12-17-2010

Toward a Database of Geometric Interrelationships of Protein Secondary Structure Elements for De Novo Protein Design, Prediction and Analysis

Augustine Ada Orgah
University of New Orleans

Toward a Database of Geometric Interrelationships of Protein Secondary Structure Elements for *De Novo* Protein Design, Prediction and Analysis

A Thesis

Submitted to the Graduate Faculty of the
University of New Orleans
in partial fulfillment of the
Requirements for the degree of

Master of Science
in
Computer Science
Bioinformatics

by

Augustine Ada Orgah

B.S. Xavier University of Louisiana, 2008

December, 2010

Table of Contents

List of Figures

List of Tables

Abstract

Computational methods of analyzing, simulating, and modeling proteins are essential towards understanding protein structure and its interactions. Computational methods are easier as not all protein structures can be determined experimentally due to the inherent difficultly of working with some proteins. In order to predict, design, analyze, simulate or model a protein, data from experimentally determined proteins such as those located in the repository of the Protein Data Bank (PDB) are essential. The assumption here is that we can use pieces of known proteins to piece together a "new" protein hence, *de novo* protein design. The analysis of the geometric relationships between secondary structure elements in proteins can be extremely useful to protein prediction, analysis, and *de novo* design. This thesis project involves creating a database of protein secondary structure elements and geometric information for rapid protein assembly, *de novo* protein design, prediction and analysis.

Keywords: *de novo* protein design, ProtCAD

Chapter 1: Introduction

1.1 About Proteins

Proteins are said to be the building blocks of life. They play an indispensable role in cellular function, growth, health, and disease. Proteins are the key means through which many diseases have their effect [1]. In order to better understand the underlying mechanisms of diseases, or design drugs to treat diseases, an in-depth understanding of protein structures, as well as how they interact with biological systems and drug molecules are paramount. Leading the way towards this goal is the use of computational structural biology.

Computational structural biology plays a major role in understanding protein structures and its interaction with biological systems. Computational structural biology deals with the simulation, modeling and computational methods used to understand protein structures, prediction, function, folding, design, and the dynamics of biological molecules [1]. In order to understand the mechanisms of computational structural biology, a background of protein structure, protein secondary structure elements, protein folding, protein prediction and *de novo* design is necessary.

A simple definition of a protein is that it is a compound made up of a sequence of amino acids. A protein may have more than one chain in its composition. Amino acids are the building blocks of proteins. They are molecules that contain an amine group, a carboxylic acid group and a side chain that varies between the different amino acids. There are 20 standard amino acids and a unique combination and sequence of them make up a protein and determine that protein's function, fold, and activity. The amine group of an amino acid is a compound made up of a nitrogen atom bonded to hydrogen

atoms. The carboxylic acid group is another constituent of the amino acid that consists of at least one carboxyl group. A carboxyl group is a molecule made up of a carbonyl and hydroxyl group. A carboxyl group is molecule made up of a carbon atom double bonded with oxygen atom while the hydroxyl group is a molecule made of a hydrogen atom covalently bonded with an oxygen atom [2]. All amino acids found in proteins have the basic structure in **Figure 1** and only differ in the structure of the side chain or R-group.

The side chain is a group attached to the carbon alpha atom and specifies the chemical and physical properties of each of the 20 amino acids. **Figure 2** shows the structures of the different amino acids and their side chains.

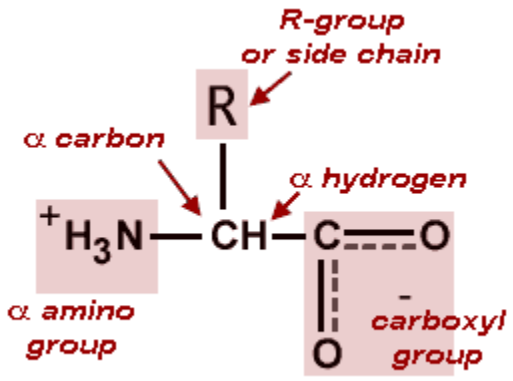

Figure 1. Basic structure of an amino acid. Reprinted from [3].

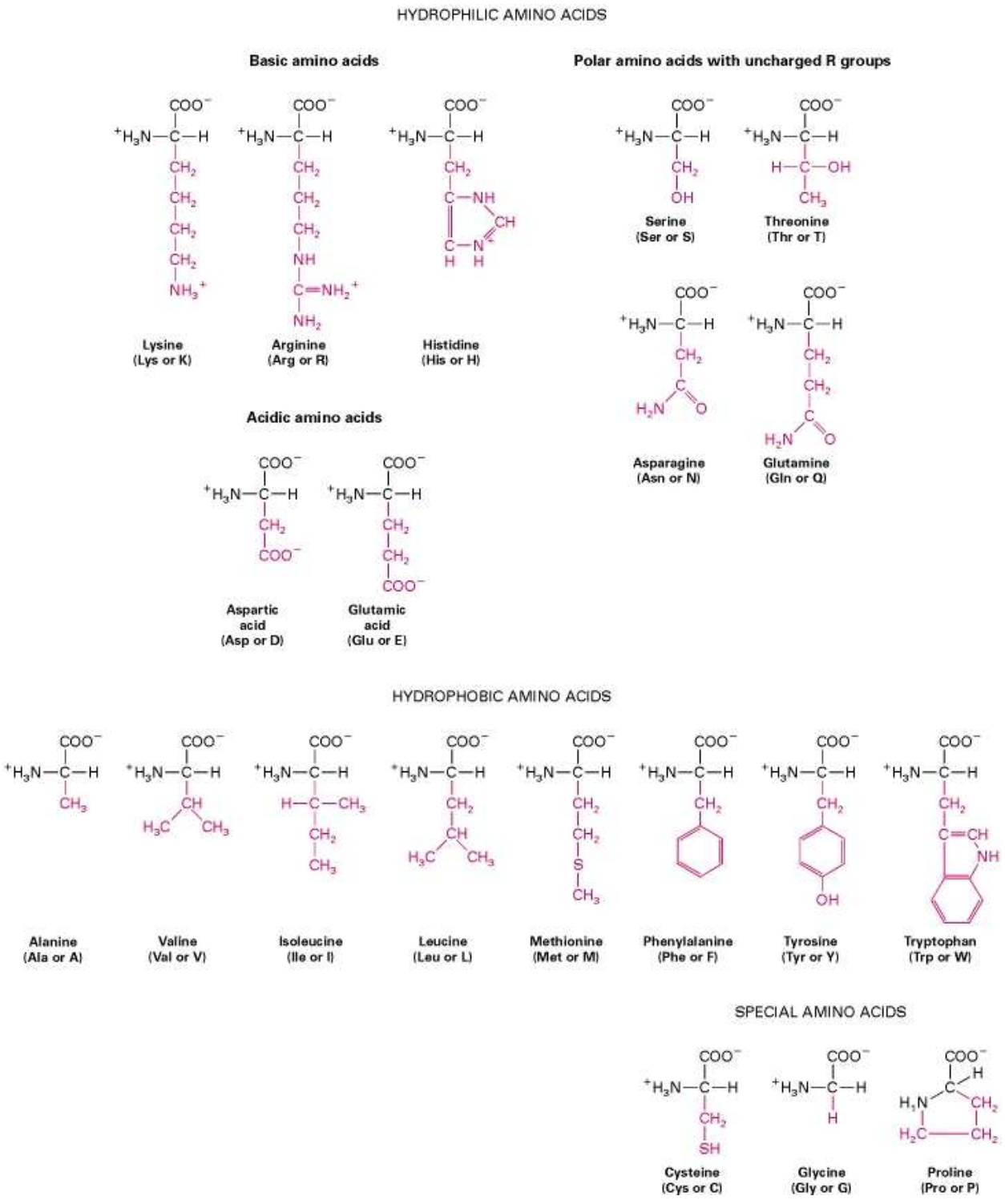

Figure 2. The Structure of the 20 Amino Acids. Reprinted from [4].

1.2 Protein Structure & Hierarchy

Proteins have primary, secondary, tertiary and quaternary structures. In hierarchical order, the primary structure of a protein is the basic or simplest structure with complexity and difficulty increasing towards the quaternary structure.

1.2.1 Protein primary structure

The primary structure of a protein is the linear amino acid sequence that makes up that particular protein. Covalent bonds exist between the amino acids that make up the sequence or chain. Due to how they are linked, the opposite ends of a protein chain usually end up with free groups. On the left, an N-terminus (amino group) and a C-terminus (carboxylic group) on the right end [4].

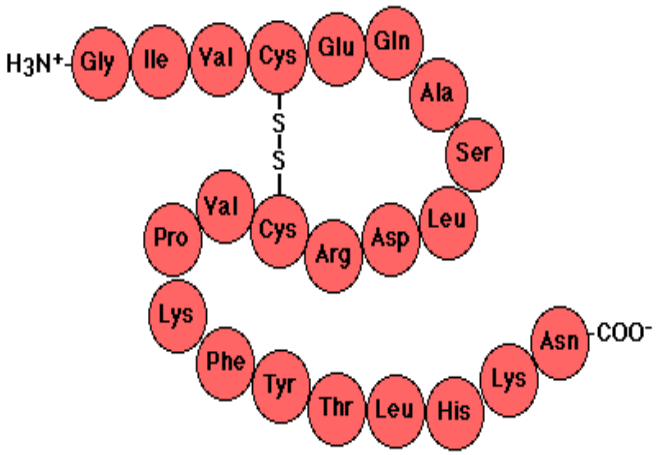

Figure 3. The Primary Structure of a Protein Showing its N and C Termini. Reprinted from [5].

1.2.2 Protein secondary structure

The secondary structure of a protein is most commonly classified into 2 categories: the alpha helix and the beta sheet. Other categories like coils and turns are

present and link these basic elements of secondary structure together. Secondary structures are regions of a protein that form into a helical or beta sheet conformation when stabilizing hydrogen bonds form between certain residues of the amino acids of the protein. If these stabilizing hydrogen bonds are absent, a random-coil structure is assumed by the protein. The helix possesses a spiral structure while the structure of the beta sheet is that of a twisted, pleated plane.

An alpha helix is a spiral looking structure that is formed when the carbonyl oxygen of each amino acid is hydrogen-bonded to the amide hydrogen of the amino acid four residues toward the C-terminus. The phenomenon is as if the primary structure is twisted upon itself for the region(s) assuming a helical conformation. There are about 3.6 amino acid residues per turn.

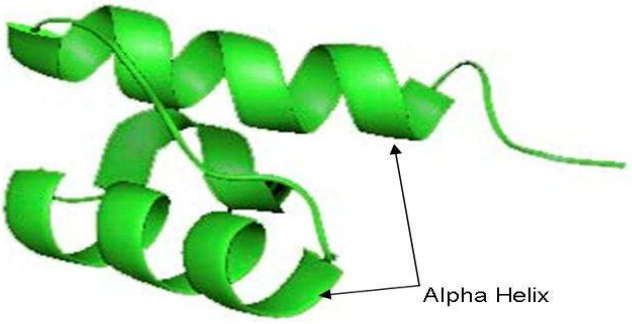

Figure 4. Helix Secondary Structure of the protein 1H1J

The beta sheet is the other common secondary structure made up of laterally packed beta strands. Hydrogen bonds formed between the backbone atoms in adjacent beta strands, within either the same or different polypeptide chains, form a

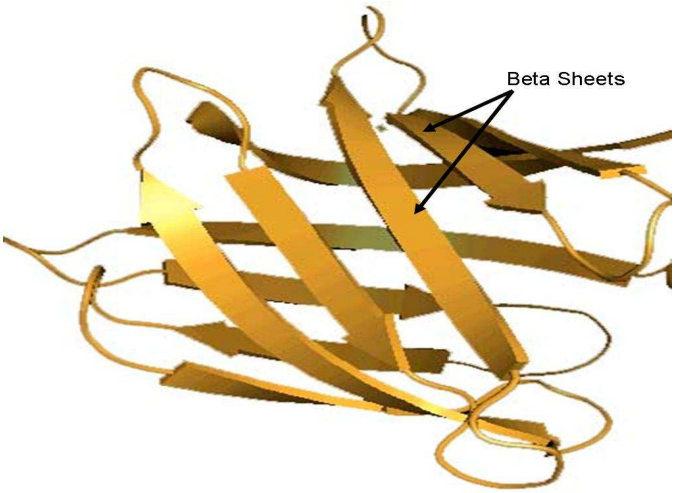

Figure 5. Beta Sheet Secondary Structure for Protein 1ICM (Partial Representation)

beta sheet. They are usually 5-8 residues long [4].

1.2.3 Protein tertiary structure

The next order in the hierarchy of protein structure, it is the overall conformation of a protein, the 3-dimensional conformation of all the amino acid residues in the protein. Tertiary structures are composed of secondary structures held together by hydrophobic interactions between the non-polar side chains or the disulfide bonds in some proteins [4].

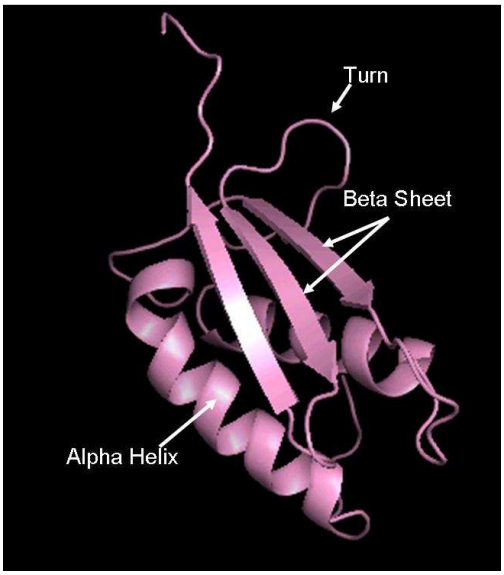

Figure 6. Tertiary Structure of a Protein 2DIR (Partial Representation)

1.2.4 Protein quaternary structure

The final order of the protein structure hierarchy, it is a combination of tertiary structures. They are described as the number and relative positions of the subunits in a

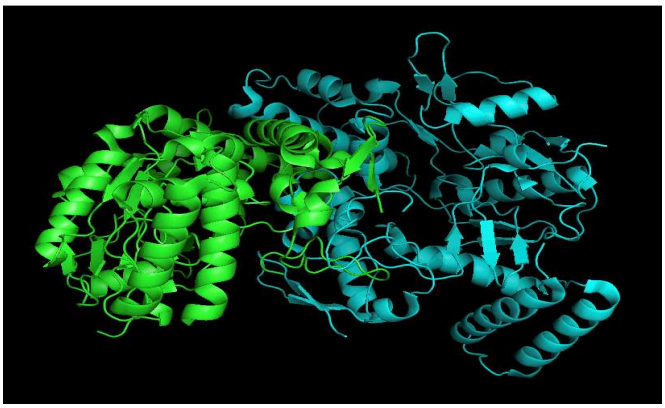

Figure 7. Protein Quaternary Structure for Protein 1ZZQ with a subunit in blue and the other in green

multimeric protein. A multimeric protein contains multiple subunits bound together by noncovalent interactions [4].

1.3 Protein Folding

The “thermodynamic hypothesis” asserts and finds that the 3-dimensional structure of a protein is determined by its amino acid sequence [6]. Protein folding is the way proteins organize themselves into unique 3-dimensional structures (native state) *via* a myriad of conformational changes or forms. These conformations are a complex network of atomic interactions. Patterns of protein folding kinetics, such as linear free energy relationships depict the landscape of the energies of protein conformations that can be modeled computationally [7]. Modeling protein folding is at the center for structure prediction, design, and analysis.

1.4 Protein Structure Prediction

The “thermodynamic hypothesis” [6] proposed by Christian B. Anfinsen is at the core of protein structure prediction. In this domain, the question posed is thus: “given just the amino acid sequence of a protein, can the 3-dimensional structure be

predicted?” This is a very active area of research in the field of computational structural biology. The 3-dimensional structure of a protein can be determined experimentally and, in recent years, a computational model of reasonable accuracy can often be produced. Due to the nature of some proteins, they do not allow for structure determination *via* the experimental methods - some proteins are extremely difficult to work with and some cannot be readily crystallized [8] (an integral step in X-ray crystallographic structure determination), leaving computational methods as the primary means for the structural study for some proteins.

The experimental solution to determining the 3-dimensional structure of a protein involves the use of procedures such as X-ray crystallography, Nuclear magnetic resonance spectroscopy (NMR), and Electron microscopy. X-ray crystallography provides detailed atomic information, however not all proteins structures can be determined *via* this method due to the inability of some proteins to readily crystallize. It works by passing an intense x-ray beam through the protein that has been crystallized and purified. The proteins in the crystal diffract the X-ray beam into one or another characteristic pattern of spots, which are then analyzed to determine the distribution of electrons in the protein. The resulting map of the electron density is then interpreted to determine the location of each atom.

NMR involves probing a protein with radio waves in a strong magnetic field after it has been purified. It is limited to medium and small proteins but to its advantage, it can determine the structure of a protein in solution. It also collects information about the conformation of atoms in a protein. Finally electron microscopy is used to determine

the structure of large proteins. The protein is subjected to an electron beam to capture an image of the protein [8].

There are three main computational methods for the prediction of protein structures - in hierarchical order of difficulty and complexity they are: Homology, Fold Recognition (Threading), and *ab initio* modeling. As it stands, the average cost to experimentally determine the structure of a protein is between U.S. $250,000 to $300,000 [9]. There is also a high demand for the prediction of structures shown by the popularity of prediction servers on the web providing predictions for over 20,000 unknown proteins submitted by about 2000 registered scientists [10].

Homology modeling involves comparing the sequence of an experimentally solved protein with that of an unknown protein in a one-to-one manner. If there is a 35% or more sequence similarity between the two, it is safe to assume that the unknown protein will assume the same overall fold of the known protein.

Fold recognition or threading involves comparing an unknown sequence against a library of structural templates and producing a list of scores. The fold with the best score is assumed to be the one adopted by the unknown sequence.

Ab initio modeling involves building three-dimensional protein models from scratch or without the aid of previously solved structures. It uses energy minimization and physical principles.

1.4 De Novo Protein Design

The term “*de novo*” means new, a first occurrence, to begin afresh or to start from the beginning anew. *De novo* protein design is the design of functional proteins

from scratch. It involves making a protein that will fold into a specifically defined 3-dimensional structure, with a sequence that is not related directly to that of any of the natural proteins. It also can be said to be the creation of protein sequence that will adopt a general fold given without any prior knowledge about the atomic level information of the desired target [11]. Computationally, *de novo* design can be accomplished by building a backbone framework representing the desired fold and designing an amino acid sequence that is compatible with that fold [12].

1.5 Defining Terms

The work within this project involves several concepts, applications, software and hardware. A brief history and explanation of what and where each of the pieces used is provided in this section in order to provide the intent and purpose.

1.5.1 PDB

The acronym PDB stands for protein data bank. Established in 1971 at Brookhaven national laboratory, it is a repository for the 3-dimensional structures of the experimentally determined proteins and nucleic acids. It is managed by the Research Collaboratory for Structural Bioinformatics (RCSB) and is updated weekly. As of October 20th 2010, 68,701 experimentally determined structures were available from the PDB repository [13]. Proteins are stored as files in several formats and can be viewed using text editor or a visualization program. The files contain atomic 3-dimensional coordinates in space and other information that describes each protein found in the

repository [14]. The data found in the PDB is paramount to this project and lays a foundation for this project in regards to protein prediction, design and analysis.

1.5.2 DSSP

DSSP stands for Dictionary of Secondary Structure of Proteins. Authored by Wolfgang Kabsch and Christian Sander, given atomic coordinates in the PDB format, it uses hydrogen bonds as a metric in a pattern recognition process to define secondary structure, geometrical features and solvent exposure of proteins [15, 16]. Secondary structures are recognized as repeating bridges and turns, which are the elementary hydrogen bonding patterns.

Secondary Structure Name	**Designation/Assignment**
Alpha Helix	H
A residue in isolated beta-bridge	B
An extended strand, participates in beta ladder	E
3_{10} Helix	G
π-Helix (5-Helix or pi helix)	I
Hydrogen bonded turn	T
Bend	S

Table 1. DSSP Output for Secondary Structure Designation [14].

When a repeating bridge is encountered a ladder is formed. A set of one or more consecutive bridges of the same type constitute a ladder and connected ladders make up a sheet (beta sheet). Helices are made up of repeating turns [16]. Each of the PDB files gotten from the repository is defined and stored for further analysis using the DSSP program. For this project we are focused primarily on the alpha helix secondary

structures. Table 1 shows the outputs from the DSSP program that corresponds to the secondary structures assignments or definitions.

1.5.3 ProtCAD

ProtCAD stands for Protein Computer Assisted Design. It is a software library of object-oriented general-purpose protein modeling and analysis tools written in C++ by my thesis advisor Christopher Summa. It was originally designed to be used primarily for *de novo* protein design. A variant of the library is being developed for use in protein structure prediction and design. For this project the library was used to develop a program to calculate a series of geometric entities for alpha helices based on both the 3-dimensional atomic coordinates of a protein (PDB file) and the corresponding DSSP designation for the protein (PDB file) in question.

1.5.4 MySQL

MySQL is the name of one of the world's most popular open source relational database systems. Originally founded and developed in Sweden by David Axmark, Allan Larsson and Michael Widenius. Today, the company Oracle owns the MySQL trademark. It is know for its speed, reliability, ease of use, and compatibility on a host of different platforms [17]. Relational database systems such as MySQL are used manage, create and control databases. A database is a collection of data, organized with a type designation as opposed to text files which store all data as text. The information in a database can be queried or retrieved using SQL. The acronym SQL stands for structured query language. It is the language of operation in the database world.

Storing, searching, updating and other operations are facilitated through this language. For this project we use version 5.0.82 to create and manage our database.

1.5.5 MySQL++

This is a C++ wrapper built for MySQL. It enables MySQL to be interacted with from within a C++ program. It is built with the same principles as the standard C++ library so as to make working with a MySQL database as easy as dealing with C++ Standard Template Library (STL) containers. Also, it offers native C++ interfaces for common tasks [18]. MySQL++ is used in this project to aid a direct interaction from within the ProtCAD library to our MySQL database. The version we use for this project is version 3.1.0.

1.5.6 CATH

The acronym CATH translates to Class, Architecture, Topology, and Homology. It is a database of hierarchical domain classification of protein structures in the PDB. The proteins in the CATH database are divided into structural domains and assigned into homologous superfamilies (groups of domains that are related by evolution). The protein structures in the CATH are classified by class – a classification according to the secondary structure composition: mostly alpha, mostly beta, mixed alpha/beta or few secondary structures. Architecture – involves classifying structures according to their overall shape as determined by the orientations of the secondary structures in 3 - dimensional space without taking into consideration the connectivity between them. Topology/Fold – involves classifying the structures according to fold groups. Structures

are grouped based on their overall shape and connectivity of the secondary structures. Lastly, homology – involves grouping together those proteins that share a common evolutionary ancestor.

In general, the CATH database provides an overall view of the known protein structure universe to date. It provides information on the structure, function, and evolutionary relationships of a protein as well as the information involving a structure of interest against other proteins in the database [19].

Chapter 2: Project Overview

The work described in this thesis project entails building a database of secondary structures to aid our efforts in protein design, structure analysis and structure prediction. It involves populating a MySQL relational database with protein secondary structure geometry and sequence information. With a copy of the PDB repository on our dedicated server, the first table in the database is created. This table contains all the proteins on our dedicated server copied from the PDB, categorized by secondary structure classifications: alpha helix, beta sheet, coil, non-grouped, non-grouped alpha helix, and non-grouped beta sheet as portrayed by the flowchart diagram in **Figure 8**. The second table (full geometric dataset) in the database contains geometric information of all the proteins on our server that contains at least two helices. The helices also have to meet a criterion of containing at least two turns (8 or more residues long). The third table is a subset of the second with information pertaining to only those proteins that have non-redundant chains and have less than a 35% sequence similarity. The fourth table is made up of a set of the proteins from the CATH (Class, Architecture, Topology and Homology) database. The fifth table a smaller subset of the second table (full geometric dataset) is made up of the set of membrane proteins gathered from the Stephen White Laboratory at UC Irvine [20]. The sixth table is a set of soluble proteins, a larger subset of the second table. Tables 5 and 6 make up the second. The flowchart for the creation of these tables is depicted by **Figure 9**.

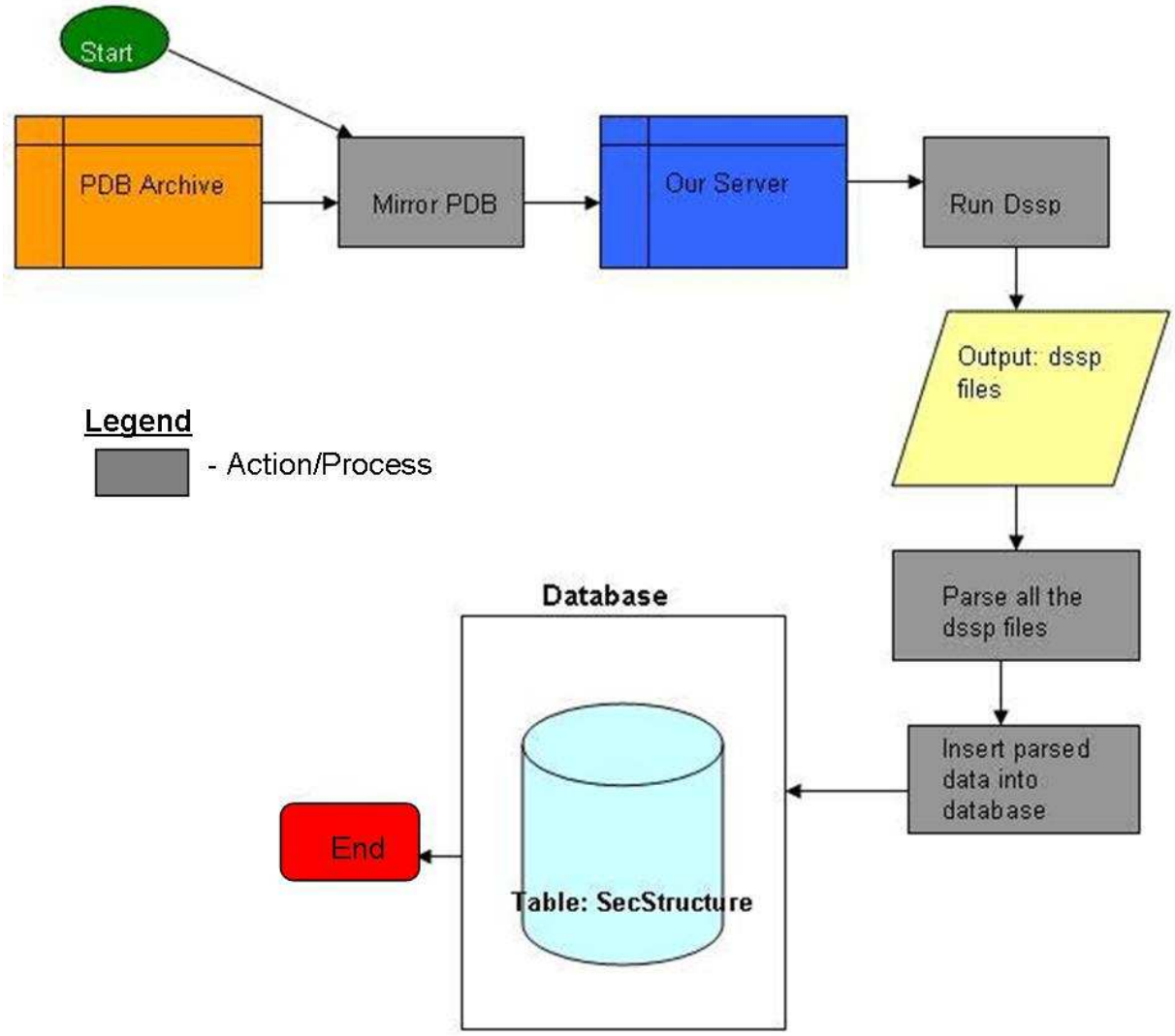

Figure 8. System Flowchart for Secondary Structure Table in the Database

Chapters 3 and 4 describe in detail, what has been done and how the database is created. Chapters 5 and 6 results obtained from the data, conclusions, and the future work for this project.

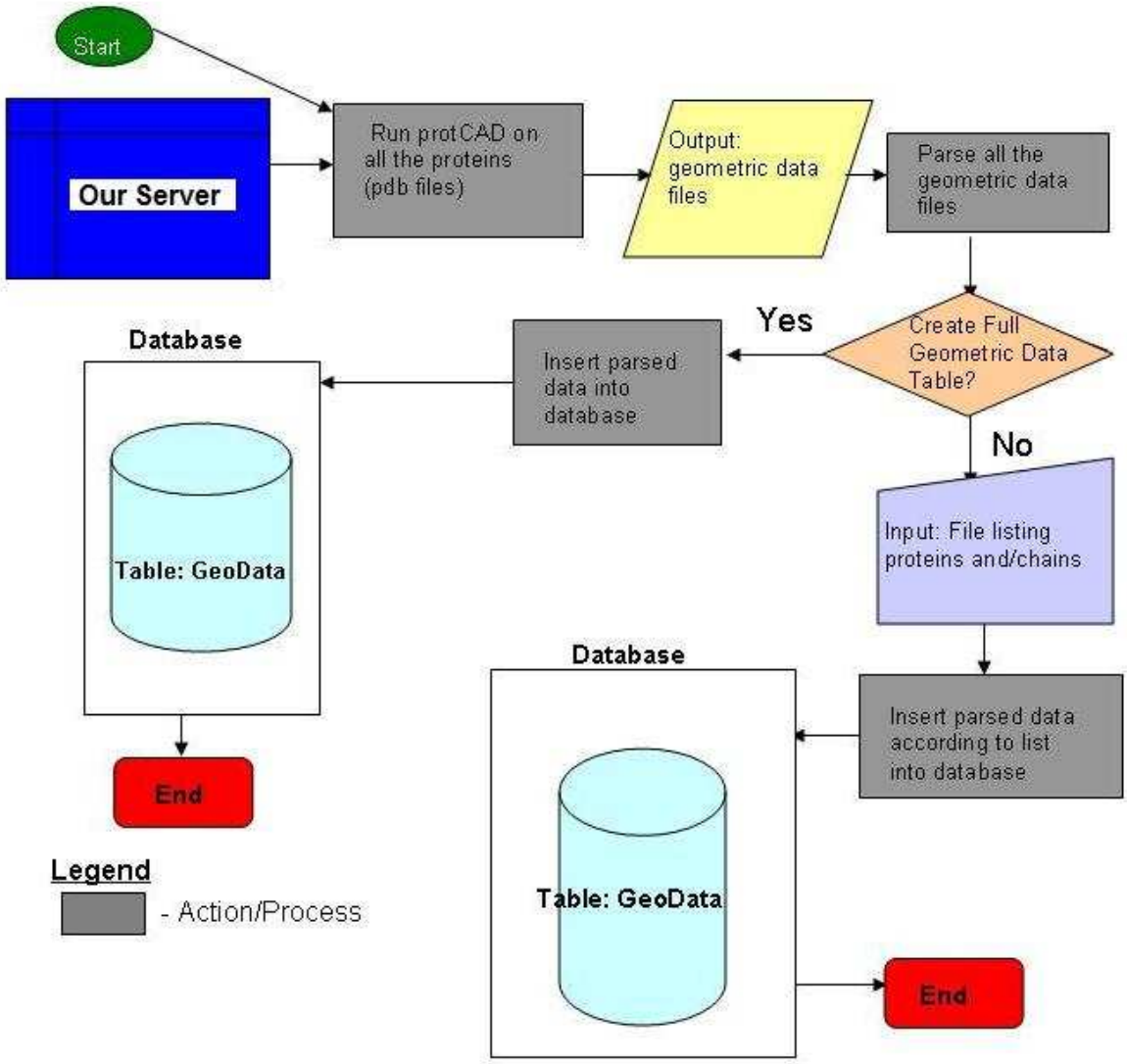

Figure 9. System Flowchart for Geometric Data Tables in the Database

Chapter 3: Database of Secondary Structure Elements

3.1 Introduction

In order to create a database of secondary structure elements, data of all the experimentally determined proteins from the protein data bank (PDB) have to be gathered and processed. With a database created, a table therein is also created to hold the elements of secondary structures as defined. Using the data from each protein in PDB format, we create attributes that make up the table in the database to hold the secondary structure elements. The table named *SecStructure* has nine (9) attributes/fields.

Field	Type	Null	Key	Default	Extra
UniqueID	varchar(11)	NO	PRI		
PDB_ID	varchar(5)	YES		NULL	
SSElementType	varchar(25)	YES		NULL	
Start_Res_ID	int(11)	YES		NULL	
End_Res_ID	int(11)	YES		NULL	
Chain_ID	varchar(5)	YES		NULL	
Sequence	longtext	YES		NULL	
Relative_Index	int(11)	YES		NULL	
Absolute_Index	int(11)	YES		NULL	

Figure 10. Attributes of Database Table: SecStructure

3.2 Generation of Secondary Structure Elements Dataset

The dataset is created from all the protein data culled from the Protein Data Bank. As mentioned earlier, all the protein data in the PDB is processed, assigned secondary structure elements and categorized using nine attributes. These attributes make up the data for a protein that is stored or held in a table in this database. The attributes for a protein are described as follows:

1. UniqueID – A combination of ‘Absolute_Index’ and ‘PDB_ID.’ The primary key of the table ensures uniqueness of entries.
2. PDB_ID – The PDB accession code of the protein (e.g. 1ROP for the ROP protein).
3. SSElementType – Secondary structure assignment: Alpha Helix, Beta Sheet, Coil, Not Grouped, Not Grouped_Helix, Not Grouped_BSheet. They are made up of one or several amino acid residues.
4. Start_Res_ID – Beginning/First residue ID e.g. 45 for a ‘SSElementType.’
5. End_Res_ID – Ending/Last residue ID for a ‘SSElementType.’
6. Chain_ID – The chain designation for the ‘SSElementType.’
7. Sequence – The residues that make up the ‘SSElementType.’
8. Relative_Index – The count per ‘SSElementType’ for a protein. For instance, the first Alpha_Helix encountered in the protein has a ‘Relative_Index’ of 1 and the second has an index of 2 and so on.
9. Absolute_Index – The count of the ‘SSElementType’ in the protein. For instance, the first ‘SSElementType’ encountered in the protein has an ‘Absolute_Index’ of 1 and the second has an index of 2 and so on.

3.3 Methods/Tools

Our programs were developed using C-Shell scripts, PERL (Practical Extraction Report Language) scripts, XGrid (Apple’s Distributive Computing Platform), and the DSSP program for defining the secondary structure of each protein culled from the PDB, MySQL version 5.0.82, under Mac OSX 10.6.4.

3.4 Implementation of MySQL Database

In order to create the database that contains the table of secondary structure elements categorized by the attributes described in section 3.2, a copy of the PDB repository is made on our dedicated server using the unix synchronization utility *'rsync.'* Using the following command:

rsync -a --port=33444 ftp.wwpdb.org::ftp_data/structures/divided/pdb/ .

We create a directory on our server called 'pdb' in which the protein data from the PDB is stored. As soon as the protein data is culled into our local repository, we run a PERL program/script which uses XGrid to distributively uncompress the protein data, create result folders corresponding to all the protein subfolders and define the proteins using DSSP. For instance, for every PDB file a DSSP file is created as its definition. As jobs from within our program are sent from a client assuming all authentication protocols are satisfactory, the controller (our server) looks for an available agent that can handle our request. Once these conditions are satisfied, the agent performs our job request and sends the controller the results, which are now relayed back to the client upon request.

Figure 11 shows a how XGrid facilitates our DSSP execution.

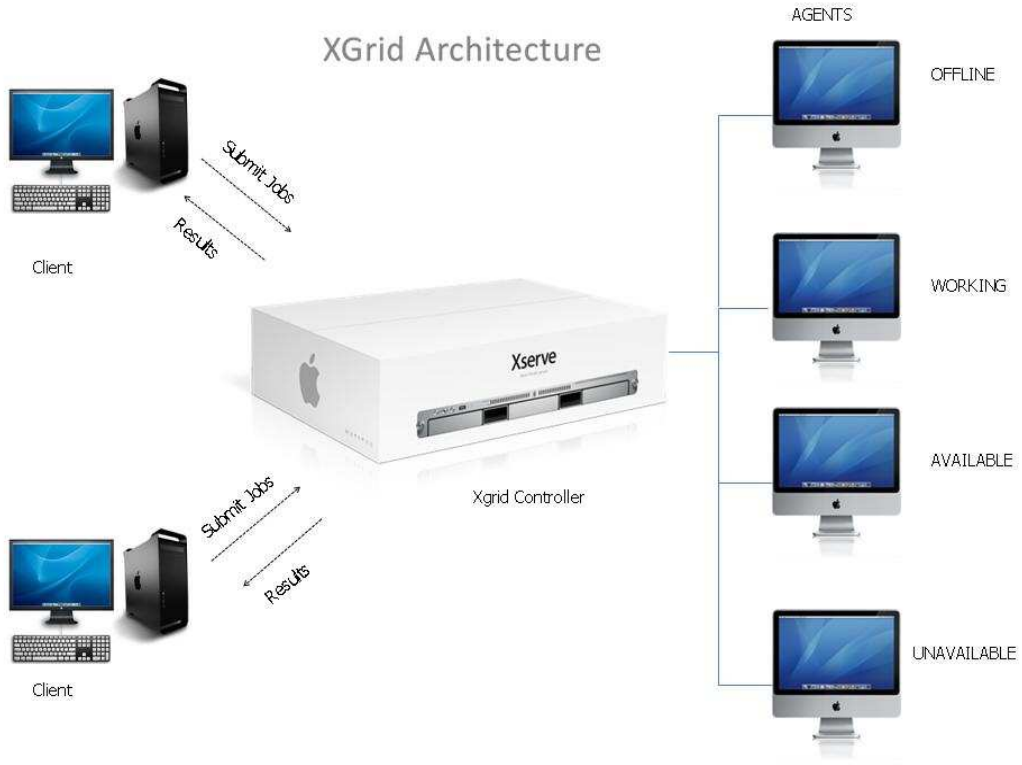

Figure 11. Apple's XGrid Architecture

The next step in the process of creating the database is parsing through all the DSSP files created. A PERL program does the parsing of the DSSP files and creates parsed text files corresponding to each DSSP file that is finally used to populate the SecStructure table of the database. In this program, we define the nine attributes mentioned in section 3.2. Despite the definition of secondary structure elements by the DSSP program it is important that as these DSSP files are being parsed we classify the elements appropriately. An alpha-helix is identified by the 'H' in the 'Structure' attribute of the DSSP file. They are classified as being an alpha-helix when they are at least eight (8) sequential residues long. This is to ensure that for any alpha-helix classified, we are guaranteed at least two (2) turns. Any alpha-helix that is less than eight residues

long is classified as a ‘Not Grouped_Helix.’ Beta Sheets are identified by the ‘E’ in the ‘Structure’ attribute of the DSSP file. They are classified as a beta sheet when they are at least five (5) sequential residues long. Those less than five residues long are classified as ‘Not Grouped_BSheet.’ Coils have no restrictions like those for sheets and helices rather, residues that are identified by a ‘space, T, and S’ in the ‘Structure’ attribute in DSSP are classified as coils. Everything else is classified as ‘Not Grouped.’ This classification is uniformly followed throughout this project unless stated specifically.

Finally, another program whose purpose is to insert the contents of the parsed files created from the DSSP files into the SecStructure table of the database is executed. Its input is a parsed DSSP file and after validation of input, a connection to MySQL server is made and a database is created, it then checks to see if the SecStructure table has been created and if not create it and begin population.

3.5 Results

For creating this table, 68,470 proteins were used and from these proteins 6944257 records/elements where created classified according to the various classifications described in section 3.4. Table 2 shows some statistics for this table and **Figure 12** shows a sample DSSP file.

Table	Record Count
SecStructure	6,944,257
Element Type	
Alpha-Helix	768,284
Beta Sheet	852,130
Coil	345,5405
Non Grouped	875,095
Non Grouped_Helix	332,382
Non Grouped_BetaSheet	660,961

Table 2. Database Statistics for Table SecStructure

```
#  RESIDUE AA STRUCTURE BP1 BP2  ACC     N-H-->O   O-->H-N   N-H-->O
1    1 A M              0   0   84      0, 0.0     2,-0.3     0, 0.0
2    2 A N     >  -     0   0   71    156,-0.0     4,-2.7     1,-0.0
3    3 A I   H  > S+     0   0   26     -2,-0.3     4,-3.1     1,-0.2
4    4 A F   H  > S+     0   0   77      2,-0.2     4,-2.0     1,-0.2
5    5 A E   H  > S+     0   0   98      1,-0.2     4,-1.8     2,-0.2
```

Figure 12. Sample DSSP File for Protein 200l (Partial Representation)

3.6 Discussion/Conclusion

The data gathered, processed and input into a database is the data of 68,470 proteins culled from the PDB on October 11th 2010. The full-atom data in the individual PDB files are interpreted by the DSSP program, and each amino acid is assigned a secondary structure class. This data is further analyzed by collapsing contiguous elements of secondary structure into multi-amino acid stretches with the following stipulations: alpha-helices must have eight or more consecutive amino acids classified as helical (H's) in the DSSP file, beta sheets must have at minimum five consecutive ladders (E's). Those elements not meeting these criteria are classified as not grouped_helix and non grouped_betasheet. Coils are classified as any residue defined

as unclassified (i.e. represented by a ‘space’ in the DSSP file), ‘T’ or ‘S’ in the DSSP file while all the other definitions are classified as not grouped. The data held by this table provides a means to evaluate or analyze for all the proteins in the PDB their various secondary structure elements and attributes at a glance. Though our database is not as categorized as the CATH database, it provides a framework for analysis in regards to protein, design, prediction and folding.

Chapter 4: Database of Geometric Information of Secondary Structure Elements (SSE)

4.1 Introduction

In order to create the database of geometric information of secondary structure elements, we use the data gathered during the process of creation of the database of secondary structure elements. The process of gathering data for this dataset is similar to that described in section 3.4. Using the data for each protein in PDB format as well as its corresponding DSSP file, we calculate several geometric attributes of helix-helix interactions and create fields that represent them in a table in the database. As mentioned in chapter 2 we create several tables: *SecStrucGeoData, NonRedundantChainsGeoData, CathGeoData, MembraneGeoData,* and SolubleGeoData. These tables hold data with the same attributes but with different classification and can be linked together. The *SecStrucGeoData* holds all the helix-helix interactions of all the proteins in the PDB as of October 11th 2010. The table *NonRedundantChainsGeoData* holds all the helix-helix interactions of all the proteins culled from the PDB as of the same date above that have a sequence homology of less than 35%. The *CathGeoData* table holds the helix-helix interactions gathered from the CATH database of the protein classifications till date. We use version 3.3.0 which was filed June 24th 2009. The *MembraneGeoData* table is generated using a list of membrane proteins of known 3-dimensional structure gathered from the Stephen White Laboratory at

UC Irvine. Lastly, the *SolubleGeoData* table consists of the helix-helix interactions of the non-membrane proteins in the PDB. The protein data of both the soluble and membrane proteins make up the full dataset of the *SecStrucGeoData*. The tables described all have twenty-seven (33) attributes/fields each. **Figure 13** shows the attributes for these tables.

Field	Type	Null	Key	Default	Extra
UniqueID1	varchar(15)	NO	PRI		
UniqueID2	varchar(15)	NO	PRI		
Helix_Seq1	longtext	YES		NULL	
Helix_Seq2	longtext	YES		NULL	
Start_Res_ID_H1	int(11)	YES		NULL	
End_Res_ID_H1	int(11)	YES		NULL	
Start_Res_ID_H2	int(11)	YES		NULL	
End_Res_ID_H2	int(11)	YES		NULL	
Chain	varchar(2)	YES		NULL	
Distance_ClosestApproach	double	YES		NULL	
CrossingAngle	double	YES		NULL	
ChothiaCrossingAngle	double	YES		NULL	
MinDistance_AAPair	double	YES		NULL	
ResidueID_AAPair1	int(11)	YES		NULL	
ResidueID_AAPair2	int(11)	YES		NULL	
POCA1_X	double	YES		NULL	
POCA1_y	double	YES		NULL	
POCA1_Z	double	YES		NULL	
POCA2_X	double	YES		NULL	
POCA2_Y	double	YES		NULL	
POCA2_Z	double	YES		NULL	
EndPts_Axis_11_X	double	YES		NULL	
EndPts_Axis_11_Y	double	YES		NULL	
EndPts_Axis_11_Z	double	YES		NULL	
EndPts_Axis_12_X	double	YES		NULL	
EndPts_Axis_12_Y	double	YES		NULL	
EndPts_Axis_12_Z	double	YES		NULL	
EndPts_Axis_21_X	double	YES		NULL	
EndPts_Axis_21_Y	double	YES		NULL	
EndPts_Axis_21_Z	double	YES		NULL	
EndPts_Axis_22_X	double	YES		NULL	
EndPts_Axis_22_Y	double	YES		NULL	
EndPts_Axis_22_Z	double	YES		NULL	

Figure 13. Attributes of the Database Tables mentioned above

4.2 Generation of Geometric Dataset of Secondary Structure Elements (Helices)

The dataset is created from all the protein data culled from the Protein Data Bank and their corresponding DSSP files. The data gathered as explained in section 3.4 are processed to calculate and categorize the helix-helix interactions. Thirty-three attributes make up the data for all the helix-helix interactions if any for each protein in the PDB. **Figure 14** illustrates the attributes of helix-helix interactions that we calculate and gather, while these attributes for each protein are described as follows:

1. UniqueID1 – A combination of 'Absolute_Index' of helix1 in the **helix**-helix interaction and 'PDB_ID.' One half of the composite key (primary key of more than one attribute) of the table ensures uniqueness of entries.
2. UniqueID2 – A combination of 'Absolute_Index' of helix2 in the helix-**helix** interaction and 'PDB_ID.' The other half of the composite key (primary key of more than one attribute) of the table ensures uniqueness of entries.
3. Helix_Seq1 – The residues that make up helix1 of the **helix**-helix interaction.
4. Helix_Seq2 – The residues that make up helix2 of the helix-**helix** interaction.
5. Start_Res_ID_H1 – Beginning/First residue ID of helix1.
6. End_Res_ID_H1 – Ending/Last residue ID of helix1.
7. Start_Res_ID_H2 – Beginning/First residue ID of helix2.
8. End_Res_ID_H2 – Ending/Last residue ID of helix2.
9. Chain – The chain designation of the helices interacting.
10. Distance_ClosestApproach – This is a length of the line segment formed at the point where both helical axes are closest. The helical axes are defined by the 3-dimensional atomic coordinates of helix1 and helix2.

11. CrossingAngle – The angle formed between helix1 and helix2 axes.
12. ChothiaCrossingAngle – the angle formed between helix1 and helix2 axes as defined by Chothia et al [21].
13. MinDistance_AAPair – The least inter-atomical distance between the atoms of both helices.
14. ResidueID_AAPair1 – The ID of the atom from helix1 that produces the least distance as described in 13 above.
15. ResidueID_AAPair2 – The ID of the atom from helix2 that produces the least distance as described in 13 above.
16. POCA1_(X,Y, Z) – The ‘X,Y, Z’ coordinates of the start point of the line segment of closest approach on helix1.
17. POCA2_(X,Y,Z) – The ‘X,Y, Z’ coordinates of the end point of the line segment of closest approach on helix2.
18. EndPts_Axis_11_(X,Y,Z) – The ‘X,Y, Z’ coordinates of the start point of the line segment defined by helix1.
19. EndPts_Axis_12_(X,Y,Z) – The ‘X,Y, Z’ coordinates of the end point of the line segment defined by helix1.
20. EndPts_Axis_21_(X,Y,Z) – The ‘X,Y, Z’ coordinates of the start point of the line segment defined by helix2.
21. EndPts_Axis_22_(X,Y,Z) – The ‘X,Y, Z’ coordinates of the end point of the line segment defined by helix2.

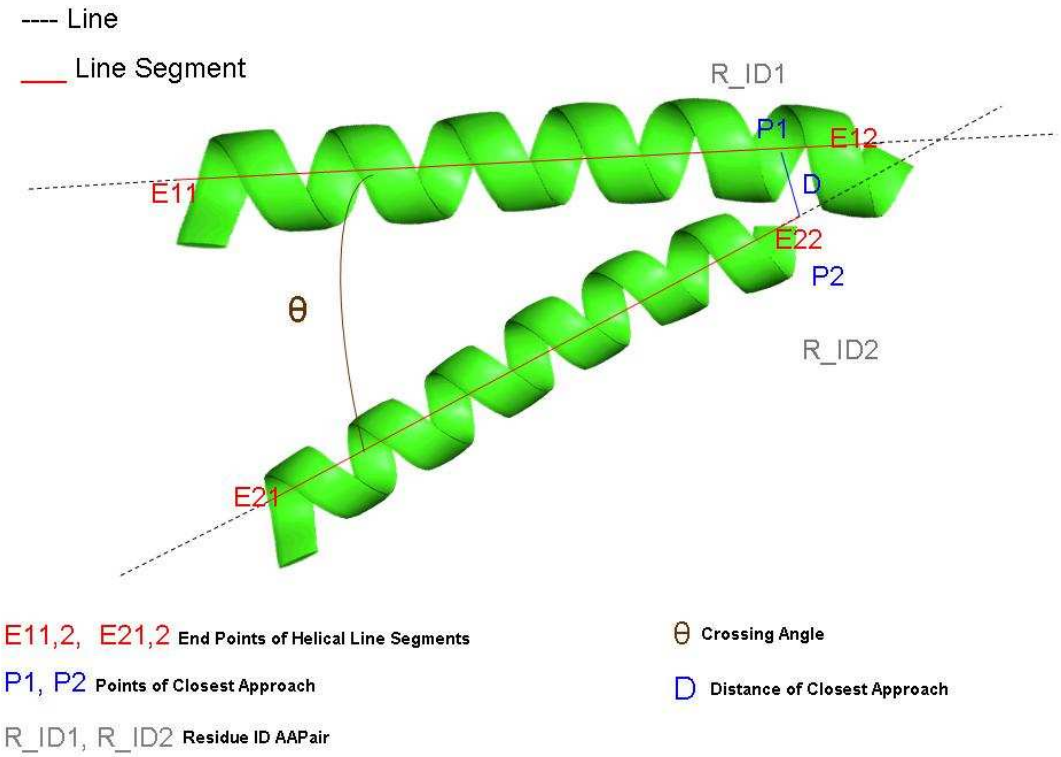

Figure 14. Helix-Helix Interactions

4.3 Generation of Non Redundant Geometric Dataset of Secondary Structure Elements (Helices)

In order to create this dataset, a list of non redundant proteins is created *via* the 'Pieces' [22] web server. These proteins and their respective chains have a sequence homology of less than 35%. The list of 12009 proteins and their chains were culled October 6th 2010. The protein and chains in the list corresponding to the PDB and DSSP files we have gathered from 3.4 are used to create the *NonRedundantChainsGeoData* table containing geometric information for the helix-helix interactions. The data is filtered from the full dataset described in 4.2 by protein and chain.

4.4 Generation of CATH Geometric Dataset of Secondary Structure Elements (Helices)

This dataset is created using the list of the most recent proteins as classified by the CATH database. For this project the CATH list used was version 3.3.0 which was created on July 24th 2009 and made up of 2490 unique proteins and their respective chains. The geometric information relating to their helix-helix interactions are retrieved from within the geometric data gathered for all the proteins in the PDB as in 4.2 and input into the *CathGeoData* table.

4.5 Generation of Membrane Proteins Geometric Dataset of Secondary Structure Elements (Helices)

The membrane proteins dataset is created using a list culled from the Stephen White Laboratory at UC Irvine of 3-dimensional structures of membrane proteins. This list of 432 proteins was gathered on the 29th of October 2010. The geometric data corresponding to the proteins in the membrane proteins list are filtered from the generation of the geometric information of all the proteins in the PDB and input into the *MembraneGeoData* table.

4.6 Generation of Soluble Proteins Geometric Dataset of Secondary Structure Elements (Helices)

The soluble geometric dataset like the membrane dataset is a subset of the full dataset of geometric information of all the proteins in the PDB. The *SolubleGeoData*

table is made up of the non-membrane proteins. The data for this table constitutes data from 68,038 proteins. The geometric information corresponding to these proteins are sieved from the full geometric dataset and input into the table.

4.7 Methods/Tools

Our programs were developed using C++, C-Shell scripts, PERL (Practical Extraction Report Language) scripts, XGrid (Apple's Distributive Computing Platform), and the DSSP program for defining the secondary structure of each protein culled from the PDB, GNU C compiler (gcc) version i686-apple-darwin10-gcc-4.2.1, MySQL version 5.0.82, under Mac OSX 10.6.4.

4.8 Implementation of MySQL Database

We developed a C++ program that uses the protCAD library developed by my thesis advisor to calculate the various geometric attributes that constitute to helix-helix interactions. The program takes as input the name of PDB file but requires for optimum results that both the PDB and DSSP files are in a similar location to ensure accurate assignment of secondary structure elements within the program. The output is a text file of geometric information calculated for all helix-helix interactions for the PDB file input. We ensure that no inter-chain helix-helix interactions are performed. For instance, in a protein with chains 'A' and 'B', we make sure that we are not calculating and collecting geometric information for helix-helix interactions for any helix in chain 'A' and another in chain 'B'. All the helix-helix interactions we collect are for helices in the same chain. Symmetry is accounted for in the interaction process as well. For example, the

geometric calculation involving helix1 and helix2 is the same as that involving helix2 and helix1 therefore, only one of these calculations are performed and collected.

We utilize a C-Shell script that executes the C++ program which utilizes as input all the proteins and their DSSP files gathered. The executions are done *via* XGrid (Apple's Distributive Computing Platform). Assuming all authentication protocols are satisfactory, the controller (our server) looks for an available agent that can handle the job requests sent from within our C-Shell program on a client machine. **Figure 11** shows how the distributive computation takes place.

Once we have collected all the geometric result files of helix-helix interactions corresponding to all the proteins in our local repository, using Perl scripts we parse each of them and create parsed text files which we use to create the tables in our database. **Figure 15** shows the geometric information for the helix-helix interaction in the ROP protein.

```
header line: 0
Atom: CE
Atom: NZ
Atom: CG
Atom: CD
Atom: CE
Atom: NZ
Atom: CE
checking for existence of pdb1rop.dssp
file exists
EXPDTA    X-RAY DIFFRACTION
XRAY  RESOLUTION: 1.7

# ----- Geometric Data Collection----- #

 ----- START HANDLING 2 HELICES -----
 1 - Line of Closest Approach between residues 3  to 28  and 32  to 55

Sequences of Secondary Structure Elements
 2 - Sequence of SS Element 1: KQEKTALNMARFIRSQTLTLLEKLNE - chain: A
 3 - Sequence of SS Element 2: DEQADICESLHDHADELYRSCLAR - chain: A

Points of Closest Approach
 4 - Point 1: 53.2758   5.74593     34.3415
 5 - Point 2: 52.9874   14.0367     32.3811

 6 - Distance of closest Approach: 8.52429

Endpoints of the Axes of the Helices
 7 - Axis 1 - Point 1: 36.604 0.685471    15.3938
 8 - Axis 1 - Point 2: 61.1106      8.12403     43.2458

 9 - Axis 2 - Point 1: 63.8988      15.9149     38.7184
 10 - Axis 2 - Point 2: 34.1291     10.7907     21.4284

 11 - The cross angle between the current 2 helices: 18.453
 11b - The Chothia cross angle between the current 2 helices: 18.453
 12 - Minimum Distance of the pair of atoms between both helices: 4.98261
 13 - Residue ID for the pair Start: 16  End: 45
 ----- DONE HANDLING 2 HELICES -----
```

Figure 15. Geometric Information for Protein 1ROP

Finally, we create tables: *SecStrucGeoData*, *NonRedundantChainsGeoData, CathGeoData, MembraneGeoData,* and SolubleGeoData using the parsed geometric text files. For the *SecStrucGeoData* table we use all the geometric information that has been created. In creating the *NonRedundantChainsGeoData* table, we use the list of

protein chains with less 35% sequence homology culled from Dunbrack Lab 'Pieces' web server [22] to sieve out our non redundant data from the batch of geometric information collected. In essence, the non redundant geometric data is a subset of all the geometric data by protein chain of all the proteins gathered. For the *CathGeoData* table we utilize the most recent list of classified protein structures from the CATH database to filter out the proteins by their designated ID and chain. The *MembraneGeoData* table, a subset of all the geometric data gathered is created using a list of all the membrane proteins in the PDB culled from the Stephen White Laboratory. Finally, the *SolubleGeoData* table, a much larger subset of all the geometric data is created from all the non-membrane proteins.

4.9 Results

There are 68,470 proteins in our local repository but only those that have alpha-helices were used to calculate helix-helix interactions. It is important to note that of the 68,470 proteins some do not have helices or do not have more than one helix per chain to permit a helix-helix interaction calculation. From the proteins that were used, 3,064,352 records/interactions are created for the *SecStrucGeoData* table. In regards to the *NonRedundantChainsGeoData* table 211,433 records/interactions were collected and input into the table from a list of 12,009 proteins and their chain designations. 42,719 helix interactions are recorded from 2,490 proteins and their respective chain designations and make up the *CathGeoData* table. The *MembraneGeoData* and *SolubleGeoData* tables produce 49,119 and 3,015,233 records from 432 and 68,038 proteins respectively. **Table 3.** shows some statistics for the various tables.

Table	Record Count
SecStructure	6,944,257
SecStrucGeoData	3,064,352
NonRedundantGeoData	211,433
CathGeoData	42,719
MembraneGeoData	49,119
SolubleGeoData	3,015,233

Table 3. Database Statistics for all the Database Tables

Of the various geometric entities calculated for the various tables in **Table 3.** The crossing angle between helical axes are calculated in two different ways with criteria for the calculation of the angles (Ω) between helix axes having both directionality ($-180° \leq \Omega < 180°$) and non-directionality ($-90° \leq \Omega < 90°$) as defined by Chothia et al [21]. We graphed our findings for the various tables:

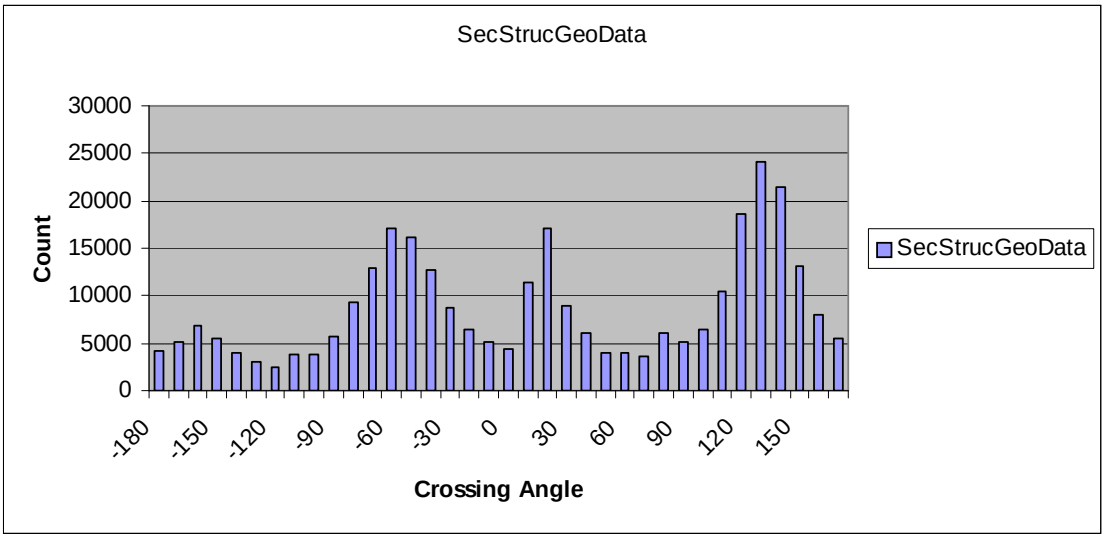

Figure 16. SecStrucGeoData with Helix Directionality

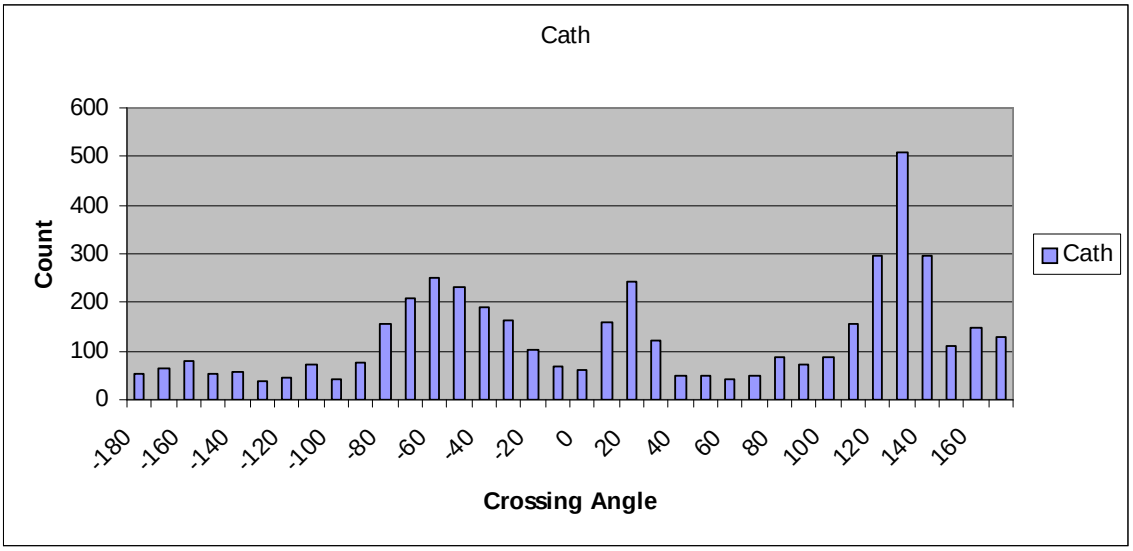

Figure 17. CathGeoData with Helix Directionality

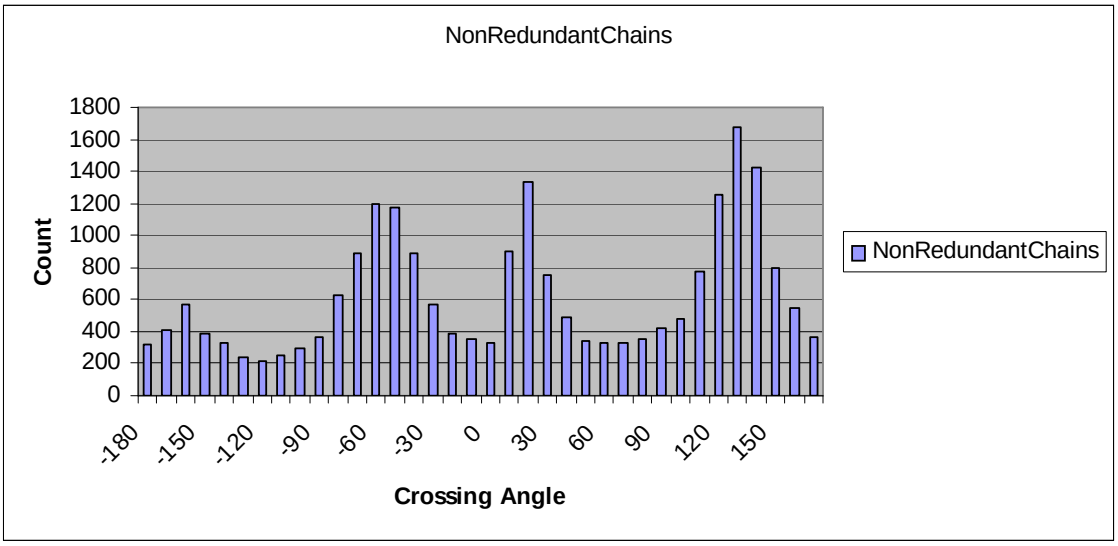

Figure 18. NonRedundantGeoData with Helix Directionality

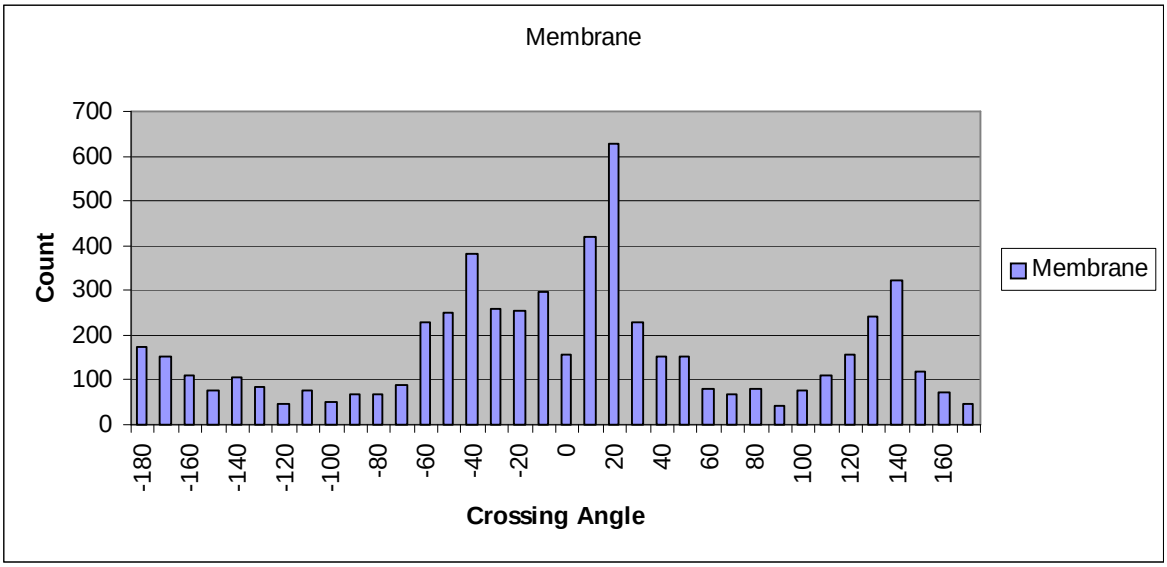

Figure 19. MembraneGeoData with Helix Directionality

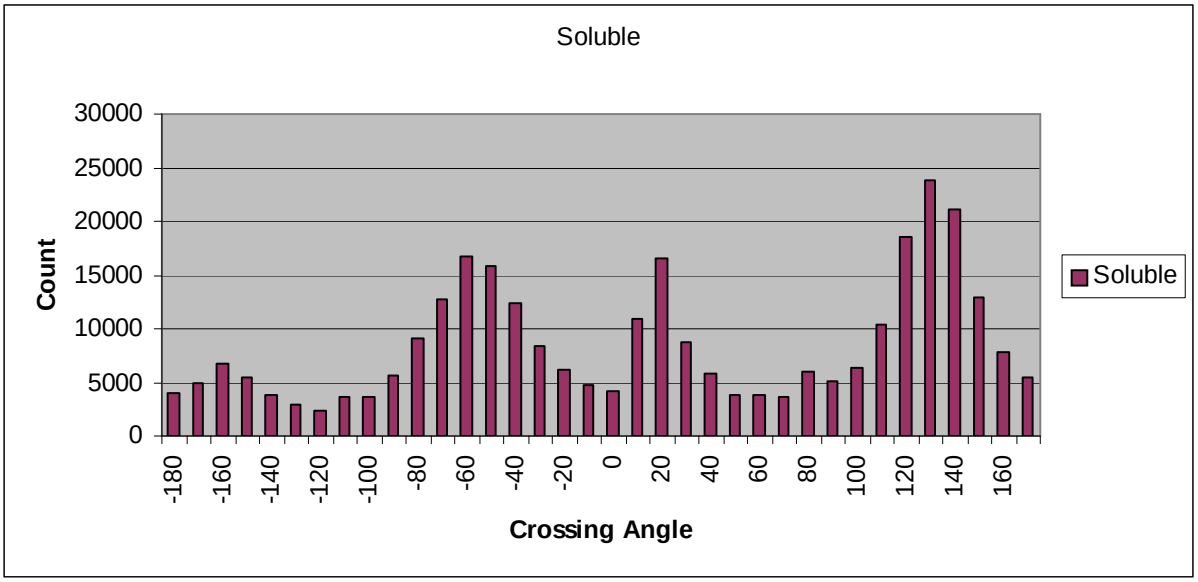

Figure 20. SolubleGeoData with Helix Directionality

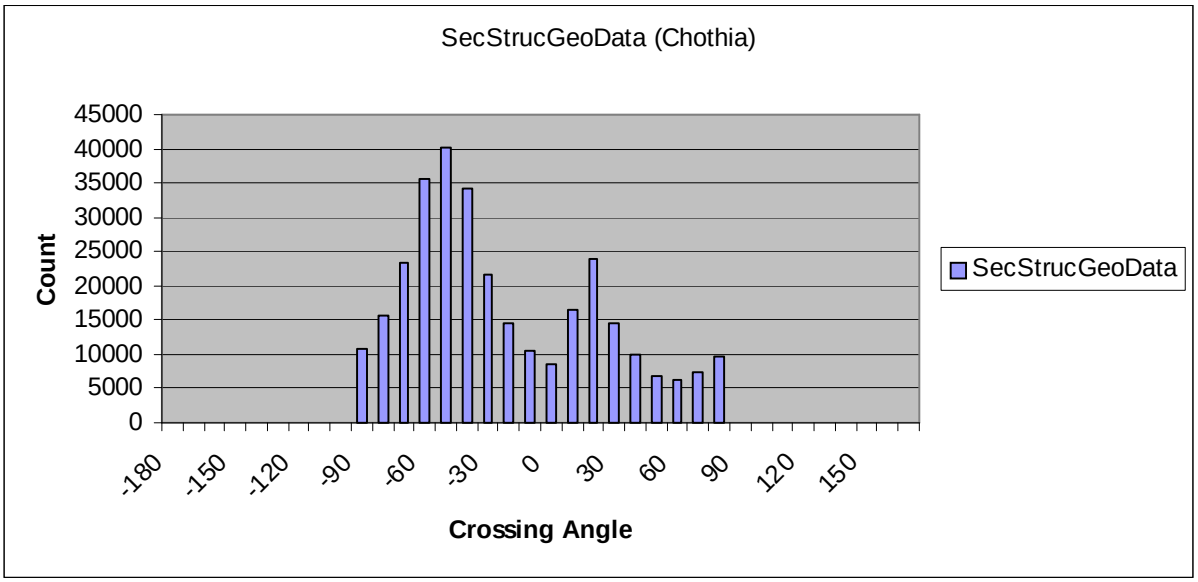

Figure 21. SecStrucGeoData without Helix Directionality

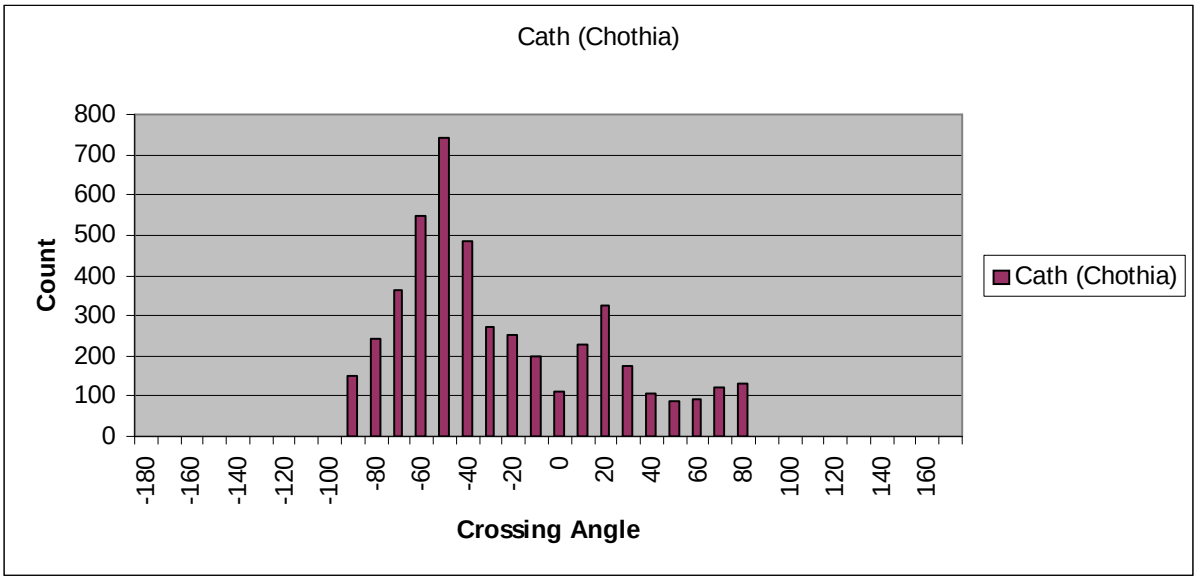

Figure 22. CathGeoData without Helix Directionality

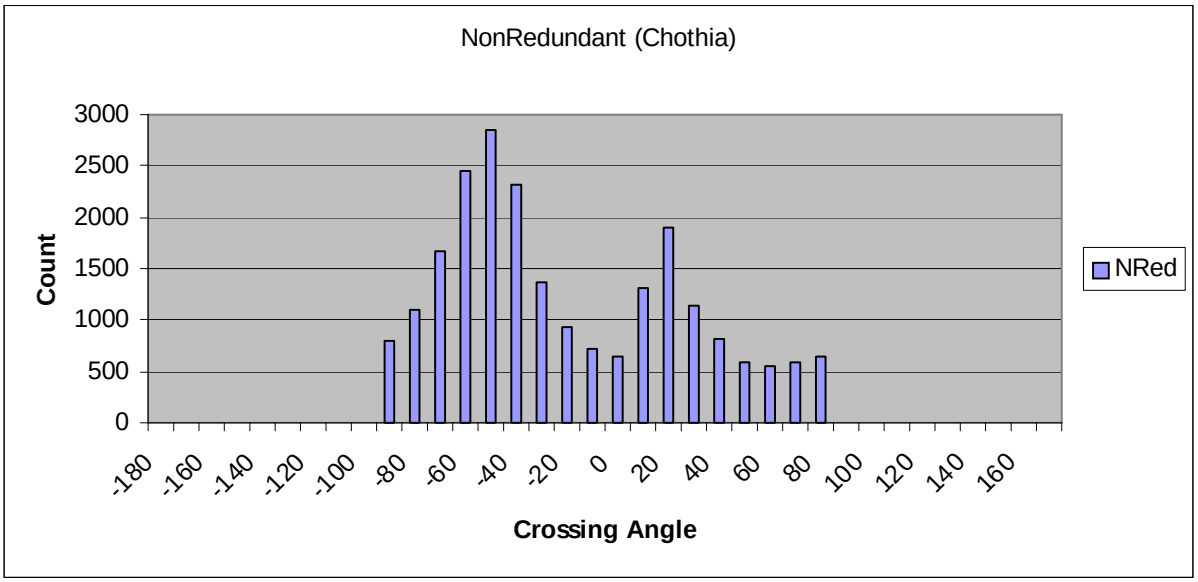

Figure 23. NonRedundantGeoData without Helix Directionality

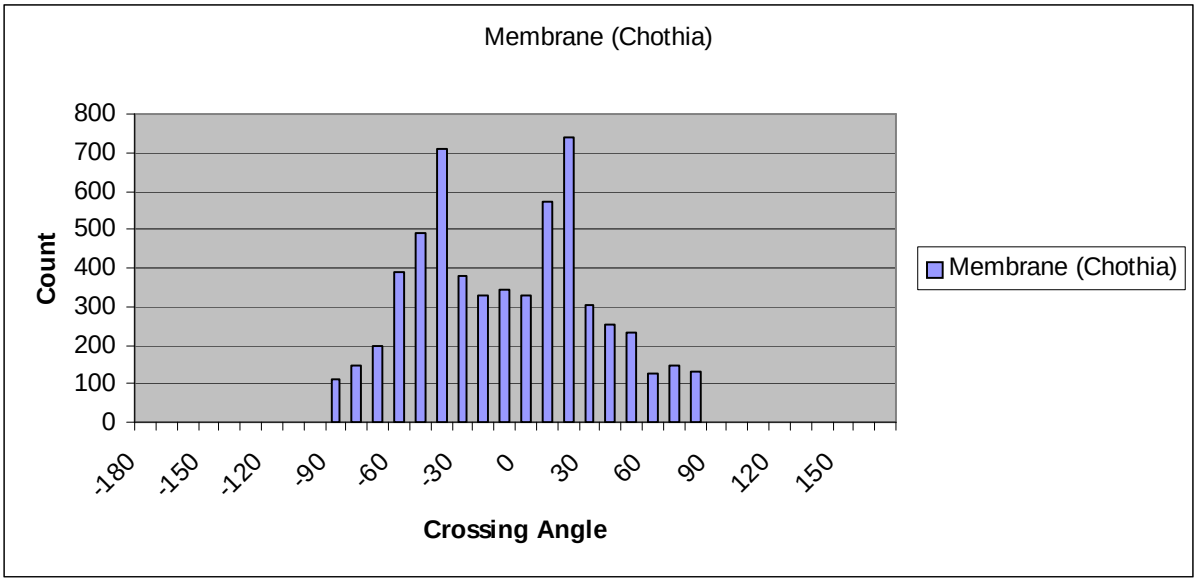

Figure 24. MembraneGeoData without Helix Directionality

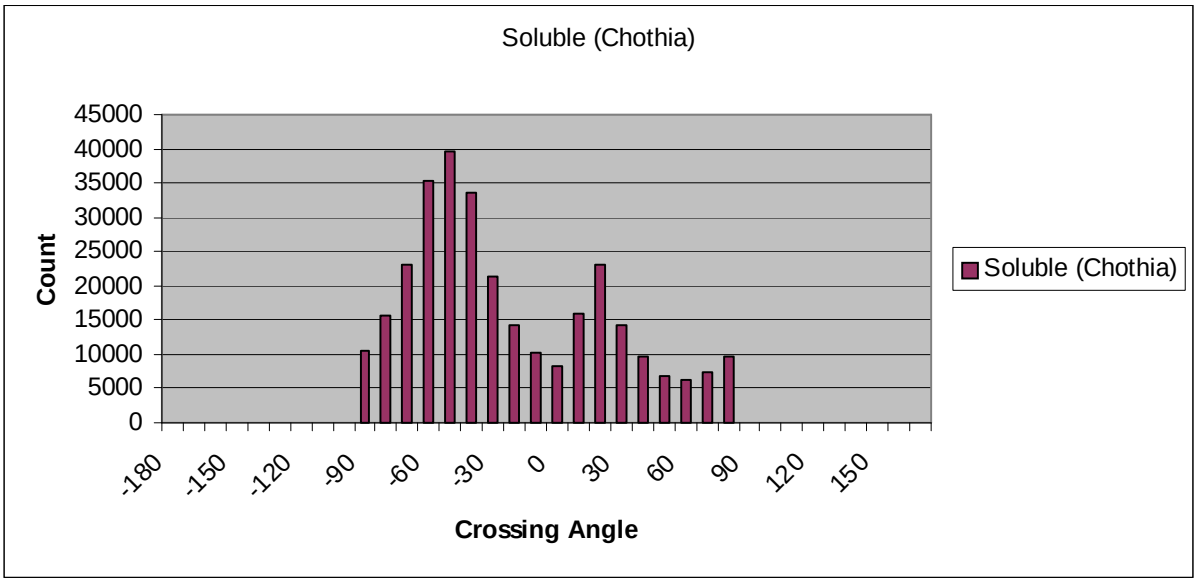

Figure 25. SolubleGeoData without Helix Directionality

4.10 Discussion

The geometric data gathered, processed and input into a database is the data from the 68,470 proteins culled from the PDB on October 11th 2010 that have at least two helices. We compared and analyzed our methods for calculating the crossing angle (Ω) between the interacting helical axes [21]. For the model presented by Chothia et al, we defined the angles between the interacting helices to lie between $-90° \leq \Omega < 90°$ ignoring the direction of individual helices. If the near helix is rotated in a clockwise direction relative to the far helix then the angle is negative between 0 to -90° and if the rotation is anticlockwise, then the angle is positive between 0 to 90° [21].

We analyzed the helix-helix interactions from all the tables we created using a crossing angle (Ω) range between $-180° \leq \Omega < 180°$ (with directionality) and $-90° \leq \Omega < 90°$ (no directionality) for interacting helix axes that are less than 10 Å (Angstroms) apart. The data values range from 0° to 10° for eve ry crossing angle (Ω). For instance,

considering **Figure 25** the data values at -50° are those helix-helix inter actions ranging between -50° to -40° and the values at 80° are thos e that range between 80° to 90°. We find that for the graphs representing the model by Chothia et al, there are peaks at angles -50°, 20°, and 80° with the highest point on these graphs at angle -50° for **Figures 21, 22, 23, 25**. The membrane graph in **Figure 24** has peaks at angles -40°, 20°, and 70° with the highest point at 20°. Our fin dings are fairly consistent with the findings of angles -66°, -32°, and 40° by Chothia e t al [21].

Using our method (with directionality) we find that there are peaks at angles 130°, 20°, and -60° for **Figures 16, 17, 18, 20** with the highest point at 130°. The membrane graph in **Figure 19** has peaks at angles 140°, 20°, and -40° with the h ighest point at 20°. Our findings are fairly consistent wi th the findings of Sangyoon et al that assert that helices have a preference to interact at angles of approximately ±160° ±20° [23]. Also Chothia et al in their 1981 paper on helix to helix packing predicted that the angles that helices tend to interact in are -52°, 2 3°, and 75° [24].

The values from **Figures 19, 24** the membrane graphs using both our method and that of Chothia et al show about a 10° to 20° d ifference in their values when compared with the data from **Figures 16, 17, 18, 20, 21, 21, 23, 25**. For instance, using the model by Chothia et al, the graphs in **Figures 21, 22, 23, 25** show peak angle values of -50°, 20°, and 80° while **Figure 24** the membrane graph, shows peak angle values of -40°, 20°, and 70°. However, Bowie 1997 a sserts that membrane proteins tend to interact mostly at angle 20° which is consi stent with the highest peak for both our method and the Chothia model for the membrane graphs as depicted in **Figures 19, 24**.

We have shown that our methods for calculation of geometric entities for helix-helix interactions are fairly consistent with the works from other studies, and with additional work, will allow us to generate physically reasonable helix-pairs in future protein modeling and design studies by querying this database and choosing fragments that conform to our geometric specifications.

Chapter 5: Overall Results

The goal of this thesis project was to create a database of secondary structure elements and geometric information to aid protein assembly, *de novo* protein design, prediction and analysis. We created a database with several tables towards the goal. All the tables as listed in 4.1 can be linked to each other if need be but more importantly, the database may be queried to produce results in seconds. **Figures 10, 13,** and **Tables 2, 3** show the attributes for the tables and the database statistics for all the tables. **Figure 26** illustrates the relationships between all the tables in the database. The tables share a zero or one to zero, one or many relationship.

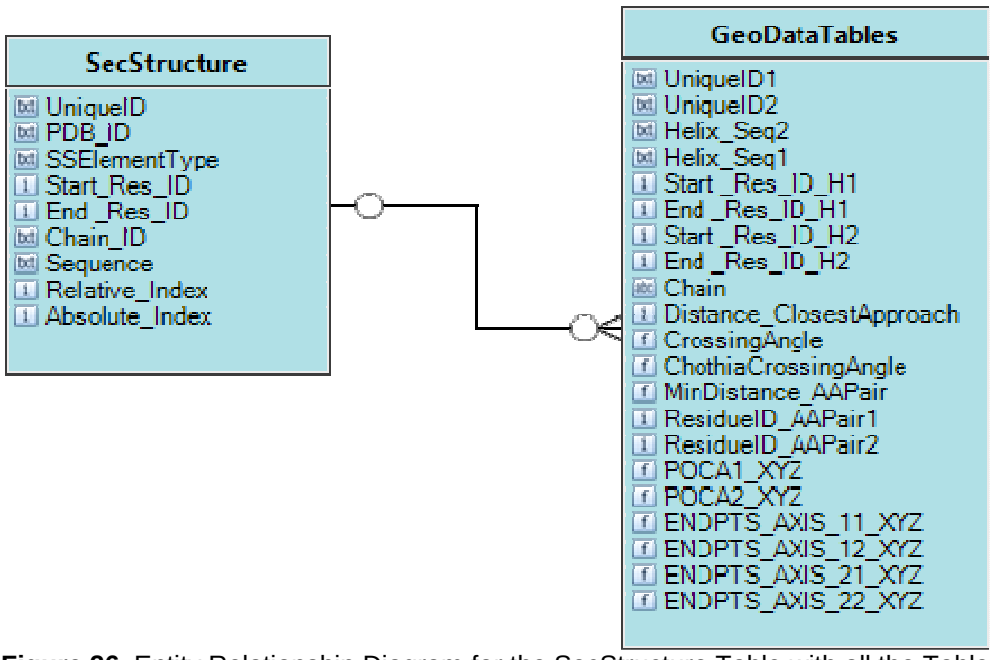

Figure 26. Entity Relationship Diagram for the SecStructure Table with all the Tables Holding Geometric Information

The information in the tables we created can be used to piece together 'new' proteins. Database approaches have the advantage of providing a global view of all the helix-helix interactions at a glance and make decisions on design, prediction, and

folding based on observations. For instance, the following queries request information about helical conformations of proteins as follows:

Query 1: All proteins that have Isoleucine (I) at the first position in the helix sequence of helix1 and an Isoleucine (I) at position 12 in the helix sequence of helix2.

SELECT UniqueID1, UniqueID2, Helix_Seq1, Helix_Seq2 FROM SecStrucGeoData WHERE instr(Helix_Seq1,'I')=1 && instr(Helix_Seq2,'I')=12;

Query 1 results in **Figure 27**

UniqueID1	UniqueID2	Helix_Seq1	Helix_Seq2
200l_2	200l_10	IFEMLRIDE	LNAAKSELDKAI
103l_2	103l_11	IFEMLRIDE	LDAAKSELDKAI
104l_2	104l_12	IFEMLRID	AAKSAAELDKAI
205l_2	205l_11	IFEMLRIDE	AKSAAAELDKAI
206l_2	206l_10	IFEMLRIDE	LNASKSELDKAI
107l_2	107l_10	IFEMLRIDE	LNAAKGELDKAI
109l_2	109l_10	IFEMLRIDE	LNAAKKELDKAI
209l_2	209l_10	IFEMLRIDE	LNAAKSELDKAI
110l_2	110l_10	IFEMLRIDE	LNAAKLELDKAI
210l_2	210l_10	IFEMLRIDE	LNAAKSELDKAI

Figure 27. Results for Query 1 (First 10 Records)

Query 2: All the records that have helix pairs with a sequence length of 14 or more and with a distance of closest approach less than 12 Å

SELECT UniqueID1, UniqueID2, Helix_Seq1, Helix_Seq2, Distance_ClosestApproach FROM SecStrucGeoData WHERE (char_length(Helix_Seq1)>=14 && char_length(Helix_Seq2)>=14) && Distance_ClosestApproach <12;

Query 2 Results in **Figure 28**

UniqueID1	UniqueID2	Helix_Seq1	Helix_Seq2	Distance_ClosestApproach
200l_14	200l_18	KDEAEKLFNQDVDAAVRGIL	AVRRAALINMVFQMGETGVA	7.96136
101m_2	101m_4	EGEWQLVLHVWAKV	VAGHGQDILIRLFKS	8.45717
101m_2	101m_10	EGEWQLVLHVWAKV	EDLKKHGVTVLTALGAIL	11.3824
101m_2	101m_14	EGEWQLVLHVWAKV	IKYLEFISEAIIHVLHSR	11.4458
101m_2	101m_18	EGEWQLVLHVWAKV	ADAQGAMNKALELFRKDIAAKYKE	8.62015
101m_4	101m_10	VAGHGQDILIRLFKS	EDLKKHGVTVLTALGAIL	7.46306
101m_4	101m_14	VAGHGQDILIRLFKS	IKYLEFISEAIIHVLHSR	8.90193
101m_14	101m_18	IKYLEFISEAIIHVLHSR	ADAQGAMNKALELFRKDIAAKYKE	10.3162
201l_14	201l_18	KDEAEKLFNQDVDAAVRGILR	AVRRAALINMVFQMGETGVA	7.59392
201l_43	201l_47	KDEAEKLFNQDVDAAVRGILR	AVRRAALINMVFQMGETGVA	7.63568

Figure 28. Results for Query 2 (First 10 Records)

Query 3: All non redundant proteins that have helix pairs with a sequence length of 14 or more and with a distance of closest approach less than 12 Å.

SELECT UniqueID1, UniqueID2, Helix_Seq1, Helix_Seq2, Distance_ClosestApproach FROM NonRedundantChainsGeoData WHERE (char_length(Helix_Seq1)>=14 && char_length(Helix_Seq2)>=14) && Distance_ClosestApproach <12;

Query 3 results in **Figure 29**

UniqueID1	UniqueID2	Helix_Seq1	Helix_Seq2	Distance_ClosestApproach
7odc_10	7odc_44	LGDILKKHLRWLKA	FEEITSVINPALDKY	10.1512
7odc_40	7odc_44	DTFVQAVSDARCVFDMATEV	FEEITSVINPALDKY	8.85141
2vke_2	2vke_10	RESVIDAALELLNET	KRALLDALAVEILARH	9.48701
2vke_2	2vke_12	RESVIDAALELLNET	WQSFLRNNAMSFRRALL	11.9664
2vke_10	2vke_12	KRALLDALAVEILARH	WQSFLRNNAMSFRRALL	8.7321

Figure 29. Results for Query 3 (First 5 Records)

Query 4: All non redundant proteins have Isoleucine (I) at the first position in the helix sequence of helix1 and an Isoleucine (I) at position 12 in the helix sequence of helix2.

```
SELECT UniqueID1, UniqueID2, Helix_Seq1, Helix_Seq2 FROM
NonRedundantChainsGeoData WHERE instr(Helix_Seq1,'I')=1 &&
instr(Helix_Seq2,'I')=12;
```

Query 4 Results in **Figure 30**

UniqueID1	UniqueID2	Helix_Seq1	Helix_Seq2
2bpo_69	2bpo_109	IADALKYLS	MAKGVSTALVGILSRG
1k0m_21	1k0m_27	IFAKFSAYIK	LADCNLLPKLHIVQVVCKKYR
1su8_24	1su8_100	IDHEIAEIMHR	PETAADKLLAAINERRAG
2q40_8	2q40_20	IPSFKKRAE	DGKRLENLVRGIFAGNI
3ctz_60	3ctz_72	ILSELKALCA	SAESEGMRRAHIKDAVALCELFNWLEKE
3i7a_10	3i7a_16	INSAVTRI	WEVMDEVWRTSIDVTAAACSLLQIYNKK
3i7a_12	3i7a_16	IKSIATSV	WEVMDEVWRTSIDVTAAACSLLQIYNKK
3gaa_23	3gaa_41	IYEISNTLMNWIDQV	LEEQVKALDEQIKKIEEQYKELQEK
3nyc_6	3nyc_34	IAGASTGYWL	TDALHQGYLRGIRR
3nyc_6	3nyc_68	IAGASTGYWL	SAAMGEASAALI

Figure 30. Results for Query 4 (First 10 Records)

Chapter 6: Conclusions & Future Work

6.1 Future Work

There are several ways through which this project could be expanded even further. Currently, the ways through which the tables in the database are created as discussed in chapters 2 and 3 could be further automated. We could increase access to our database by providing a presence on the web *via* a web site. The protCAD library's integration with the MySQL database could take a step further. This will allow protCAD to interact with the database and perform analysis using MySQL C++ hooks. Finally, the geometric data calculation, collection and analysis could be extended by adding to the already collected data for helices, other secondary structure (Beta Sheets, Coils etc).

6.2: Overall Conclusions

The analysis of the geometric relationships between secondary structure elements in proteins can be extremely useful to protein prediction, analysis, and *de novo* design. The queries shown in chapter 5 show how certain criteria of the geometric data can be used to provide a framework for designing a protein or for assembling a possible protein structure from known secondary structure fragments. Finally, our database produces response times to queries in seconds. Results to queries 1 to 4 are depicted in **Table 4** below.

Table	Query No	Number of Rows	Response Time(sec)
SecStrucGeoData	1	3769	2.42
SecStrucGeoData	1	10	0.00
SecStrucGeoData	2	187361	2.78
SecStrucGeoData	2	10	0.00
NonRedundantGeoData	3	13683	0.20
NonRedundantGeoData	3	5	0.00
NonRedundantGeoData	4	204	0.17
NonRedundantGeoData	4	10	0.1

Table 4. Query Statistics and Response Times

References

[1] Computational Structural Biology. Computer Science Division [Internet]. [modified 1999 Aug 2; cited 2010 Oct 13]. Available from: http://www.cs.unc.edu/Research/csbr/

[2] Department of Chemistry University of Calgary [Internet]. Calgary (CA): Carboxylic Acid Derivatives; [cited 2010 Oct 13]. Available from: http://www.chem.ucalgary.ca/courses/350/Carey/Ch20/ch20-1-1.html#structure

[3] Department of Biochemistry and Molecular Biophysics University of Arizona [Internet]. Sierra Vista (AZ): The Basic Structure of an Amino Acid; c2003 [modified 2003 Aug 23; cited 2010 Oct 13]. Available from: http://www.biology.arizona.edu/biochemistry/problem_sets/aa/BasicStruct.html

[4] Lodish H, Berk A, Zipursky SL, Matsudaira P, Baltimore D, and Darnell J. Molecular Cell Biology: Hierarchical Structure of Proteins [Internet]. New York (NY): W.H. Freeman and Company; 4th ed. c2000 [cited 2010 Oct 18]; [about 2 screens]. Available from: http://www.ncbi.nlm.nih.gov/bookshelf/br.fcgi?book=mcb&part=A517

[5] Kimball's Biology Pages [Internet]. Primary Structure; c2010 [modified 2006 Mar 24; cited 2010 Oct 18]. Available from: http://users.rcn.com/jkimball.ma.ultranet/BiologyPages/P/PrimaryStructure.html

[6] Anfinsen CB. 1973. Principles that Govern the Folding of Protein Chains. Science. [Internet]. 181(4096): 223-230 [cited 2010 Oct 19]. Available from: http://www.jstor.org/stable/1736447

[7] Onuchic JN, Wolynes PG. Theory of Protein Folding. Structural Biology. 2004; 14. 70-75.

[8] RCSB Protein Data Bank. Methods for Determining Atomic Structures [Internet]. [modified 2010 Oct 12; cited 2010 Oct 13]. Available from: http://www.pdb.org/pdb/static.do?p=education_discussion/Looking-at-Structures/methods.html

[9] Summa CM, Levitt M. 2007. Near-native structure refinement using in vacuo energy minimization. PNAS. 104(9): 1-7. [cited 2010 Oct 19]. Available from: http://www.cs.uno.edu/~csumma/pubs.html/ doi: 10.1073/pnas.0611593104

[10] Zhang Y. 2009. Protein Structure Prediction: Is It Useful? PubMed Central. [Internet]. 19(2): 1-17[cited 2010 Oct 19]. Available from: http://www.ncbi.nlm.nih.gov/pmc/articles/PMC2673339/ doi: 10.1016/j.sbi.2009.02.005

[11] DeGrado WF, Summa CM, Pavone V, Nastri F, Lombardi A. 1999. De Novo Design and Structural Characterization of Proteins and Metalloproteins. Annual Review

of Biochemistry. [Internet]. 68: 779-819. [cited 2010 Oct 20]. Available from: http://www.cs.uno.edu/~csumma/pubs.html

[12] Summa CM. 2002. Computational Methods and Their Applications to De Novo Functional Protein Design and Membrane Protein Solubilization. [Dissertion]. [Philadelphia (PA)]: University of Pennsylvania 494 p. [cited 2010 Oct 13].

[13] RCSB Protein Data Bank. About the PDB Archive and the RCSB PDB [Internet]. [modified 2010 Oct 19; cited 2010 Oct 20]. Available from: http://www.pdb.org/pdb/static.do?p=general_information/about_pdb/index.html

[14] RCSB Protein Data Bank. Understanding PDB Data: Looking at Structures [Internet]. [modified 2010 Oct 19; cited 2010 Oct 20]. Available from: http://www.pdb.org/pdb/static.do?p=education_discussion/Looking-at-Structures/intro.html

[15] CMBI. Dssp [Internet]. [cited 2010 Oct 20]. Available from: http://swift.cmbi.ru.nl/gv/dssp/

[16] Kabsch W, Sander C. Dictionary of Protein Secondary Structure: Pattern Recognition of Hydrogen-Bonded and Geometrical Features. Biopolymers. 1983; 22 2577-2637

[17] MySQL. About MySQL. [cited 2010 Oct 20]. Available from: http://www.mySQL.com/about/

[18] MySQL++ Homepage. [cited 2010 Oct 20]. Available from: http://tangentsoft.net/mySQL++/

[19] CATH Database. CATH FAQ [Internet]. [modified 2010 Oct 07; cited 2010 Nov 2]. Available from: http://www.cathdb.info/wiki/doku.php?id=faq

[20] The Stephen White Laboratory at UC Irvine. Membrane Proteins of Know 3D Structure [Internet]. c2010 [modified 2010 Oct 31; cited 2010 Nov 2]. Available from: http://blanco.biomol.uci.edu/Membrane_Proteins_xtal.html

[21] Chothia C, Levitt M, Richardson D. Structure of Proteins: Packing of Alpha-Helices and Pleated Sheets. Proc. Natl. Acad. Sc. 1977; 74(10): 4130-4134

[22] G. Wang, R. L. Dunbrack, Jr. PISCES: A Protein Sequence Culling Server. Bioinformatics. 2003; 19:1589-1591.

[23] Lee S, Chirikjian GS. Interhelical Angle and Distance Preferences in Globular Proteins. Biophysical. 2004; 86 1105-1117

[24] Chothia C, Levitt M, Richardson D. Helix to Helix Packing in Proteins. J. Mol. Biol. 1981; 145 215-250

[25] Bowie JU. Helix Packing in Membrane Proteins. J. Mol. Biol. 1997; 272 780-789

Vita

Augustine Ada Orgah was born in Kaduna, Nigeria. Upon obtaining his Bachelor of Science degree in Computer Science from Xavier University of Louisiana in 2008, he joined the graduate program at the University of New Orleans to purse a Master of Science in Computer Science as a member of Dr Christopher Summa's research group.