Deaminase: 1JTK	6
3.1	Illustrating Delaunay Tessellation over C- α backbone	7
3.2	Illustrating Voronoi Diagram and Delaunay Tessellation as its dual	8
4.1	Signature of alpha-helices	10
4.2	Signature of beta-strands	11
4.3	Signature of turns	12
6.1	Illustrative Alignment of Cytidine Deaminase, Ferridoxins and Hipips	16

Chapter 1

Overview of Protein Structure Comparison

1.1 Introduction

Proteins fold into beautiful and sophisticated three-dimensional structures. Recently, tertiary structures of hundreds of proteins have been solved using X-Ray Crystallography and 2D-Nuclear Magnetic Resonance (NMR) experiments, and the number is growing at a rapid pace: as many as 25115 structures (As on 13th April 2004) of proteins are already registered at the Brookhaven Protein Data Bank [1].

In literature, description of protein structures is done at various levels of abstraction: from atomic-coordinates, through secondary structures to tertiary structures. Analyzing the relationship between structure and function of a protein is of fundamental importance in rational drug design. Even when protein-sequences (arrays of amino-acids constituting the protein) of two proteins are markedly different, their 3D-structures and functionality may be surprisingly similar. A structural classification of proteins facilitates the understanding of how structural similarities effect functionality in proteins and correlates with the amino-acid sequence of the protein. Existing popular approaches for structural classifications of proteins include SCOP [2], FSSP [3] and LGA [4]. These are briefly described in the following section.

1.2 Existing Approaches

1.2.1 DALI: Distance Alignment of Residues

In DALI, [3] three-dimensional coordinates of each protein are used to calculate residue-residue ($C-\alpha - C-\alpha$) distance matrices. The distance matrices are first decomposed into elementary contact patterns, e.g., hexapeptide-hexapeptide submatrices. Then, similar contact patterns in the two matrices are paired and combined into larger consistent sets of pairs. A Monte Carlo procedure is used to optimize a similarity score defined in terms of equivalent intramolecular distances. The method allows sequence gaps of any length, reversal of chain direction, and free topological connectivity of aligned segments. Sequential connectivity can be imposed as an option. The method identifies structural resemblances and common struc-

tural cores accurately and sensitively, even in the presence of geometrical distortions.

1.2.2 LGA: Local-Global Alignment

LGA, [4] is used to facilitate the comparison of protein structures or fragments of protein structures in sequence dependent and sequence independent modes. The LGA structure alignment program is available as an online service at:

<http://predictioncenter.llnl.gov/local/lga>

Data generated by LGA can be successfully used in a scoring function to rank the level of similarity between two structures and to allow structure classification when many proteins are being analyzed. LGA also allows the clustering of similar fragments of protein structures.

1.2.3 SCOP: Structural Classification of Proteins

The SCOP database [2] provides a description of the relationships of all known protein structures. The classification is on hierarchical levels: the first two levels, family and superfamily, describe near and far evolutionary relationships; the third, fold, describes geometrical relationships. The distinction between evolutionary relationships and those that arise from the physics and chemistry of proteins is a feature that is unique to this database, so far. The database can be used as a source of data to calibrate sequence search algorithms and for the generation of population statistics on protein structures. The database is accessible from URL <http://scop.mrc-lmb.cam.ac.uk/scop/>.

1.3 Motivation for a Novel Approach

As described above, various methods for structural comparison of proteins have been developed, such as: DALI [3], SCOP [2] and LGA [4]. However, current strategies rely far too heavily in choosing pairs of atoms (one from each structure) to compare. One approach is to pairwise align protein structures and a measure of statistical deviation, e.g. a variance, between spatial locations of corresponding atoms is computed. The problem with this approach is that it differs drastically based on which atoms are compared and how structures are oriented in space.

The second approach, also unsatisfactory, is to compute internal distances between atoms within each protein and then compare the distance matrices of each protein. This method also suffers because structurally insignificant differences affect the distances far too much. For example, increasing the length of a loop in one of the structures will change the comparison too much.

Our approach is to come up with a novel structural description of proteins that carries as much biochemical information as possible and as little non-chemical information as possible.

A necessary prerequisite for such a description is that it should be computationally efficient to generate. We intend to use this description to obtain a measure of similarity between structures from the biological point of view. Since proteins are flexible macromolecules, the comparison technique should also entail possibilities of local deviations between pairs of proteins.

Chapter 2

An Overview of Our Approach

2.1 Algorithm of Our Approach

Our approach for structural classification of proteins comprises of the following steps:

1. Generating the novel structural representation of proteins from PDB: classifying edges into very-close, close, far and very-far.
2. Exploring signatures of secondary structure elements in the new representation and store the residue-type with each residue.
3. Computing the longest common subsequence between two proteins, using a residue-by-residue similarity score based on secondary structure signature and tessellation edges to preceding and succeeding residues..
4. Computing the pairwise distance between the given two proteins by assigning gap initiation and gap continuation penalties along the aligned protein-backbones.
5. Generating a database of clusters representing structurally similar proteins based on the pairwise-distance among them and choosing a representative member of each cluster.
6. Given an unclassified protein, it is classified by computing the distances with the representative members of the clusters.

2.2 Brief Description

Using the QHull program [7], Delaunay tetrahedralization of coordinates of C- α backbone atoms is obtained. PDB entries of proteins are used to obtain coordinates of backbone-atoms. Then residue-based structural representation of proteins is generated: Going down the chain and storing edges of Delaunay tetrahedralization connecting a residue, say residue number 'n' to its preceding and succeeding residues in the chain: 1 to (n-1). Using the PDB files, the edges in tessellation are classified into very-close, close, far and very-far categories. A study of such tessellation edges reveals that residues constituting helices, sheets and turns had peculiar signatures as described in Chapter 4.

This property is used to annotate residues as belonging to helices, sheets and turns. A scoring function is then constructed to compare two such representations of proteins and finding the gapped alignment between them. Separate penalties for gap-initiation and gap-continuation are assigned. Distance among chains belonging to Cytidine Deaminase, Ferridoxins, Hipips, Histones family of proteins was computed and the distance-matrix was plotted. The distance-matrix displays convincing results with chains belonging to same family of proteins being clustered in groups distant from chains belonging to other protein-families.

2.3 Using Forward and Backward Edges

Forward and backward edges of Delaunay Tessellation strings are stored in the new representation of protein structure. The scoring function is also modified to use both forward and backward edges for protein structure comparison. A significant improvement is observed while simultaneously using forward and backward edges for comparison since the tessellation strings having smaller forward edges correspond to larger backward edges (thereby providing larger data for comparison) and vice versa.

2.4 Compressed structural representation of protein

In the compressed structural representation:

- 0 denotes start of a residue.
- Codes 'S', 'H' or 'N' indicate that this residue is part of a helix, sheet or turn respectively.
- Tessellation Edges of each residue follows after the '0'. Edges are indicated by difference between residue numbers of the current residue and the residue forming the other vertex.
- Codes 'A', 'B', 'C', 'D' indicate that the other residue forming the tessellation edge with the current residue is very-close/close/far/very-far respectively.

The following chapters will explain the various steps involved in greater detail.

0S 36A 31A 28A 25B 21C 8A 7A 6A 5A 2A 0S 26B 22B 9A 8A 7A 6A 2A
0S 30A 29A 27A 26A 23A 11A 10A 2A 0S 27A 24A 13A 12A 11A 0S 28A
27A 25A 24A 15A 14A 13A 0S 25A 24A 23A 20B 18A 17A 16A 0S 27A
26A 25A 24A 3A 2A 0S 26A 25A 24A 23A 22A 2A 0S 27A 25A 24A 2A
0S 28B 26A 25A 0H 27A 26A 4A 3A 2A 0H 2A 0N 7A 5A 3A 2A 0S 29B
9A 8A 7A 6A 3A 2A 0N 10A 9A 3A 2A 0N 31B 29B 27A 11A 2A 0N 28A
27A 26A 14A 13A 12A 0H 27A 25A 15A 2A 0H 4A 3A 2A 0H 41B 17A
16A 15A 14A 5A 4A 3A 2A 0H 30A 29A 28A 20A 19A 18A 4A 3A 2A 0H
29A 4A 3A 2A 0H 17B 10C 8B 4A 3A 2A 0H 45B 43C 23A 22A 21A 18B
11C 4A 3A 2A 0H 32A 30A 26A 24A 4A 3A 2A 0H 4A 3A 2A 0H 46D 21C
19D 18D 17D 14C 4A 3A 2A 0H 49C 47D 31B 29B 27B 26B 20D 19D 4A
3A 2A 0H 32B 30B 3A 2A 0N 35A 33A 31A 5A 4A 2A 0N 36A 9A 6A 5A
2A 0N 37A 36A 35A 3A 2A 0S 40A 39A 38A 37A 11A 8A 2A 0S 70B 69B
40A 39A 38A 36A 2A 0S 71A 70A 69A 66A 42A 41A 2A 0S 67A 45A 44A
43A 42A 18A 14A 3A 0N 71A 68A 67A 64A 46A 45A 2A 0N 68A 65A
49A 48A 47A 21A 20A 2A 0N 69A 66A 65A 62A 50A 49A 0N 66A 63A
53A 52A 51A 50A 0S 67B 64A 63B 55B 54A 53A

Figure 2.1: A portion of compressed structural representation of the protein Cytidine Deaminase: 1JTK

Chapter 3

Computing the Structural Representation of Proteins

3.1 An Introduction to Delaunay Tesselation

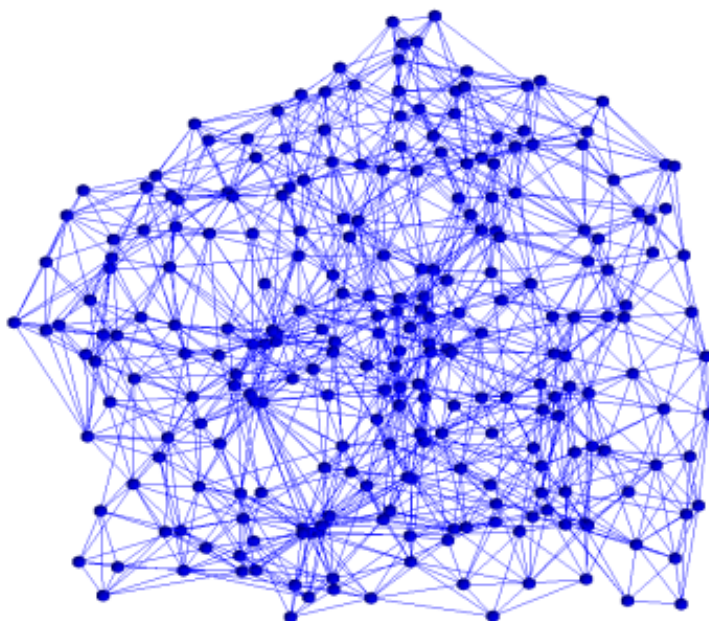


Figure 3.1: Illustrating Delaunay Tesselation over C- α backbone

A tessellation is defined as a division of a space into convex polygonal regions. Given a point process with points $\{p_i\}$, a tessellation can be formed by joining all neighboring points; by neighboring meaning pairs of points whose cells in the Voronoi diagram share an edge. The tessellation resulting from this construction is the Delaunay tessellation. It is also defined as a triangulation S of V (Set of points) such that the circum-circle of any triangle belonging to S does not contain points of V in its interior.

3.2 Empty Circle Property

Thus, the Delaunay triangulation of a set V of points is unique provided that no four or more points of V are not co-circular. This property is also known as the *local empty-circle property*: e satisfies the local empty circle property iff the circum-circle of any of the two triangles sharing edge does not contain the vertex of the other triangle in its interior.

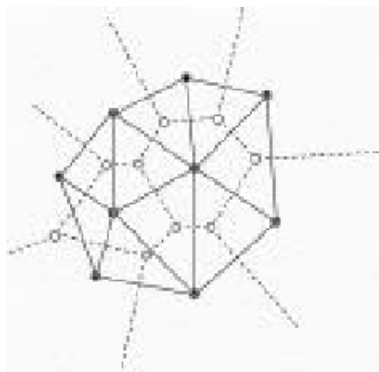


Figure 3.2: Illustrating Voronoi Diagram and Delaunay Tessellation as its dual

The Delaunay triangulation $S(V)$ and the Voronoi diagram $\text{Vor}(V)$ of a set of points are dual as plane graphs:

1. Every point p of V corresponds to a Voronoi region $\text{RV}(p)$
2. Every triangle of $S(V)$ correspond to a vertex in $\text{Vor}(V)$
3. Every edge $e=(p,q)$ in $S(V)$ corresponds to an edge shared by the two Voronoi regions $\text{RV}(p)$ and $\text{RV}(q)$

3.3 Algorithm for computing the structural representation

The algorithms for computing the novel structural representation of protein, using their PDB entries is illustrated as follows:

Data : Input PDB Files

Result: New structural representation of given protein

Extract coordinates of C- α backbone atoms from PDB File;

Using QHull, perform Delaunay tessellation over backbone coordinates;

Set current residue as first residue;

Read forward edges and backward edges for first residue;

while *not at the last residue* **do**

for *All succeeding residues having tessellation edge with current residue* **do**

 Compute euclidean-distance with the current residue;

if *euclidean-distance* $< 5 \text{ \AA}$ **then**

 | Tag forward-edge as very close;

end

if *euclidean-distance* $\in \{5 \text{ \AA}, 10 \text{ \AA}\}$ **then**

 | Tag forward-edge as close;

end

if *euclidean-distance* $\in \{10 \text{ \AA}, 15 \text{ \AA}\}$ **then**

 | Tag forward-edge as far;

end

if *euclidean-distance* $> 15 \text{ \AA}$ **then**

 | Tag forward-edge as very far;

end

 Store forward-edges corresponding to this residue;

end

 Set current residue as next residue;

end

Repeat the above procedure for backward-edges;

Algorithm 1: Generating the novel structural representation

Chapter 4

Signatures of Secondary Structure Elements

4.1 Introduction

The significant utility of performing Delaunay tessellation over protein backbone is the ability of identifying signatures of secondary structure elements in the residues. In the following sections, we describe how signatures of alpha-helices, beta-sheets and turns were identified.

4.2 Identifying α helices

Because of the geometry helical structures, α -helices have Delaunay tessellation edges to preceding three residues. This a set of consecutive residues each having close or very-close tessellation edges form parts of an alpha helix. By storing forward and backward edges, the residue type can be inferred using both kinds of edges.

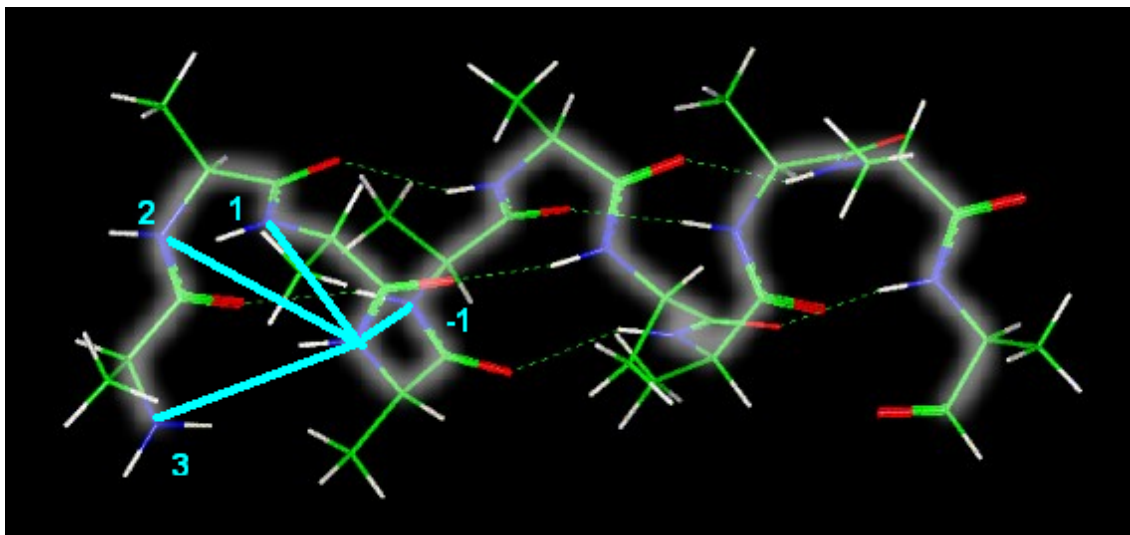


Figure 4.1: Signature of alpha-helices

Data : Edges formed by a residue 'n' in Delaunay tessellation of protein backbone

Result: Yes, if the residue is a helix, No otherwise

```

for each of the preceding three residues do
  if  $n - i^{\text{th}}$  residue ( $i \in \{1, 3\}$ ) have a tessellation edge then
    mark  $n^{\text{th}}$  residue as helix;
    output Yes;
  else
    output No;
  end
end
end

```

Algorithm 2: Identification of helix signatures

4.3 Identifying β strands

Because of the geometry beta strands, a set of consecutive residues have delaunay tessellation edges to another set of consecutive residues. These edges also fall in close or very-close category.

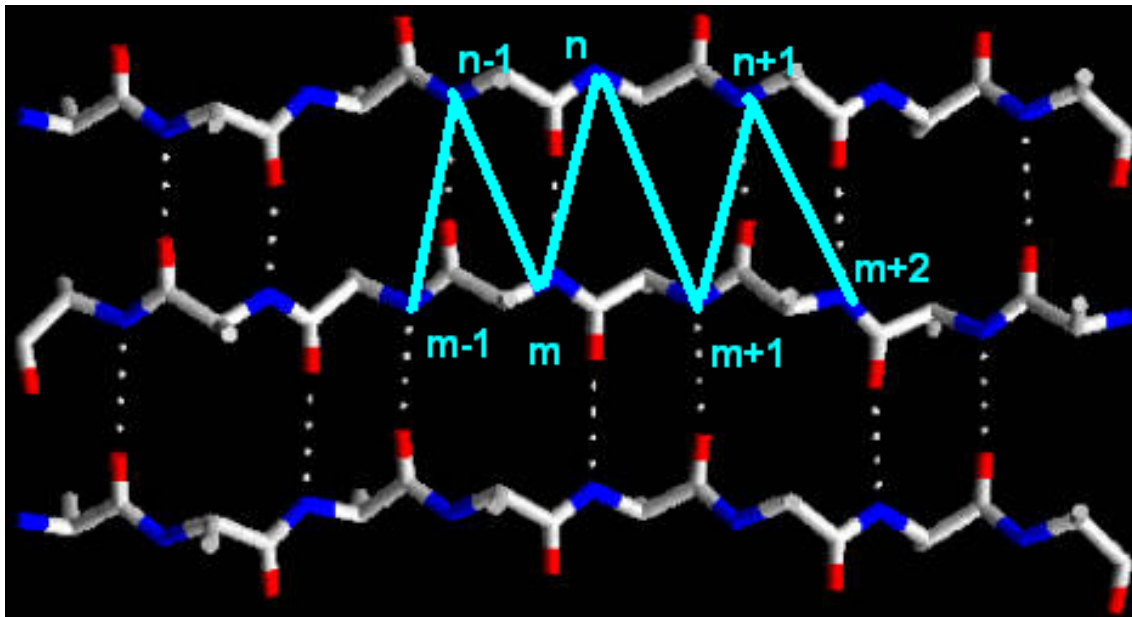


Figure 4.2: Signature of beta-strands

Data : Edges formed by residues 'n-1', 'n', 'n+1' in Delaunay tessellation of protein backbone

Result: Yes, if the residue is a beta-sheet, No otherwise

for residues $n-1, n, n+1$ **do**

if $\exists m$ — residues: $n-1$ has edges to $m-1, m$; n has edges $m, m+1$; $n+1$ has edges $m+1, m+2$ **then**

 mark n^{th} residue as part of beta sheet;

 output Yes;

else

 output No;

end

end

Algorithm 3: Identification of beta-strand signatures

4.4 Identifying turns

Because of the geometry turns, a residue forming a part of a turn has close/very-close edges to consecutive preceding/succeeding residues. By storing forward and backward edges, the residue type can be inferred using both kinds of edges.

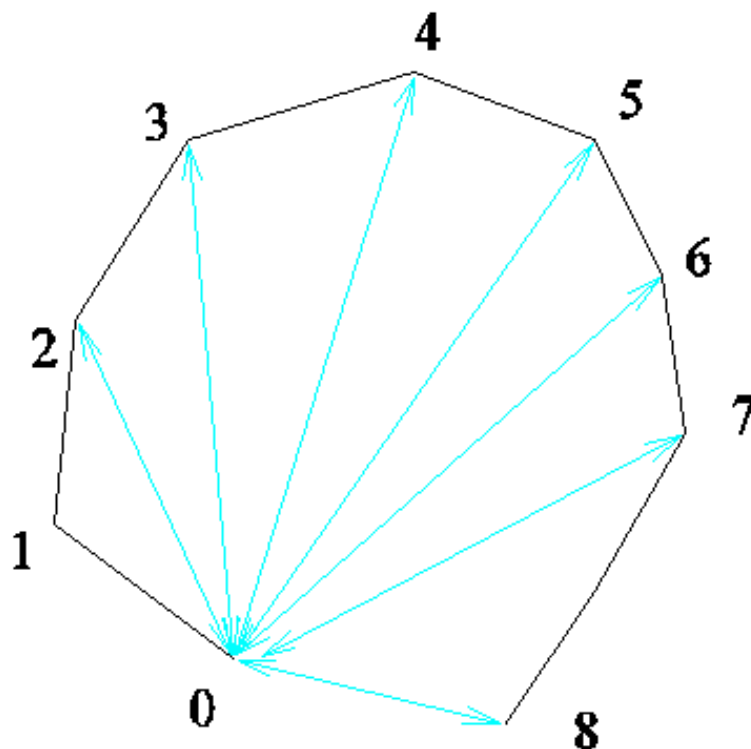


Figure 4.3: Signature of turns

Data : Forward and backward tessellation edges of a residue 'n'

Result: Yes, if the residue is a turn, No otherwise

```
for residues  $n-1, n-2, n-3, n-4$  do
  | if all residues are close/very-close and have tessellation edges with residue 'n' then
  |   | mark  $n^{th}, n-1^{th}, n-2^{th}, n-3^{th}, n-4^{th}$  residue as beta;
  |   | output Yes;
  | else
  |   | output No;
  | end
end
```

Algorithm 4: Identification of turns signatures

Chapter 5

Pairwise Comparison of Proteins

5.1 Introduction

Pairwise comparison is performed at two stages:

- Between residues, used for generating the longest common subsequence.
- Between chains, used for computing the distance between two proteins.

In the following sections, both comparisons are described:

5.2 Residue by Residue Comparison

Residue by residue comparison is performed by comparing two biophysical properties of the residue:

- The secondary structure types of the two residues, and
- The spatial arrangement of other residues forming tessellation edges with this residue

A threshold on similarity score is assigned to compare the residues. The similarity score is based on the differences between tessellation edges for the two residues. Firstly, the corresponding edges of residues are aligned with gaps incurring penalties. Then greater numerical edge-difference between aligned edges are scaled according to closeness with the current residue: very-close edges getting greatest weights. The similarity score is computed based on the aligned tessellation edges.

5.3 Algorithm for computing similarity of two residues

Here, $Edge_i$ represents the difference between residue numbers of current residue and residue having i^{th} tessellation edge, beginning from the closest residue.

Data : Two input residues

Result: Similarity score of the two residues

```

for tesselation edges in the residues:  $i \in \{1, m\}$  of first and  $j \in \{1, n\}$  of second do
  if  $Edge_i - Edge_j \in \{-3, 3\}$  and edge-types are both: close/very-close/far/very-far
  then
    Add absolute-difference between edges to distance between residues Align  $i^{th}$ 
    residue to  $j^{th}$  residue;
  else
    if  $Edge_i - Edge_j \in \{-3, 0\}$  and edge-types are both:
    close/very-close/far/very-far then
      Increment i;
    else
      Increment j;
    end
  end
end

```

Output inverse of Distance as the similarity-score between residues.

Algorithm 5: Computing similarity of two residues

5.4 Scoring Function

A scoring function is used to compute the distance between aligned set of residues in the two proteins. The scoring-function assigns greater weights to residues constituting structurally significant secondary-structure types: alpha-helices and beta-strands. Residues corresponding to mismatched secondary-structure type are given lesser weight. Also, gap initiation and gap continuation penalties are added to give lesser weightage to structural alignments with gaps.

5.5 Pairwise comparison of proteins

Data : Structural representations of two proteins

Result: Structural distance between the two proteins

```

for residues in the two proteins:  $i \in \{1, m\}$  of first and  $j \in \{1, n\}$  of second do
  Compute longest common subsequence of residues using the similarity scores
  between  $i^{th}$  and  $j^{th}$  residues;
  Compute gap initiation and gap continuation penalties for gaps in the aligned set of
  residues;
  Evaluate the distance between the two proteins using the longest common
  subsequence of residues, gap penalties and the scoring function;
end

```

Output the pairwise distance between two proteins.

Algorithm 6: Computing similarity of two proteins

Chapter 6

Concluding Remarks

	CDA	CDA	CDA	CDA	FDX	FDX	FDX	FDX	HPIP	HPIP	HPIP	HPIP
	1JTK_A	1JTK_B	1CTU_A	1CTU_B	2FDN	1DUR	1BLU	1BOT	1BOY	1CKU_A	1CKU_B	1HP
1JTK_A	0	2	184	192	261	257	301	397	352	357	327	344
1JTK_B		0	184	192	261	257	301	399	352	357	327	344
1CTU_A			0	184	268	277	380	470	317	321	321	342
1CTU_B				0	299	289	420	468	383	376	380	371
2FDN					0	54	148	95	213	213	220	206
1DUR						0	178	127	227	216	214	214
1BLU							0	230	270	306	307	272
1BOT								0	323	293	343	331
1BOY									0	34	31	36
1CKU_A										0	33	48
1CKU_B											0	60
1HP												0

Figure 6.1: Illustrative Alignment of Cytiedine Deaminase, Ferridoxins and Hipips

6.1 Distance Matrix and Clusters

Distance matrix of protein structures is generated using the pairwise-distances computed, as explained in the previous chapter. The above distance matrix indicates distances between Cytediene Deaminase (CDA), Ferridoxins (FDX), Hipip (HPIP) and Histone (HIS) family of proteins. Using these clusters, a database of structural classification of proteins is formed. Representative member of each cluster is assigned to that protein whose pairwise distance with other proteins belonging to the cluster has least variance.

The clusters are named according to the key structural motifs conserved among their constituent protein members. Whenever an unclassified protein is observed, pairwise alignments against representative members of each cluster is done and the closest cluster is assigned to the unknown protein.

6.2 Conclusion

In this project we designed and implemented a novel approach for structural classification of proteins. The Protein Data Bank has an enormous set of protein structures. As opposed to SCOP, this design can be used to develop a *fully automated* database of structural classification of proteins. Since clusters formed in the database correspond to structural motifs, this tool can also be used for motif-discovery in an unknown protein. The structural database can be of significant use in generating alignments based on structural similarity. Therefore, it can serve as a pharmaceutical tools in analysing protein-protein docking and in computer-aided rational drug design.

Bibliography

- [1] J. Westbrook, Z. Feng, S. Jain, T. Bhat, N. Thanki, V. Ravichandran, G. Gilliland, W. Bluhm, H. Weissig, D. Greer, P. Bourne, and H. Berman. The protein data bank: unifying the archive, 2002.
- [2] Tim Hubbard Alexey G. Murzin, Steven E. Brenner and Cyrus Chothia. SCOP: A structural classification of proteins database for the investigation of sequences and structures. *Journal of Molecular Biology*, 247:536–540, 1995.
- [3] Liisa Holm and Chris Sander. Protein structure comparison by alignment of distance matrices. *Journal of Molecular Biology*, 233:123–138, 1993.
- [4] A. Zemla. LGA Program: A method for finding 3-D similarities in protein structures. *Calculateurs Paralleles*, 8(2):137–150, 2000.
- [5] D. F. Watson. Computing the n-dimensional Delaunay tessellation with application to Voronoi polytopes. *Computer Journal*, 24(2):167–172, 1981.
- [6] H. Edelsbrunner and E. P. Mucke. Three-dimensional alpha shapes. 13(1):43–72, 1994.
- [7] C. Bradford Barber, David P. Dobkin, and Hannu Huhdanpaa. The quickhull algorithm for convex hulls. *ACM Transactions on Mathematical Software*, 22(4):469–483, 1996.