View metadata, citation and similar papers at core.ac.uk
brought to you by CORE
provided by Sabanci University Research Database

# VIRTUAL REALITY BASED APPROACH TO PROTEIN-PROTEIN DOCKING PROBLEM

by

SERDAR ÇAKICI

Submitted to the Graduate School of Engineering and Natural Sciences
in partial fulfillment of
the requirements for the degree of
Master of Science

Sabancı University
Spring 2009

VIRTUAL REALITY BASED APPROACH TO PROTEIN-PROTEIN DOCKING PROBLEM

APPROVED BY:

Assist. Prof. Selim BALCISOY …….

(Thesis Supervisor)

Assoc. Prof. Uğur SEZERMAN …….

(Thesis Co-Supervisor)

Assist. Prof. Serkan APAYDIN …….

Assist. Prof. Yücel SAYGIN …….

Assist. Prof. Hüsnü YENİGÜN …….

DATE OF APPROVAL: …….

© SERDAR ÇAKICI 2009

All Rights Reserved

VIRTUAL REALITY BASED APPROACH TO PROTEIN-PROTEIN DOCKING PROBLEM

Serdar ÇAKICI

EECS, MSc Thesis, 2009

Thesis Supervisor: Assist. Prof. Selim BALCISOY

Thesis Co-Supervisor: Assoc. Prof. Uğur SEZERMAN

Keywords: Molecular Docking, Protein-Protein Docking, Virtual Reality

Abstract

Proteins are large molecules that are vital for all living organisms and they are essential components of many industrial products. Protein-protein docking is the evaluation of binding of a protein to another via computer simulations. Many automated algorithms have been proposed to find docking configurations that might yield promising protein-protein complexes. However, these automated methods are likely to come up with false positives and have high computational costs. Consequently, Virtual Reality has been used to take advantage of user's experience on the problem. Haptic devices have been used for molecular docking problems; but they are inappropriate for protein-protein docking due to their workspace limitations and lack of sufficient information from force feedback. Instead of haptic rendering of forces, we provide two novel visual feedback methods for simulating physicochemical forces of proteins. We propose an interactive 3D application, DockPro, which enables domain experts to come up with dockings of protein-protein couples by using magnetic trackers and gloves in front of a large display.

# PROTEİN-PROTEİN KENETLENMESİ PROBLEMİNE SANAL GERÇEKLİK TABANLI YAKLAŞIM

Serdar Çakıcı

EECS, Yüksek Lisans Tezi, 2009

Tez Danışmanı: Yar. Doç. Selim BALCISOY
Yardımcı Tez Danışmanı: Doç. Uğur SEZERMAN

Anahtar Sözcükler: Moleküler Kenetlenme, Protein-Protein Kenetlenmesi, Sanal Gerçeklik

## Özet

Proteinler tüm canlı organizmalar için yaşamsal önem taşıyan büyük moleküllerdir ve birçok endüstriyel ürüne temel teşkil ederler. Protein-protein kenetlenmesi, bir proteinin diğerine bilgisayar simülasyonları aracılığıyla ekleştirilmelerinin test edilmesidir. Şimdiye kadar kenetlenme pozisyonlarını bulma üzerine pek çok otomatik kenetlenme algoritması geliştirilmiştir; fakat bu metodların yanlış pozitif sonuçlar bulmaları olasıdır ve hesaplama süreleri uzundur. Bunu takiben, kullanıcının deneyimlerinden faydalanmayı sağlamak adına Sanal Gerçeklik problem üzerinde kullanılmıştır. Benzer şekilde moleküler kenetlenme problemi dahilinde haptik aletler de denenmiştir. Fakat protein-protein kenetlenmesi probleminde haptik aletler çalışma alanlarının sınırlılığı ve güç geribeslemesinin yetersiz bilgi vermesinden dolayı uygunsuzdur. Tezde, güçleri haptik olarak gerçekleştirmek yerine, proteinlerin fizikokimyasal güçlerini benzeştirmek için iki yeni görsel geribesleme metodu geliştirdik. DockPro adındaki etkileşimli 3-Boyutlu uygulamamız, konu üzerindeki uzmanların büyük ekran karşısına geçip manyetik takip cihazları ve eldivenler kullanarak protein-protein çiftlerini kenetlemelerini sağlamaktadır.

Canım aileme

## ACKNOWLEDGMENTS

I would like to express my deepest gratitude to my thesis supervisor Assist. Prof. Selim Balcısoy and my thesis-cosupervisor Assoc. Prof. Uğur Sezerman for their belief in me, their efforts to lead me to higher and better, and their caring attitude throughout my long journey in Sabancı University. I will always remain grateful to them.

I am also thankful to my thesis committee members Assist. Prof. Hüsnü Yenigün, Assist. Prof. Serkan Apaydın, and Assist. Prof. Yücel Saygın for their valuable comments.

This thesis was partially supported by TÜBITAK project 106E213.

Dr. Alper Küçükural was the person who gave me the idea of my thesis's topic; thank you so much Alper. I am deeply grateful to him for his helpful and friendly manners and for bearing with me during my endless question sessions.

Selçuk Sümengen came up with the idea of "dynamic color coding". He also wrote the codes related to speeding up the system and creating tangent continuous surfaces. "3D rose glyph" idea was given by Assist. Prof. Selim Balcısoy. Assist. Prof. Serkan Apaydın gave the idea of using empirical look-up tables for simulating interaction forces between proteins. I really appreciate their supports; this thesis would not be meaningful without their ideas and efforts.

Special thanks to Ceren Kayalar and Elvin Erkut for the lovely "3D rose glyph" illustration.

Many thanks to Assist. Prof. Lucio De Paolis and his family (i.e. Silke, Alex, Sophie) for their caring and understanding attitude during my stay in Lecce, Italy. Big hugs to Lucio's students, my dear friends, Alessio Agrimi, Alessandro Zocco, Marco Pulimeno, and their families for their friendship; I was so glad to spend one whole month with them. I consider Lecce my second home.

I had two unbelievably beautiful years during working on my thesis. I owe these good times to all the CGLab members, my dear friends: Ceren Kayalar, Sina Çetin, Çağatay Turkay, Kaan Yanç, Ahmet Sülek, Berkay Kaya , Emre Koç, Işıl Demir, Merve Çaylı, Uraz Türker, İsmail Kasarcı, Billur Engin, Can Özmen, Selçuk Sümengen, Tolga Eren. I consider myself so lucky to have all of these wonderful people around me.

Thanks to all of my friends and professors that turned my experience in Sabancı to an unforgettably marvelous one.

I am so happy to have spent so many beautiful days with my aunt Asiye Şahin and my cousins Armağan Şahin and Ataman Şahin in Istanbul. They opened their doors and hearts to me. I did not feel homesick not even one time when I was with them. You are such great people.

Well, I kept the most important part to the most important people: My family. I do not know how I can thank my mother, Kumru Çakıcı, for everything she has done for me and our family. Actually, I already know I can never thank her enough. She was always there with me by my side: Sharing my happiness, getting me back on my feet whenever I felt down ("Önce sağlık"), motivating me and believing in me more than I did. I even owe her finishing writing my thesis on time and having the courage to move on to get a PhD degree. I love you mom… I think I've never let you down; and knowing this is the biggest gift I can ever have in my life.

My dad, Dr. Lütfi Çakıcı, was always understanding and helpful, no matter what situation I was in. It is such soothing and heartening to hear you say "Yaparsın koçum benim", and I know that you always really meant it. I love you dad.

My brother, my idol, my best friend, Sertaç Çakıcı. I always tried to follow his footsteps, knowing that whatever he does is good and right. He always did his best to make me happy, to show me the way, to share all the things that one can think of, and to be always by my side. "Canım abicim", I love you so much.

Thanks again to my family for being the meaning of my life. I feel like I can crush mountains with a finger with their support and -most importantly- love.

## TABLE OF CONTENTS

## LIST OF FIGURES

## LIST OF TABLES

## LIST OF ABBREVIATIONS

VR : Virtual Reality

DNA : Deoxyribonucleic acid

RNA : Ribonucleic acid

PDB : Protein Data Bank

Å : Ångstrom

API : Application Programming Interface

NMR : Nuclear Magnetic Resonance

RMSD : Root-Mean-Square Distance

VDW : Van Der Waals

SAS : Solvent Accessible Surface

2D : Two dimensional

3D : Three dimensional

DOF : Degree of Freedom

GPU : Graphics Processing Unit

# Chapter 1

# INTRODUCTION

Proteins are organic compounds that are essential for proper functioning of the body as a whole. They take place in every action in the metabolism. Proteins are made of building blocks called amino acids. An amino acid (alpha-amino acid, to be exact) is a molecule which consists of several specific parts, namely: an α-carbon, a carboxyl group, an amino group, a hydrogen atom, and a side chain. Side chain (or R-group) varies from amino acid to amino acid. Actually, variations of side chains are the causes of variations of amino acids. There are 20 different standard amino acid types.

A protein's functions are defined related to which other protein(s) it interacts. One has to understand protein-protein interactions in order to understand all kinds of cellular events. The question of how proteins bind to other proteins is a hot topic since the problem of protein-protein interaction is at the heart of many different industrial products, such as biofuel industry, starch industry, and detergent industry.

Docking is the process in which at least two molecules bind to each other, in a specific position and orientation, and create a molecular complex; afterwards, results are evaluated. Knowing bound configurations of interacting proteins requires protein-protein docking, which enables us to understand:

- How two proteins interact with each other,
- Spatial configurations of possible protein-protein complexes,
- Specific properties of interactions on the surface (i.e. protein interface), where binding takes place.

It is important to have a stable concatenation of proteins in order to have a successful docking. In the process of protein-protein docking, two aspects should be taken into account: Physicochemical properties of proteins, and their shape complementarity. A protein can have different physicochemical characteristics on different surface regions (e.g. one region is attracted by water molecules, while another is not). Shape complementarity should also be considered since proteins have curved surfaces containing large number of cavities and knobs.

Several algorithms have been proposed for protein-protein docking. Fully automated applications are first introduced in early 1990s [15]. Depending on the complexity of the input proteins, docking process can take up to several hours, and may compute false positives.

Virtual Reality (VR) has been used to take advantage of an expert's domain knowledge and experience. Several VR tools on attacking docking problem have been proposed. However, each of them has serious usability drawbacks [11].

We propose an easy-to-use application, DockPro (Figure 1.1), addressing the two issues of protein-protein docking: i) Shape complementarity. DockPro employs direct manipulation interaction technique, allowing a biologist to explore possible spatial configurations in real time. ii) Physicochemical properties. Our contributions are two novel visual feedback methods (i.e. "dynamic color coding" and "3D rose glyphs") for simulating forces on proteins. Unlike fully automated systems that find relatively successful configurations of protein-protein couples in hours, similar results can be obtained in minutes with DockPro.

Since our application provides the means for figuring out the mechanism of protein-protein docking process, it can be used for i) educational purposes, ii) manufacturing industrial products, and iii) drug design.

There is a large class of proteins that governs important roles in many industrial areas. These proteins are called enzymes. Some areas that make use of enzymes are: Photographic industry, biofuel industry, starch industry, detergent industry. Our application is designed to be used during the course of protein engineering. During a protein design process, the protein's functionality can be evaluated via DockPro.

Drug design relates to ligand (small molecule)-protein docking. Since the main principles of protein-protein docking and ligand-protein docking are the same, our application can also be used for ligand-protein docking.

In Chapter 2, biological background that is necessary to understand the problem and its components is provided. In Chapter 3, an outline of previous algorithms and applications is given. In Chapter 4, we dwell on the scientific visualization methods we have developed. In Chapter 5, we present our application, DockPro; which also contains implementations of our novel scientific visualization methods of dynamic color coding and 3D rose glyphs. Finally in Chapter 6, we discuss force representation and visual feedback. Afterwards, we mention future work and then conclude our thesis.

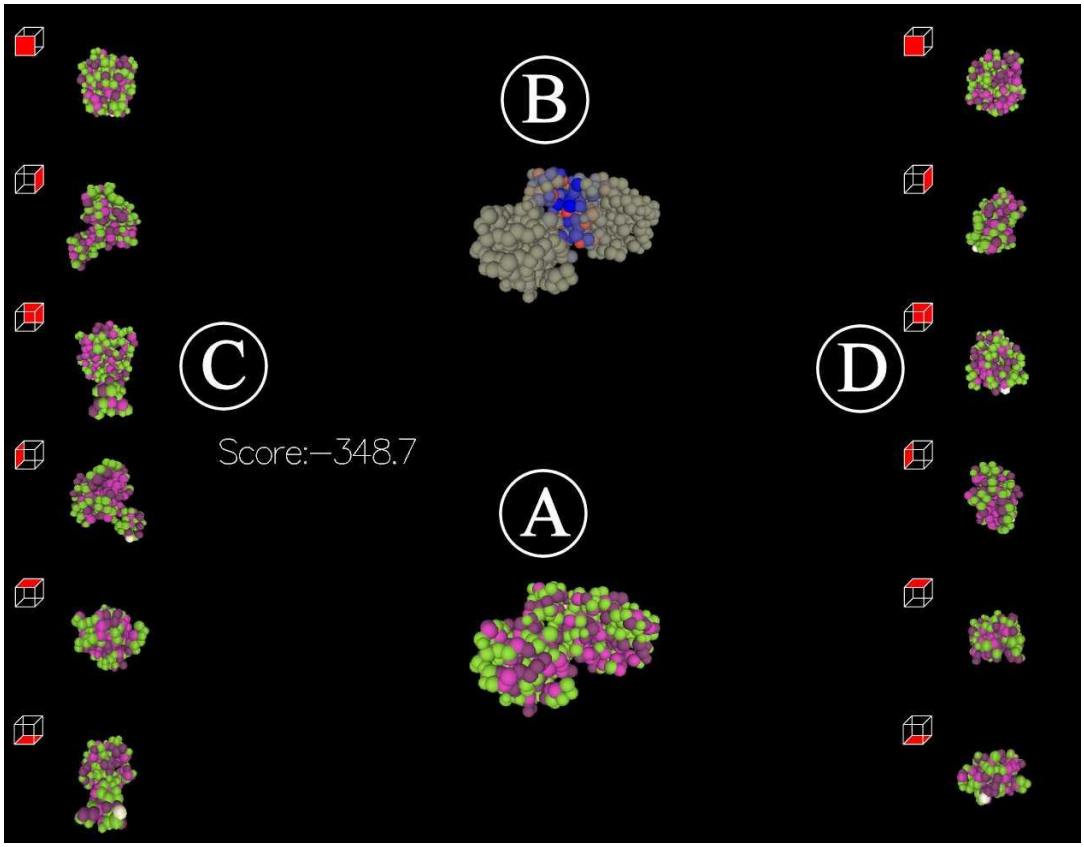


Figure 1.1 : DockPro's main window. a) Regular docking view. b) Scientific visualization of docking via dynamic color coding method. c) Dynamic legend for the protein on the left. d) Dynamic legend for the protein on the right.

# Chapter 2

# BIOLOGICAL BACKGROUND

To have a complete understanding of protein-protein docking problem, the reader should have prior knowledge about aspects that resides in the problem definition. In the first part of this chapter, we are providing some background information on proteins: What they are made of, what they are good for, which different regions they have, and how the data they contain are represented. In the second part, we concentrate on docking by building on top the information coming from the first part.

## 2.1 An Overview of Proteins

### 2.1.1 General Information

Proteins are organic matters that are of primary importance in any cell function. They can be thought as operators carrying out the tasks specified by genes. Proteins are categorized into groups according to the functions they accomplish. Some of these groups are as follows:

- Enzymes (catalyze chemical reactions)
- Antibodies (disrupt functioning of foreign molecules)
- Structural proteins (give rigidity to biological components)
- Motor proteins (transform chemical energy to mechanical energy)
- Hormones (lead to effects in cells at different body parts)

Proteins can pursue their functions by binding to other proteins or small molecules. These bindings are accomplished under specific conformations at specific locations of actors. That is, a protein binds to another molecule by contacting it from a specific region and by being in a necessary conformation.

Sequence of a molecule determines its structure; and structure of a molecule determines its function. This fact goes also with proteins. A protein is a large molecule organized as a linear chain consisting of repeating units called amino acids. Throughout this linear chain, amino acids are connected with specific types of chemical bonds called peptide bonds. Although amino acids are joined in a linear fashion, a protein should not be conceptualized like a smooth rope. In nature, a protein resides in a folded state. This folded state is not an arbitrary confirmation; it is determined by the amino acid sequence it is made of [Figure 2.1].

The information about any protein lies in DNA (in some cases, RNA). Gene, the part of DNA which contains information about a protein, consists of "codon"s. A codon is a sequence of three nucleotides (nucleotide is the basic unit of DNA). Each codon contains information of a specific amino acid. That is, this information is transcribed to a special type of RNA (messenger RNA), and information on RNA is then translated on a cellular complex (ribosome) to an amino acid.

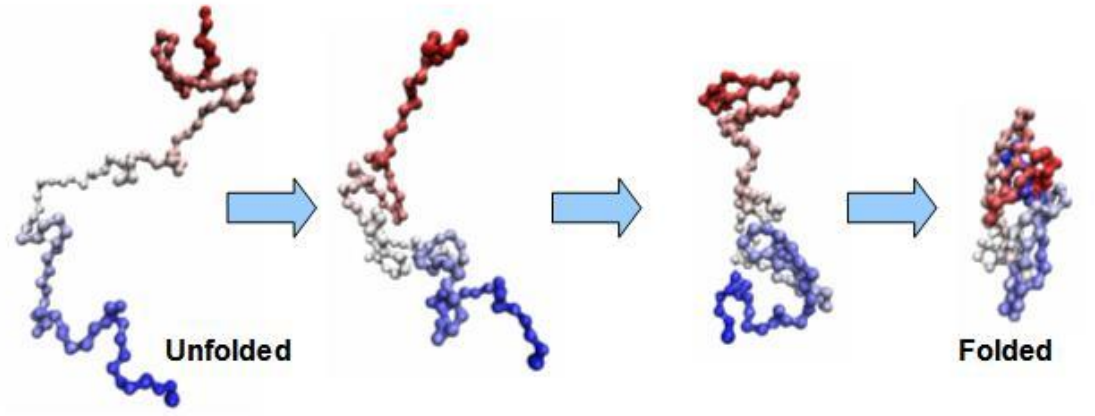


Figure 2.1 : Unfolded and folded states of a protein [8]

Amino acids, the building blocks of proteins, are small molecules that consist of the following parts:

- a central alpha-carbon ($C_\alpha$)
- a hydrogen atom (H)
- a carboxyl group (COOH)
- an amino group ($NH_2$)
- a side chain (R)

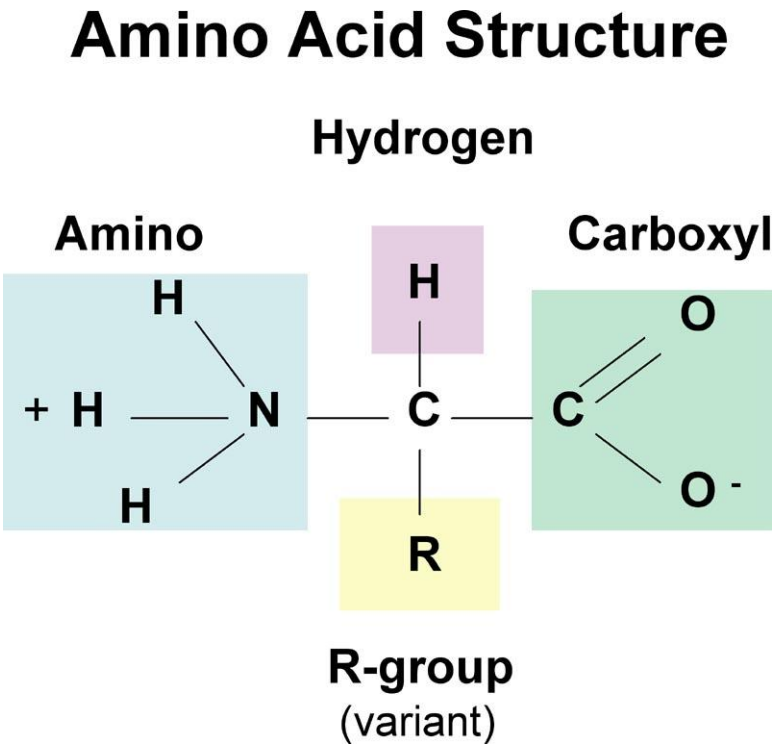


Figure 2.2 : Amino acid structure [24]

The part that gives rise to different amino acids and hence different functions is the side chain part (i.e. R group). There are 20 standard amino acids in nature, and these amino acid types differ from one another just by their different side chains; all the other parts amino acids have are common to all. As a matter of fact, amino acids are named by the side chains they carry.

Amino acids in a protein molecule are connected end-to-end in a linear fashion. A peptide bond between two neighboring amino acids is formed by a reaction between the carboxyl group of one amino acid and the amino group of the other. As a result of this reaction, a water molecule is formed and amino acids are connected. If an amino acid is a part of a protein (i.e. it resides in the linear chain), then it is called as a "residue".

Due to the amino acids' chemical characteristics and their distinct sequence, the linear chain folds into the specific three dimensional structure of a given protein. There are several categorizations of amino acids according to their chemical properties. The following table shows one of the most common categorizations:

| CHARGED | HYDROPHOBIC | POLAR | OTHER |
|---|---|---|---|
| Arginine (Arg ; R) | Alanine (Ala ; A) | Asparagine (Asn ; N) | Glycine (Gly ; G) |
| Aspartic Acid (Asp ; D) | Isoleucine (Ile ; I) | Cysteine (Cys ; C) | |
| Glutamic Acid (Glu ; E) | Leucine (Leu ; L) | Glutamine (Gln ; Q) | |
| Lysine (Lys ; K) | Methionine (Met ; M) | Histidine (His ; H) | |
| | Phenylalanine (Phe ; F) | Serine (Ser ; S) | |
| | Proline (Pro ; P) | Threonine (Thr ; T) | |
| | Valine (Val ; V) | Tryptophan (Trp ; W) | |
| | | Tyrosine (Tyr ; Y) | |

Table 2.1 : A possible grouping of amino acids

In the table above, we can see the grouping of all 20 standard amino acids. There are four groups: “charged”, “hydrophobic”, “polar”, and “other”. Charged residues are the ones that have positively or negatively charged side chains. Hydrophobic residues are nonpolar molecules which have side chains that do not want to be near water molecules. Polar residues have side chains that are hydrophilic (i.e. water loving). Amino acid glycine has different properties than the amino acids in other three groups; so it is put into a group on its own. Each block (other than the ones with group names) contains the name of the aminoacid, its three letter code, and its one letter code. For example, arginine is an amino acid with a charged site chain; its three letter code is Arg, and its one letter code is R.

Proteins, since many of them reside in aqueous environments, tend to fold such as hydrophobic regions have minimum contact with water, whereas hydrophilic regions have maximum contact with water. Thus water loving/hating is an important agent for folding. To minimize contact with water, residues with hydrophobic side chains take place at the core of the protein. Residues with polar side chains like to contact with water. The same goes for residues with charged side chains: They like to contact with ions. Therefore, polar and charged residues take place on the surface of the protein.

There are also other participants at folding other than hydrophobicity. These are:

- Hydrogen bonding: Residues with hydrophobic chains are at the core of proteins. However, backbones (part of residue excluding side chain) of proteins are polar. To cancel out polarity of backbones residing at the core, these backbones tend to create hydrogen bonds with one another. It goes the same for polar residues at the core of protein.
- Salt bridges: Charged residues that are at the core of a protein tend to couple with oppositely charged residues (i.e. negative with positive, vice versa) to cancel out their charges. The resulting chemical interactions are called “salt bridges”.
- Disulfide bridges: Adjacent cysteine residues may form disulfide bridges. However, it is not the case that adjacent cysteine residues should always create disulfide bridges. These bridges are generally found at proteins that do not reside in cells (i.e. extracellular proteins).

There are four different levels of a protein's structure:

- Primary structure : It is the structure which solely gives information about the amino acid sequence (i.e. the ordering of amino acids along the polypeptide chain).
- Secondary structure : It tells where local three dimensional structures are in a protein. Some popular examples for local structures are "β sheet" and "α helix".
- Tertiary structure : It gives all the information related to a given protein chain. Apart from the information that comes from primary and secondary structures, tertiary structure tells where each secondary structure unit stands with respect to other units in space. Thus, it gives the global three dimensional structure of the amino acid chain. Tertiary structure of a protein is also used as the synonym for "fold".
- Quaternary structure : There are lots of proteins which do not function as a single chain. These proteins have subunits (chains) which come together to create a multi subunit complex. Beside the structures mentioned above, quaternary structure tells about the arrangement of these subunits. Subunits can be identical or different from each other. Eventually, quaternary structure of a protein gives the information related to the associations of these multiple subunits.

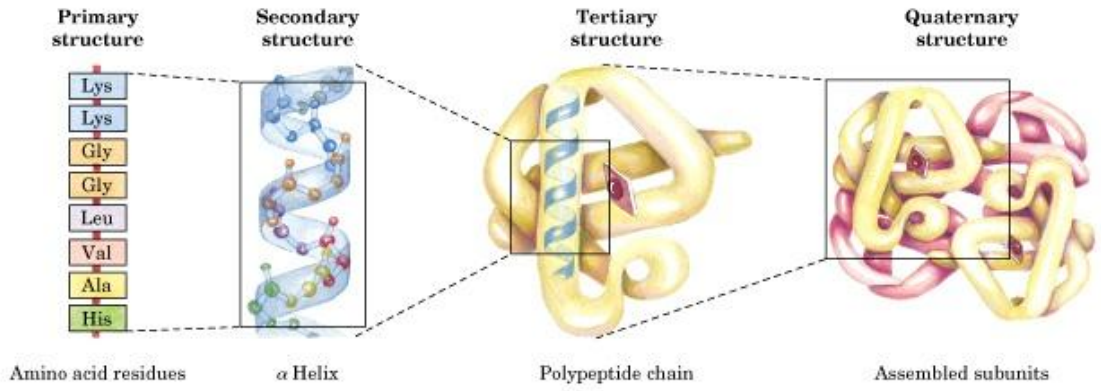


Figure 2.3 : Different structure types of a protein. From left to right: Primary structure, secondary structure, tertiary structure, and quaternary structure [26]

### 2.1.2 Representation

Protein Data Bank (PDB) [3] is a free-of-charge database of large biological molecules that have been found, solved, and documented. Any information related to a protein of which structure has been solved by experimental methods can be found in PDB. The database is updated weekly. As of May 12, 2009, there are data about 53188 proteins, 2012 nucleic acids, and 2325 protein/nucleic acid complexes.

Apart from the fact that it contains all the information on any known proteins, PDB also presents this information in a solid and easy-to-understand way. PDB file format is used for the task. This file format is a text file format that consists of format records, which contains data like details on how a given structure is determined, its structural features, and its atomic coordinates. Each structure is denoted by a PDB ID, which is a four-character alphanumeric identifier.

The PDB file format records that we are interested in this research are the ATOM records. Atomic coordinates of a given protein are found in these records. Below is an extract from the PDB text file of 1BRC:

```
ATOM     14  CG1 VAL E  17      61.463 -64.985 -10.037  1.00  3.21           C
ATOM     15  CG2 VAL E  17      62.954 -64.559 -11.989  1.00  7.93           C
ATOM     16  N   GLY E  18      60.259 -61.946  -9.099  1.00 14.67           N
ATOM     17  CA  GLY E  18      59.740 -61.645  -7.774  1.00 16.33           C
```

Figure 2.4 : An extract from the PDB file of 1BRC

In order to explain the ATOM format, let us take the last line of the extract into account. The line actually says that atom with serial number 17 and residue sequence number 18 is the carbon-alpha atom of a glycine residue of chain E, and its orthogonal coordinates in Ångstroms are 59.740(x axis), -61.645(y axis), -7.774(z axis). The remaining part of the line is not important for our purposes.

## 2.2 Protein-Molecule Docking

In nature, it is quite common that proteins interact with other proteins or small molecules to create specific impacts. Some of the protein groups have been mentioned in Section 2.1. Out of those groups, we can give enzymes as examples to these protein-molecule interactions. Enzymes are proteins which catalyze (increase the rates of) chemical reactions (not all enzymes are proteins, to be exact). They accomplish their tasks by binding to other molecules (substrates) in specific conformations.

It is a difficult task to realize and understand protein-molecule binding by regular experiments. Stemming from this fact, computational algorithms have been developed to simulate and predict protein-molecule binding via computers. For this task, we should first have three-dimensional structural information of the molecules we want to investigate. Well-known and most used techniques for structure determination are NMR spectroscopy and X-ray crystallography. Afterwards, we can go to the "docking" phase. Docking is a method by which bindings can be evaluated via computer simulations. During docking process, different conformations are tried to find the best fit. By best fit, we mean that the resulting protein-molecule complex has the lowest free energy.

### 2.2.1 Decisive Aspects

There are two main aspects in docking that are to be considered. One aspect is shape complementarity. In order to have successful docking, molecules should be positioned in appropriate ways. The other aspect is related to physicochemical properties. Even if a given geometry of the docking conformation may seem feasible, if the molecules do not want to be at that conformation because of emanating forces, then we cannot call such a conformation a good docking. Therefore, these two aspects have to go hand in hand during docking process.

### 2.2.2 Forces

In the molecular level, there are several different forces that play important roles during binding of molecules. Most important ones can be classified as follows:

- Bonded (intramolecular forces)
    - Bond stretching
    - Angle bending
    - Torsional
- Non-bonded (intermolecular forces)
    - Electrostatic force: It is the force that is present between electric charges.
    - Van der Waals force: It is a force that can be attractive and repulsive; it emanates from fluctuating polarizations of particles.
    - Hydrogen Bonds: It is an attractive force between a hydrogen atom (bonded to an electronegative atom) and an electronegative atom.

### 2.2.3 Search Algorithms

There are many automated algorithms for searching favorable conformations of molecules. Although simulating molecular dynamics would be the most accurate way for the purpose of docking, due to its high complexity, algorithms based on approximations are preferred. In most fully automated applications, genetic algorithms and Monte Carlo method are used.

### 2.2.4 Scoring

Scoring functions are used to evaluate the feasibility of dockings. We can roughly classify scoring functions into three groups:

- Molecular mechanics force field: It contains the intramolecular and intermolecular forces that were mentioned above. It is the best way in terms of accuracy, and the worst way in terms of its high complexity.
- Knowledge-Based Potentials: They are based on interatomic contact preferences between atoms.
- Empirical methods: Empirical scoring functions are derived by calculating related parameters from training sets. They can be calculated very rapidly.

### 2.2.5 Ranking

If we are docking proteins for which there exist experimental result of binding, then we can compare that experimental result with our application's output conformation and understand how close we were able to get to the real conformation. For this task, the most common criterion is RMSD (Root-Mean-Square Distance):

$$RMSD = \sqrt{\frac{\sum_{i=1}^{n} d_i^2}{n}} \tag{2.1}$$

In Equation 2.1, "n" is the number of atoms. "$d_i$" is the distance between the coordinates of $i^{th}$ atoms (in the experiment and in docking) when these atoms are superimposed. A smaller RMSD points to a better superimposition, hence a better docking.

# Chapter 3

# RELATED WORK

There are two competing approaches to attack protein-molecule docking problem: i) Automated, ii) VR-based.

## 3.1 Automated Approach on Protein-Molecule Docking

Several research groups are working on automated docking [20; 13] which can be reformulated as an optimization problem. The goal is to minimize binding energy of the molecular complex, however finding the best bound configuration requires checking unlimited spatial combinations of proteins. Consequently, heuristics are used with the goal of finding "good" results. The problem with these approaches, as indicated by Ferey et al. [11], is that finding notable solutions is not guaranteed and this process might take up to several hours.

Ferey et al. [11] mention three problems with fully automated approaches:

- Search space is very large (i.e. complexity is extremely high) due to the large number of probable binding sites.
- Most of the time, search algorithms come up with local minima.
- Probability of coming up with false positive results is high (i.e. lack of accuracy)

In their paper, it is noted that molecular visualization tools are being used in order to overcome these problems. Nevertheless, since it is not an easy task to orient and

rotate both proteins on a desktop application, VR-based applications are pointed out as better candidates for protein-protein docking problem in their paper. We dwell on VR-based approach to the docking problem in the following section.

### 3.2 VR-Based Approach on Protein-Molecule Docking

VR approaches have also been used for aiding molecular visualization and docking. Beginning from 1990s, several VR techniques have been proposed. In Akkiraju et al. [1], they design a system to view geometric protein structures from inside a CAVE (Cave Automatic Virtual Environment). In this work, they are able to visualize proteins in three different representative ways: space filling model, solvent accessible model, and molecular surface model. According to authors, unlike other virtual environments, the CAVE enables multiple viewers observe the same scene at the same time and place. The authors believe that virtual environments pave the way for new insights and discoveries. To support this belief, they state that they were able to notice the high frequency of protein self intersections only after using CAVE visualization of their software.

As in the case of Akkiraju et al. [1], Levine et al. [17] also try to create an immersive virtual reality based system via using a CAVE. The purpose of their system, “Stalk”, is creating an interactive virtual molecular docking environment; hence it is more comparable to our application in terms of its general aim. The backbone of this system is the usage of genetic algorithms. Other main portions of the system are taking advantage of parallel computing and distributed computing to speed up the system. Two molecules are drawn into the walls. One of them is receptor, the other is ligand. If the user wishes, she is able to translate and rotate the ligand. Conformation (i.e. translation and rotation) of the ligand is treated as a “string” in their genetic algorithm. At each run of their genetic algorithm, by creating new conformations, they try to reach a state of lower free energy. Although the system has to be called interactive, most of the job is done by the genetic algorithm itself. The user is limited to define a conformation at the beginning or in-between of a docking process. Moreover, they declare in their paper that user intervention did not enable the system to come up with lower free energies. Results

acquired by choosing random strings turned to be more successful than the ones gained by user intervention.

Figure 3.1 : User is trying to dock a ligand via using Stalk [17]

Anderson and Weng [2]'s paper has been the most enlightening work throughout my thesis research. They developed an interactive protein docking program which visualizes proteins in a virtual reality environment. In their program, VRDD (**V**irtual **R**eality Visualization to Protein **D**ocking and **D**esign), user is able to choose among different type of visualizations. Among these types, Van Der Waals (VDW) space-filling models and Solvent Accessible Surface (SAS) representation are of particular interest to our work. They mention VDW representation as the most attracting; they also note that they cannot render VDW models in a reasonable amount of time. As a result, they prefer SAS representation. The main reason why SAS can be rendered faster is that interior details are not included in SAS. Only the parts that belong to the surface are rendered and displayed.

To speed up the system further, Anderson and Weng use a "space compartmentalization scheme". By dividing the virtual space into cubes and assigning atoms to the cubes they reside in, it is enough only to check related cubes during energy calculation between molecules: The atoms that are far away (i.e. they do not interact) are not included in energy calculations, and this brings along an improvement to the speed of these calculations.

Bearing in mind that the user would like to keep the information related to best docking conformations so far, VRDD keeps some number of best orientations (i.e. rotation and translation data of both molecules) in a list. If the user wants, she can return to any of these saved orientations.

After the user reduces the search space by defining a possible docking confirmation, she can fine-tune this conformation by letting VRDD run an automatic Metropolis Monte Carlo local search. By using Metropolis Monte Carlo algorithm, a random conformation is selected for the ligand. If this new conformation is more favorable than the former one (i.e. it has a lower free energy), than this new conformation is always accepted. If this new conformation has a higher free energy than the former, then it is accepted with a probability. These actions are iterated for some predetermined number of steps.

All of these actions are realized in front of an Immersadesk. Unlike CAVE, Immersadesk does not completely extract the user from his previous environment. It has an inclined screen with one projector, whereas a CAVE covers the user from all directions (i.e. back, front, right, left, up, down) and needs four projectors. Immersadesk enables multiviewing too, but it does not provide the same level of immersion as

CAVE. Nonetheless, when its small size and cost are taken into account, Immersadesk system can be preferred, as it is in Anderson and Weng's work.

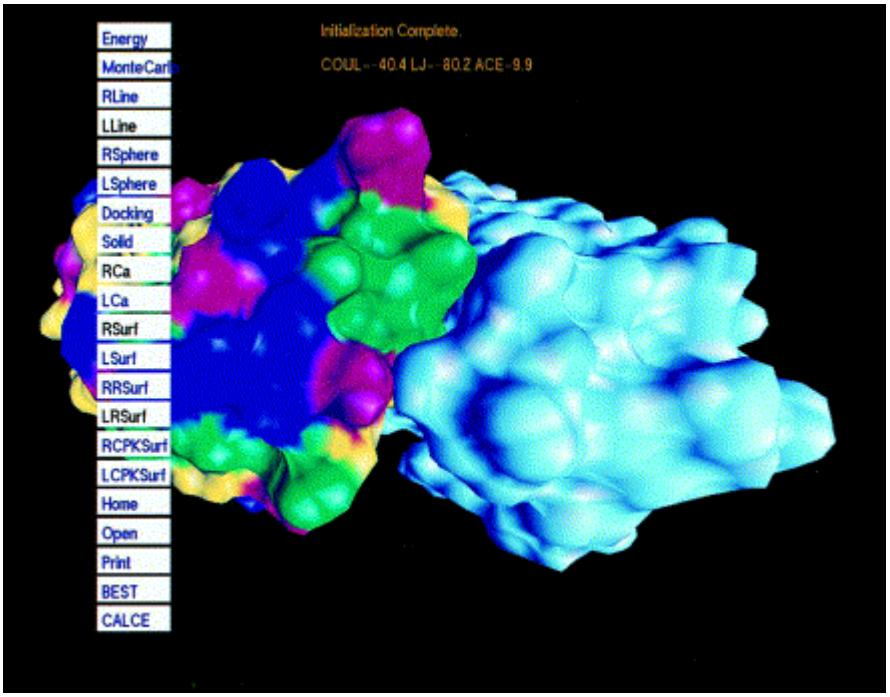


Figure 3.2 : A snapshot from VRDD display [2]

More recently, haptic interfaces have been introduced. Subasi and Basdogan [25] developed a haptic approach for ligand-protein docking problem. The user is able to control a rigid ligand and she tries to fit this ligand into a given rigid protein. The protein's position and orientation are fixed. The aim of the user is to find the best possible gap (for docking) on the protein. By the help of force feedback provided by the haptic device, the user feels a strong pull towards the gap if the gap is a feasible one. In addition to this pull, it is hard to take the ligand off a gap once it is roughly placed. It is harder to escape from a gap if the docking created at that gap gives rise to lower free energy than the dockings created at other gaps. First, possible binding sites are determined beforehand by using an automated application, "Pocket" [10]. Afterwards, the user tries to find the correct gap out of all possible gaps. When she decides on a certain gap, she tries to roughly align the ligand to the gap she has chosen. Fine-tuning on ligand's conformation is done via offline molecular dynamics calculations. These calculations are done up to the point when lowest energy possible is reached.

Figure 3.3 : Testing possible binding sites [25]

In their paper, Subasi and Basdogan [25] present their Active Haptic Workspace (AHW) concept. Since a ligand is a very small molecule and a protein is an extremely large molecule compared to a ligand, it would be hard to dock a ligand into a small gap on the protein if we had the whole protein visible in front of us: It would be nearly impossible to fit the ligand correctly since we would hardly be able to see the ligand itself. In order to overcome this problem, Subasi and Basdogan's application zooms into the area where the ligand is present, and the user can slide the ligand on the protein. When the user wants to move the ligand onto a part of the protein that is not visible on the current view, the workspace is moved towards the region onto where the ligand is moved. This concept, AHW, enables the user to focus on possible binding sites.

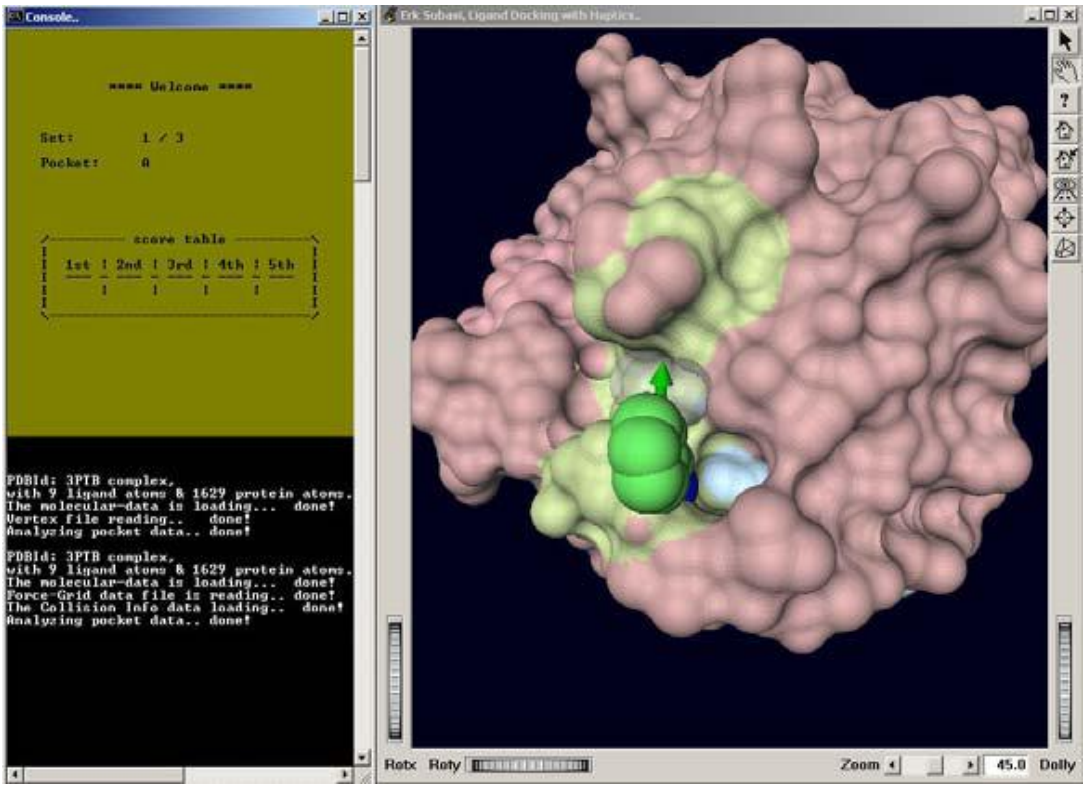


Figure 3.4 : Screenshot of the interface used during experiments [25]

Ferey et al. [11] have designed a haptic-based user interface with user needs in mind. They use many devices for the task of protein-protein docking. The virtual reality platform is a system that resembles CAVE. Shutter glasses, 3D mouse, 3D audio feedback and 6 DOF (degrees of freedom) haptic devices are used in their docking application. First, the user is able to choose several hotspots using 3D mouse. Then, the selection is given to an automated program to shorten docking time and search space. After this step, docking takes place. One of the proteins is manipulated via a 3D mouse while the other one is manipulated via a 6 DOF haptic device. The user wears shutter glasses for stereo view. When the haptic feedback is not applicable due to the complexity of global docking, audio feedback is used for the same purpose (i.e. rendering collision & outputting surface complementarity scores).

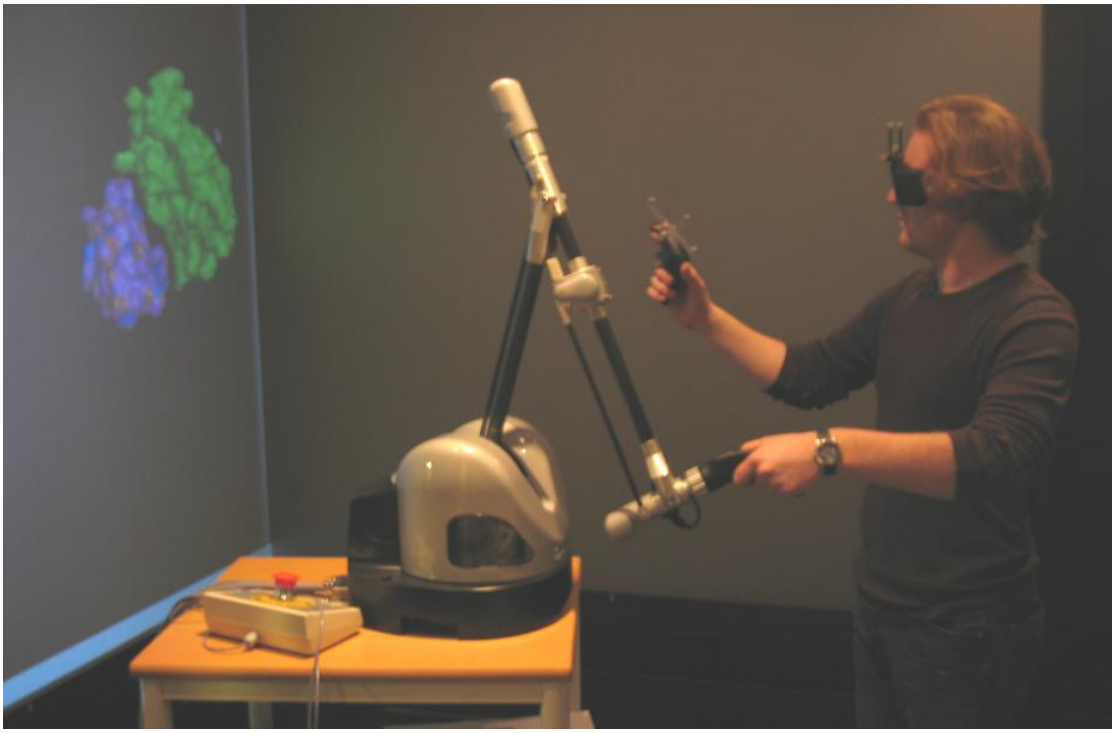

Figure 3.5 : Testing haptic docking [11]

# Chapter 4

## SCIENTIFIC VISUALIZATION

As it has been discussed in Chapter 3, haptics have been used to aid users at protein-ligand docking problems. In general, in haptic based applications for protein-ligand docking, protein is fixed (it is not translated during the application) and ligand is controlled by the user. Binding energies are calculated by taking into account interactions of each atom in a molecule with each atom in the other molecule. Due to the restriction of haptic devices, the user cannot understand each ligand atom's "intention", that is, she cannot have output related to which direction each ligand atom wants to go as a result of ligand's interaction with the protein. The output of haptic device is an overall average of forces impacted on each atom of the ligand. The reason behind this is the limitation of haptic devices. Force feedback can be provided from just a single point, not from several points as desired (i.e. we would like to get information from each ligand atom). Nevertheless, this limitation does not cause any notable problem since ligands are generally composed of very small number of atoms. Consequently, haptics is finely applicable to protein-ligand docking problem.

In our main problem of protein-protein docking, even if one of the proteins is fixed as in the earlier case of ligand-protein docking, the other protein should be able to translate and rotate freely. In a given protein, there are many regions of different physicochemical characters. Some regions may be hydrophobic, while some regions are polar, whereas some other regions are charged. In order to lead the user in a well-informed manner, we should provide force feedback data for each atom in a protein. Since this is not doable using haptic devices, we tried to attack the problem via scientific visualization methods. To put in a nutshell, scientific visualization is an area where the aim is to render surfaces or volumes of three dimensional phenomena in order to simplify the understanding of the underlying huge and complicated data sets.

We have developed two novel scientific visualization methods. In the upcoming subsections, these methods are going to be covered in detail.

## 4.1 Dynamic Color Coding

Force is an agent that changes a matter's velocity either as a pull or a push. As it is a vector, it has a direction and magnitude. It has been explained in Chapter 2 that there are several types of forces to be considered at molecular level.

In haptic devices, direction and magnitude of a given force is output in a straightforward way: The robotic arm moves in the direction and with the specified magnitude. Since we are not using haptic devices for protein-protein docking problem as a result of the aforementioned reasons, we have to find other methods to deliver data related to direction and magnitude of the force. One method we have derived is "dynamic color coding". Basically, each atom –no matter what representation style is used- is colored to give information about direction and magnitude of the net force imposed on it by the accompanying protein. For our task, it would be enough to decide on:

- a color to denote that net force is zero on a given atom
- a color to denote that net force is a pull
- a color to denote that net force is a push
- tones of colors chosen for denoting push and pull

We basically choose two colors denoting the ends of our scale. One color stands for the maximum pull possible, while the other means maximum push. Respective color of a force can be:

- a mixture of the "zero force" color and "maximum pull" color
- a mixture of the "zero force" color and "maximum push" color
- "zero force" color if the net force is zero
- "maximum push" color if the net force equals to the maximum push possible in the given case
- "maximum pull" color if the net force equals to the maximum pull possible in the given case

Depending on the magnitude of a push/pull, percentages of colors in the mixture vary. This issue is going to be touched upon further in Chapter 5.

Dynamic color coding does not give perfect information about direction of the net force on a given atom. It just notifies how much that atom wants to be near the accompanying protein in its present conformation. Although we can understand whether our atom wants to go towards or apart from the protein, we do not know which way it wants to move exactly.

## 4.2 3D Rose Glyph

The other method we have developed to visualize force interactions is called "3D rose glyph". It may be conceptualized as a three dimensional histogram where "direction" of data plays an important role as well as its magnitude.

To understand 3D rose glyph adequately, prior works that have given rise to this notion should be covered. Florence Nightingale, who was also a statistician besides being a pioneer of modern nursing, developed a chart type called "rose diagram" [22]. Nightingale owes her fame mostly to her works during the Crimean War in 1854 at Selimiye Barracks, Istanbul, Turkey. By developing necessary sanitary conditions, she was able to cut mortality rates from 42 percent to 2 percent. In her rose diagram [Figure 4.1], there are sectors, each of which corresponds to death rates in a given month. In the same sector, there are three different wedges: Blue indicating death from lack of hygiene, red indicating death from wounds, and black indicating death from all other reasons. Each wedge is measured from the center of the circle and a given wedge's importance is understood by its radius' length. In this diagram, she showed that lack of hygiene was by far the superior cause of mortality.

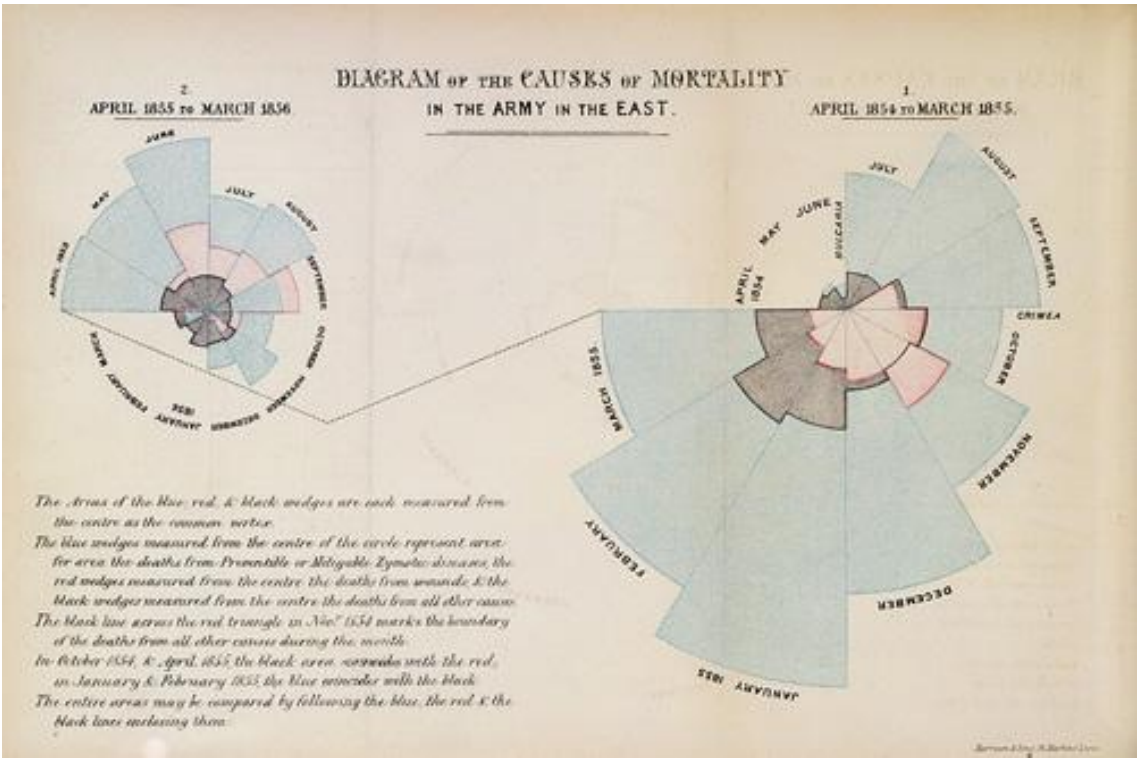


Figure 4.1: Nightingale's rose diagram [22]

In Nightingale's rose diagram, direction of a sector does not carry any information in itself. Modifying her notion of rose diagram, Meier et al. [18] developed a type of interactive rose diagram where each sector's direction, along with its magnitude, has a meaning. In their work, a rose diagram [Figure 4.2] is used to visualize particle contacts in granular media. They work on 2D data sets, and hence a 2D diagram to depict force relations (i.e. rose diagram) is sufficient. They mention transforming their "rose diagrams into diagrams that are able to display 3D force relations". Although the underlying topic is different from ours, we used this idea to build our "3D rose glyphs". Owing the term "rose glyph" to Meier et al. [18], a 3D rose glyph visualizes 3D force interactions among atoms of a protein and atoms of another protein.

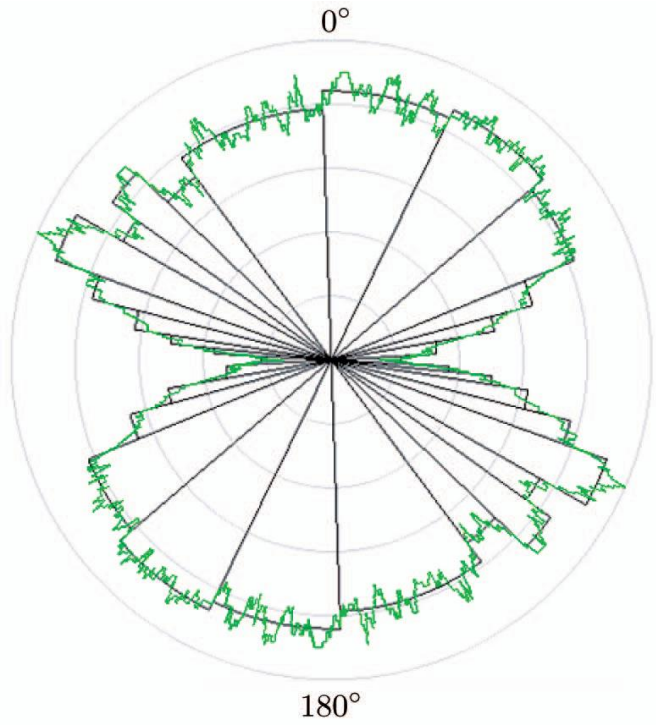


Figure 4.2: A rose diagram depicting 2D force interaction between granular media [18]

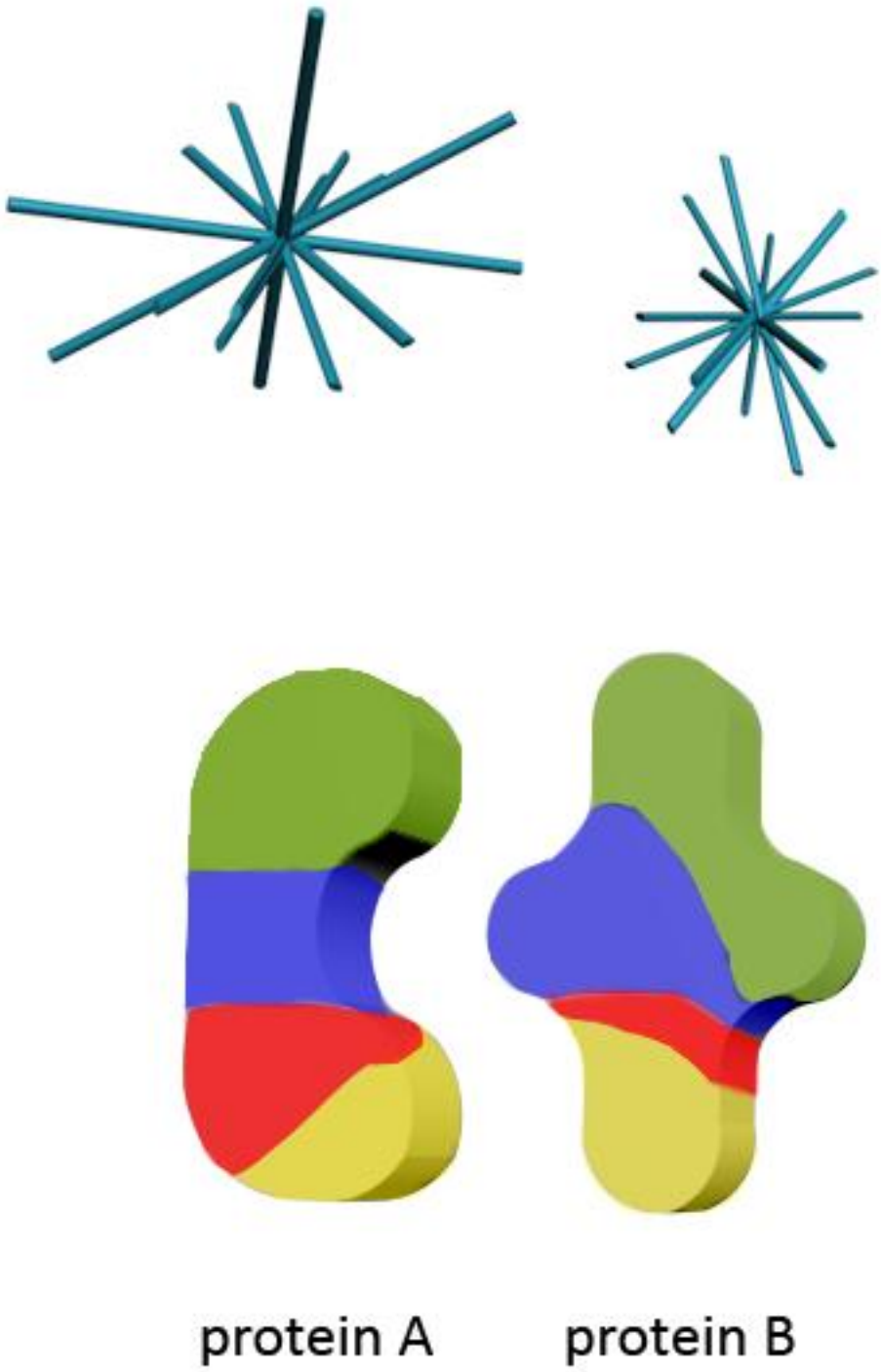


Figure 4.3: A sketch of 3D rose glyphs (up) to be used in our application. The 3D rose glyph on top of protein A is the glyph of protein A. The same goes for protein B.

In Figure 4.3, a sketch of 3D rose glyphs is shown. As it can be seen in the figure, in a given rose glyph, there are several cylinders of different lengths and different directions. In a 3D rose glyph, each cylinder carries information related to a specific atom in the related protein. Each cylinder shows the total force on the related atom that arose from its interactions with all the atoms of the other protein. The length of the cylinder tells the amount of force, and the direction of the cylinder tells the direction of this total force. Therefore, unlike dynamic color coding, 3D rose glyph method provides both direction and magnitude of force vectors. Eventually, it provides all the information that comes from the interaction between proteins.

# Chapter 5

# DOCKPRO

In specific systems, there are certain general constraints that must be satisfied for a successful docking to happen. These constraints are usually known by molecular biology experts. With their knowledge, search space can be limited and hence a more accurate and faster solution could be expected.

We can give the role of hydrophobicity (water-fearing) as an example. If the expert using the system thinks that hydrophobicity plays a great a role for input proteins, she can come up with a good solution faster if she is given the chance to group amino acids accordingly (e.g. hydrophobic, polar and charged) and then give highly negative scores between the hydrophobic group and other groups. These tasks, among other important aspects, can be accomplished by using our application.

We developed an interactive 3D application, DockPro. Humans are good at "put-the-block-into-the-gap" type of problems. A molecular biology expert can come up with a successful docking by changing translation and orientation of the proteins. Chemical and physical characteristics of atoms also play a key role in the docking problem. It is not a sufficient aspect to have good surface complementarity on its own. In addition, care must be taken to concatenate atoms, which like to be near to each other, to have a stable complex.

For the purpose of interaction in protein-protein docking, we use magnetic trackers and gloves. User wears magnetic tracking sensors that are attached to the top of each glove on both hands (Figure 5.1 & 5.2). Direct manipulation of proteins is done with hands. The virtual environment is displayed on an immersive workbench. The system provides a natural and easy way to work in front of a large display. It is natural; because at the time of docking, expert uses her hands as if she is trying to concatenate

plastic protein models. It is user-friendly; because expert can carry out each necessary step while standing in front of the large display.

The system runs on Intel Pentium D 3.2 GHz CPU and NVIDIA Quadro FX1500 GPU. We use Flock of Birds [12] magnetic sensor system to track user's hands. To switch between different tasks, we developed a hand gesture recognition scheme on CyberGlove [9], which considers only six basic hand postures. Input files of proteins of interest are provided in PDB file format [3]. From these files, c-alpha atoms (i.e. alpha carbon atoms) of each protein are extracted to be rendered later on.

Throughout my thesis, we have gone through two different phases. In the first phase, along with the initial docking simulation, we have simulated **dynamic color coding**, details of which was given in Section 4.1. In the second phase, we further developed the quality of visual aspects of the system, integrated an empirical scoring function, and simulated our notion of **3D rose glyph**, which was mentioned in Section 4.2.

## 5.1 First Phase

DockPro consists of four windows, which are displayed to user one after another. These windows are used in order to:

- Group amino acids,
- Assign score (force) relations among groups,
- Assign colors to groups,
- Assist user to perform protein-protein docking.

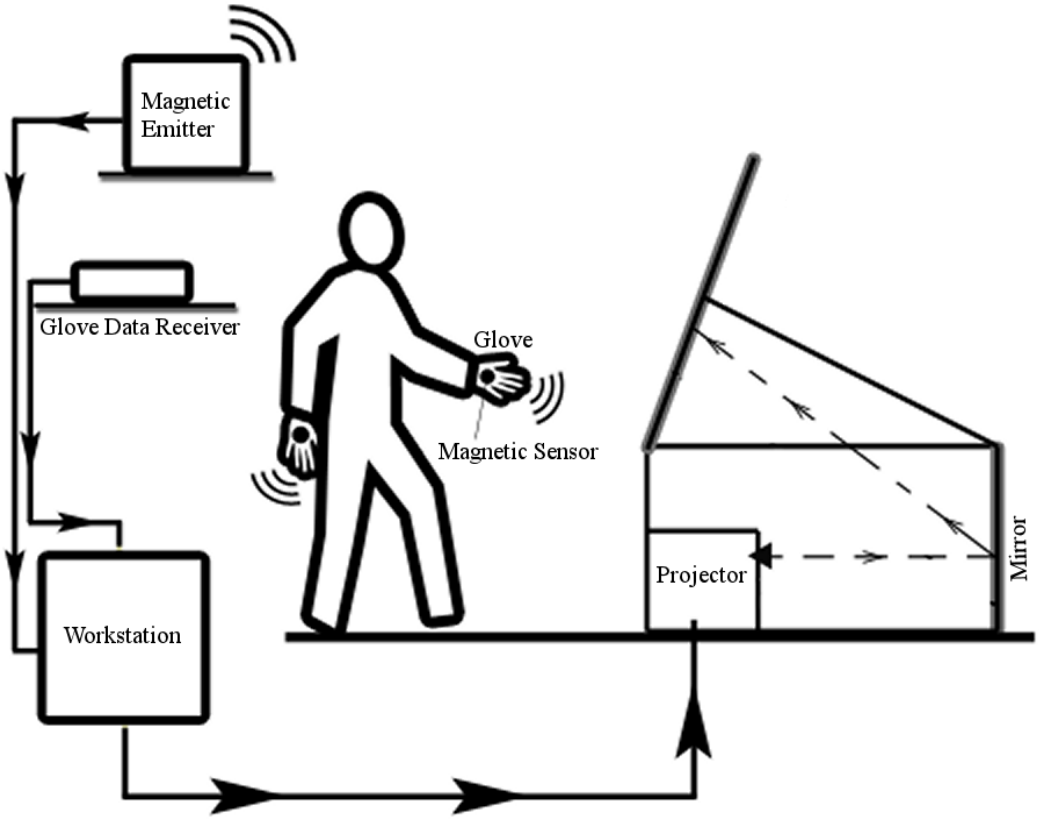


Figure 5.1 : System overview


Figure 5.2 : User is standing in front of the workbench and realizing the docking process.

### 5.1.1 Grouping

The grouping window enables the expert to group amino acids in any order. A group contains amino acids that have similar properties. Each group has distinctive characteristics which arise from their amino acids' physicochemical attributes. Every amino acid should be assigned exactly to one group.

In this window, user begins by choosing amino acids of the first group to be assigned. In order to accomplish this task, she moves the cursor to the related amino acids' icons by moving her right hand onto which a tracker is attached. Then, she chooses the amino acid (on which the cursor stands) by a hand gesture that is captured via the glove. Undoing is possible; she can remove any amino acid from the group that amino acid was previously assigned. Assignment of amino acids to the first group ends when a unique hand gesture is made.

These steps are done for other groups yet to come, if there is any. Expert can see the groups created so far at the bottom of the window (i.e. related legends are present). Groups are numbered, in the order they are created, from 0 to n-1, where n is the total number of groups. When there is no amino acid left that is uncovered, it means that we are done with the grouping phase and hence we move into the second window.

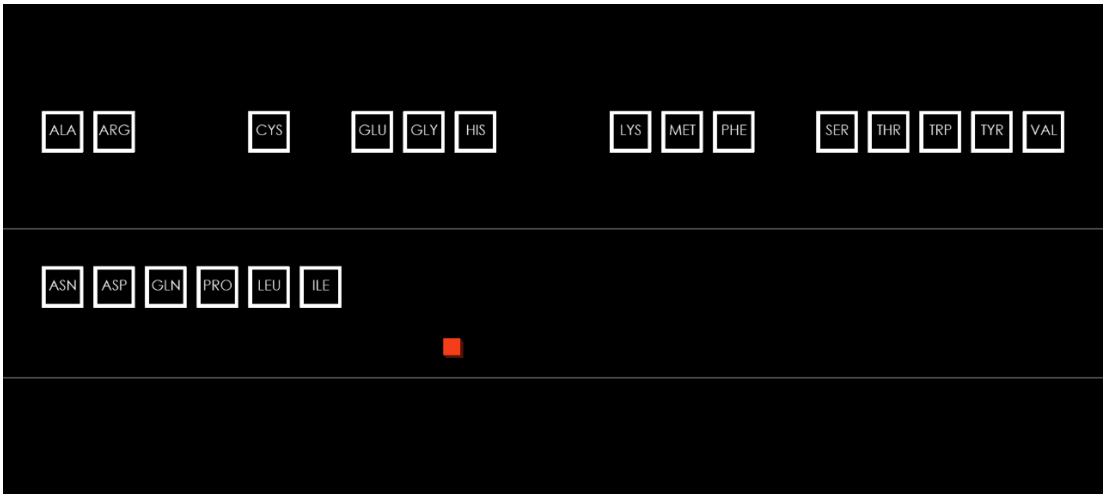


Figure 5.3 : Grouping amino acids. The cursor (i.e. red square) is moved on top of the desired amino acid icon and a hand gesture is done to select it. At this specific time point, six amino acids in the middle of the screen are chosen to be assigned to the same group.

### 5.1.2 Score Relations

In the score relations window, we are able to determine the scores (i.e. physicochemical forces) between any pair of groups, which we have created in the grouping window. Each score between two groups shows how much amino acids in one group like to be around amino acids in the other group. Using these scores, we try to mimic physicochemical forces that play key roles in docking. Assigning score relations between amino acid groups constitutes binding energy functions, which determine the binding strength of the complex. Higher overall score means higher binding strength (i.e. better docking). Initially, all possible pairs have a score value of 0. User can assign, and also reassign, scores in any order. User is not obliged to appoint a score to each relation; untouched relations will have values of 0. There are icons of group numbers and a scale bar. There are also informative legends about members of each group and scores assigned.

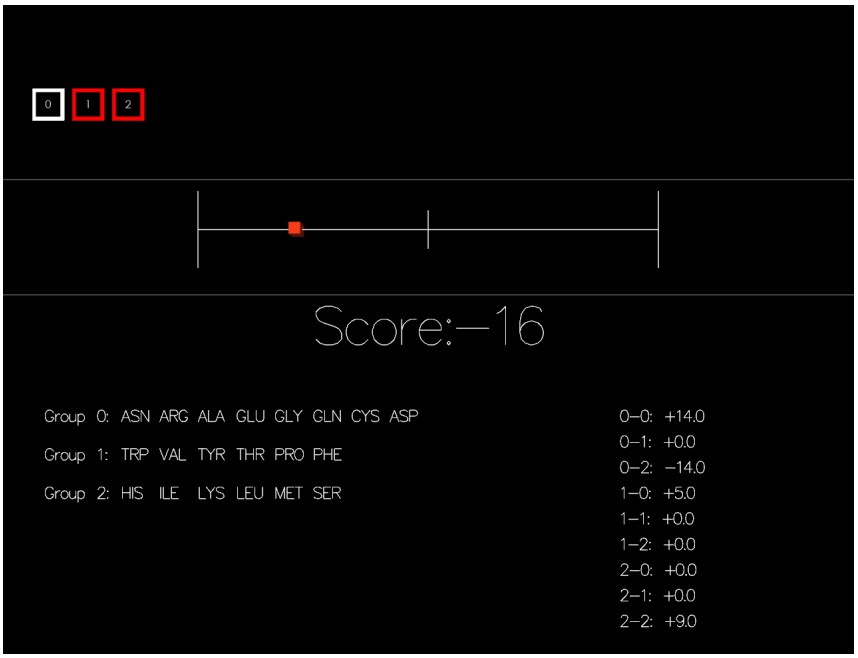


Figure 5.4 : Assigning scores to groups. In this figure, we can understand that three groups have been created in the previous window. Groups and their members can be seen at lower left corner. Scores between groups can be seen at lower right corner.

### 5.1.3 Color Assignment

In the color assignment window, unique colors are assigned to groups. These colors are set to be used in the docking window to mark which amino acid group each atom belongs to. In this window, we have icons of group numbers and also 20 colors. User is expected to match each group with a color.

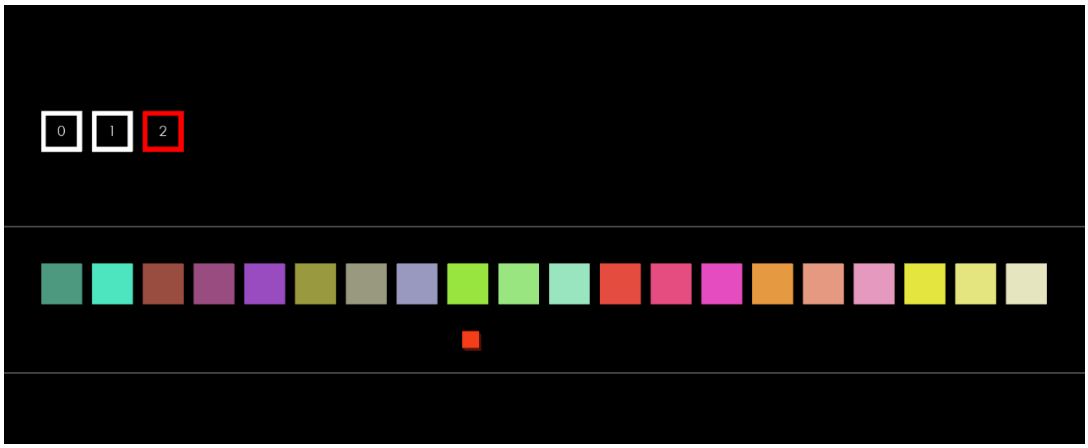


Figure 5.5 : Color assignment. There are 20 different predefined colors that can be assigned to our groups. In this figure, color code for group 2 is being chosen.

### 5.1.4 Docking

In Figure 1.1, docking window of DockPro, where main actions take place, can be seen.

In Figure 1.1.a, DockPro's regular docking view is shown. There are three view modes. In the default view, both proteins are fully visible, and they can be docked as anticipated. Each atom takes up space relative to the Van Der Waals volume of the amino acid it belongs to. Manipulations on translations and rotations are nonisomorphic: Both of them are scaled down. As well as shape complementarity, the overall score of the system also plays an important role in the docking process. The score of the system is shown on the corner. Our score relations are dynamic and computed in real time: Their magnitudes change inversely with the distance between atoms. Expert can understand how good a docking is by checking the overall score. When any two atoms of proteins collide, those atoms are highlighted: They blink at a constant frequency. Colliding proteins cannot penetrate into each other. Instead, user should adjust proteins' orientation/translation to relax the collision.

Apart from the default view, we have one general view mode for visualizing the protein complex, and one for fine-tuning. The former enables the expert to see previous dockings and examine structures by treating them as a single entity rather than two different proteins. This view has two options. When a collision occurs, one protein's collision surface is drawn as semi-transparent. Hence, expert can see totally opaque protein through semi-transparent amino acids of the other protein. This enables an in-depth analysis of the collision surface.

We have one more view mode for fine-tuning. In this mode, the expert chooses a previous docking's collision surface and tries to increase the score continuing from this configuration. Both manipulations (translating and rotating) are nonisomorphic like in the default view; but they are scaled down further.

For each view mode and its submodes, a symbolic hand gesture is assigned. By this way, user can switch from one view to another without any interruption. In addition, expert can also halt the system by a hand gesture. In our application, all gestures are done via left-hand glove.

When user finds a configuration noteworthy, she can save the data about resulting complex by a gesture. This data contains:

- Atom coordinates of both proteins,
- Score of the complex.

If there is data about the complex in a protein-protein docking benchmark (e.g. Chen et al. [5]), user can compare her docking with the benchmark and check how successful her docking is.

In Figure 1.1.b, we can see the inside story of 1.1.a. This part differs from 1.1.a in that we are able to visualize the force relations that contribute to the total score. Here, no group coloring is present. Every atom has the same initial color: gray. Gray is the neutral color in our scale, meaning that the net force on an atom is zero. When net force on an atom changes, that atom's color changes too. Color of an atom signifies the charge and magnitude of force that is exerted. In Figure 5.6, you can see the corresponding map. Blue color corresponds minimum score value, and red color corresponds to maximum score value out of all relations. Current subscreen enables us to see how the overall score is constructed. Having the aim of maximizing the overall score in mind, user gets auxiliary information to come up with a successful docking. So it can be proposed that visual feedback by color coding helps user by reducing the search space further (see Figure 5.7 and 5.8 for closeups).

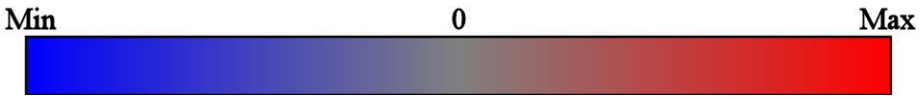


Figure 5.6 : Color mapping of score relations

Figure 1.1.c and 1.1.d are dynamic legends to help user see the colliding atoms (if any) with the aim of making the docking process easier. In 1.1.c, there are six views of the protein that is controlled by user's left-hand. Each view plane is orthogonal to each other: Imagine that this protein is surrounded by a transparent cube. Each view is taken from one of the six sides of a cube. Auxiliary information about the view is provided by legends of imaginary cube: It shows from which side we are looking at our protein. Part 1.1.d is the dynamic legend for the other protein. At the time of docking, any colliding

atom pairs are highlighted at Figure 1.1.c and 1.1.d as in 1.1.a. Consequently, we are able to see each possible collision which may not be seen from 1.1.a's default view. These legends, along with part a's collision surface view, enhance user's understanding of the process.

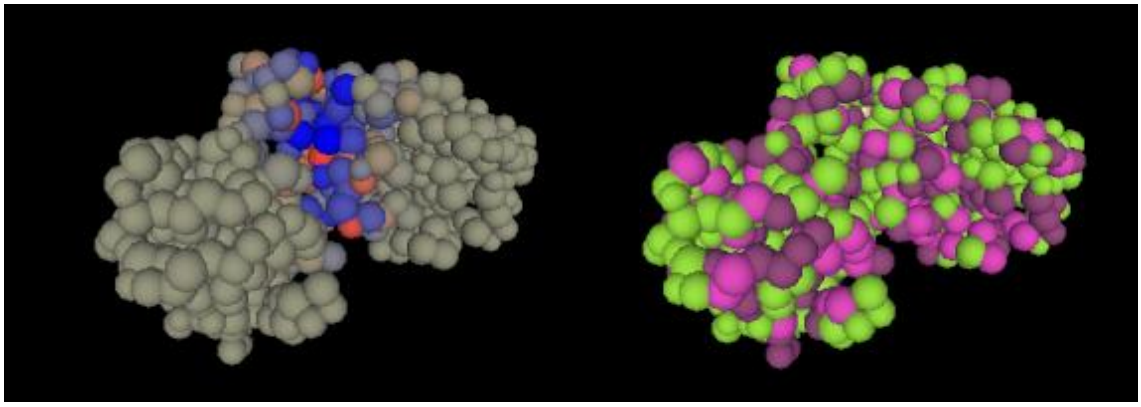

Figure 5.7 : Different views of docking. The complex on the left side visualizes force relations. The one on the right side visualizes distinct amino acid groups.

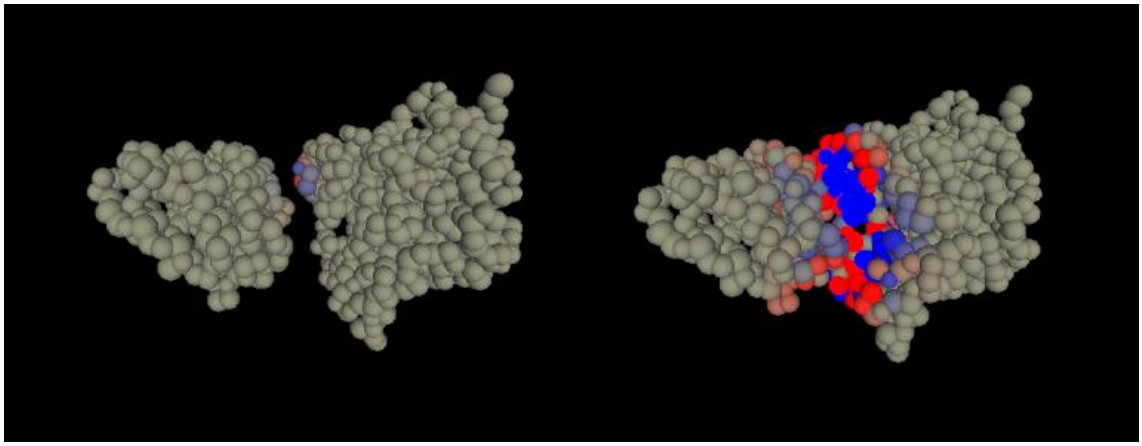

Figure 5.8 : Force calculation.

## 5.2 Second Phase

In the second phase of our application, as a first step, we have implemented our idea of “3D rose glyph”.

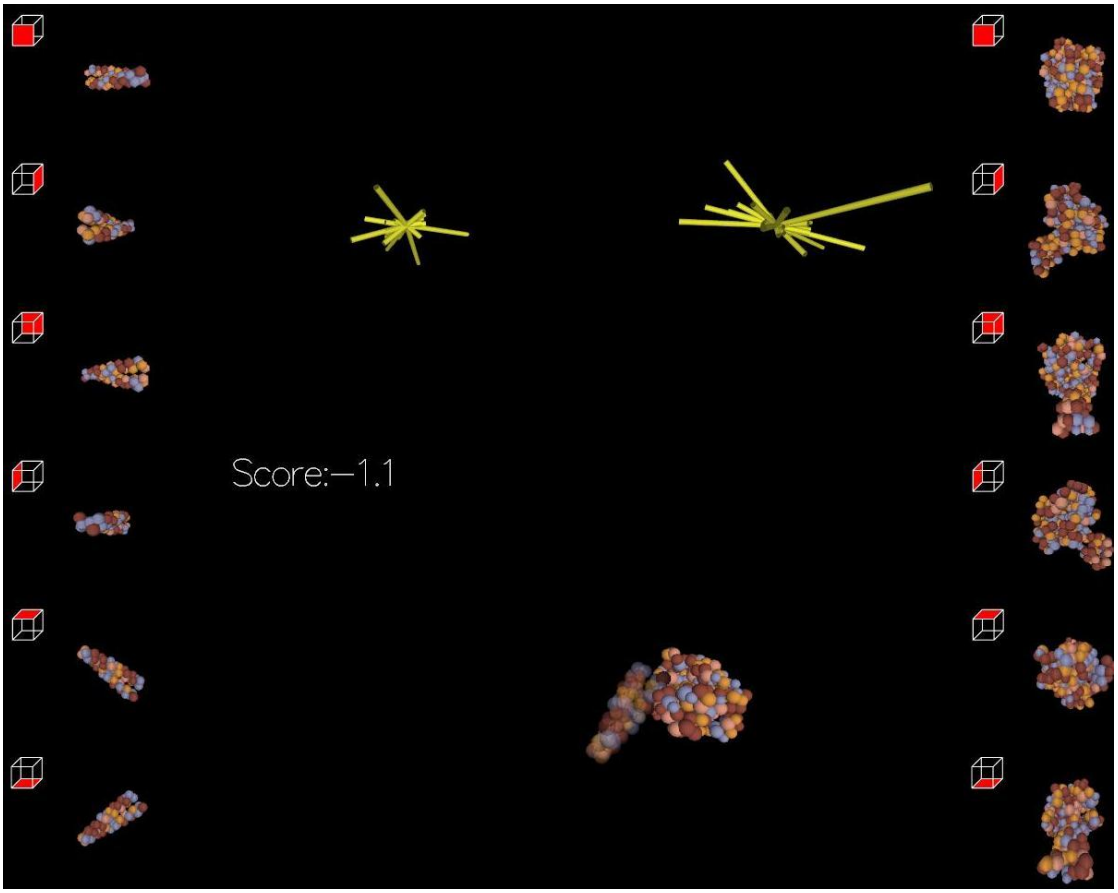


Figure 5.9 : The only difference between the main window of the first phase and this stage of the application is the implementation of 3D rose glyphs rather than dynamic color coding of proteins.

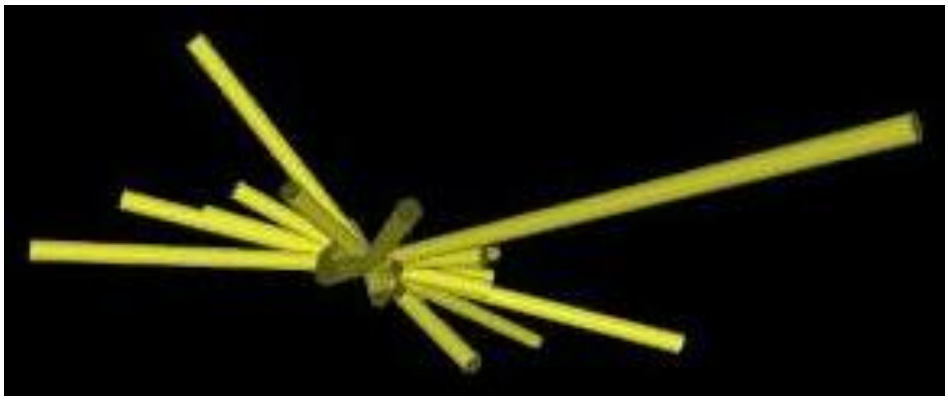

Figure 5.10 : Close-up view of a 3D rose glyph.

In Section 5.1, it has been discussed that dynamic color coding gives information on how much each atom “likes” its current position. We can understand by looking at the color of a given atom and conceive whether it wants to get closer or further away. Nonetheless, we cannot perceive the exact direction that atom wants to follow. Bearing in mind this lack of information, we tried to find another scientific visualization method. As a result, after examining Meier et al. [18]’s work and realizing that we can modify their rose glyphs to satisfy our needs, we came up with the idea of 3D rose glyph, which was mentioned in Section 4.2.

In Figure 5.9, the glyph to the left depicts the forces on the protein to the left, and the glyph to the right shows the forces on the protein to the right. These glyphs are interactive. When the user moves a hand, forces interacting on both proteins change. Hence, shape of glyphs change accordingly.

In Figure 5.10, we see a close-up view of the glyph that belongs to the protein to the right in Figure 5.9. For each atom in the protein, there is a related cylindrical shape. A given cylindrical shape depicts the total force on the corresponding atom. It also tells the direction of this total force. For example, if we look at the longest cylinder to the right in Figure 5.10, we understand that the atom that corresponds to that cylinder strongly wants to go the right. Since the other protein is to the left of this protein, it means that our atom really “hates” the other protein at that conformation and wants to go away.

It is sufficient for the user to have a quick look to both of the glyphs to understand that current conformation does not lead to a successful docking. In both of the glyphs, there are big and more or less equal chunks of cylinders to the left and right. Therefore, it means that, for each protein, nearly half of the atoms want to go towards the other protein, while the other half want to go away from the other protein. Unlike this case, in

a successful docking we would expect to see many of the cylinders of a given glyph to point towards where the other protein resides (same goes for the other glyph).

In the current state of our application, both 3D rose glyph and dynamic color coding methods are implemented. Only one of these methods is shown to the user at a given time. User can switch between these auxiliary views by a specific hand gesture.

After implementing 3D rose glyph method, we concentrated on other aspects of our application. We enriched the system, made it more modular, and sped up the energy function calculations.

Like the first phase of our application, the code of the second phase is written with C++ language. The main library that our application extensively depends on is OpenGL [23]. OpenGL (Open Graphics Library) is a widely known and used cross-platform 2D/3D graphics API (Application Programming Interface).

At the current state of our system, two different IDEs are used: Microsoft Visual Studio [19] and Code::Blocks [7]. The system is run both on Windows and Linux operating systems. CMake [6], which is a cross-platform build automation system that generates compiler independent configuration files, is used during transitions between different environments we use.

In order to increase the visual perception, most importantly the perception of depth, we used the CGAL library [4] to create 3D skin surface meshes of proteins. CGAL (Computational Geometry Algorithms Library) provides efficient algorithms in computational geometry. CGAL has many packages; for our purpose, we use the mesh generation package, which contains algorithms for 3D skin surface meshing. To put in a nutshell, we use this package to come up with skin surfaces that are tangent continuous. As a result of using this package, protein images look more attractive and visual perception is increased. Comparison between the previous method and skin surface method can be made by checking Figure 5.11 and Figure 5.12.

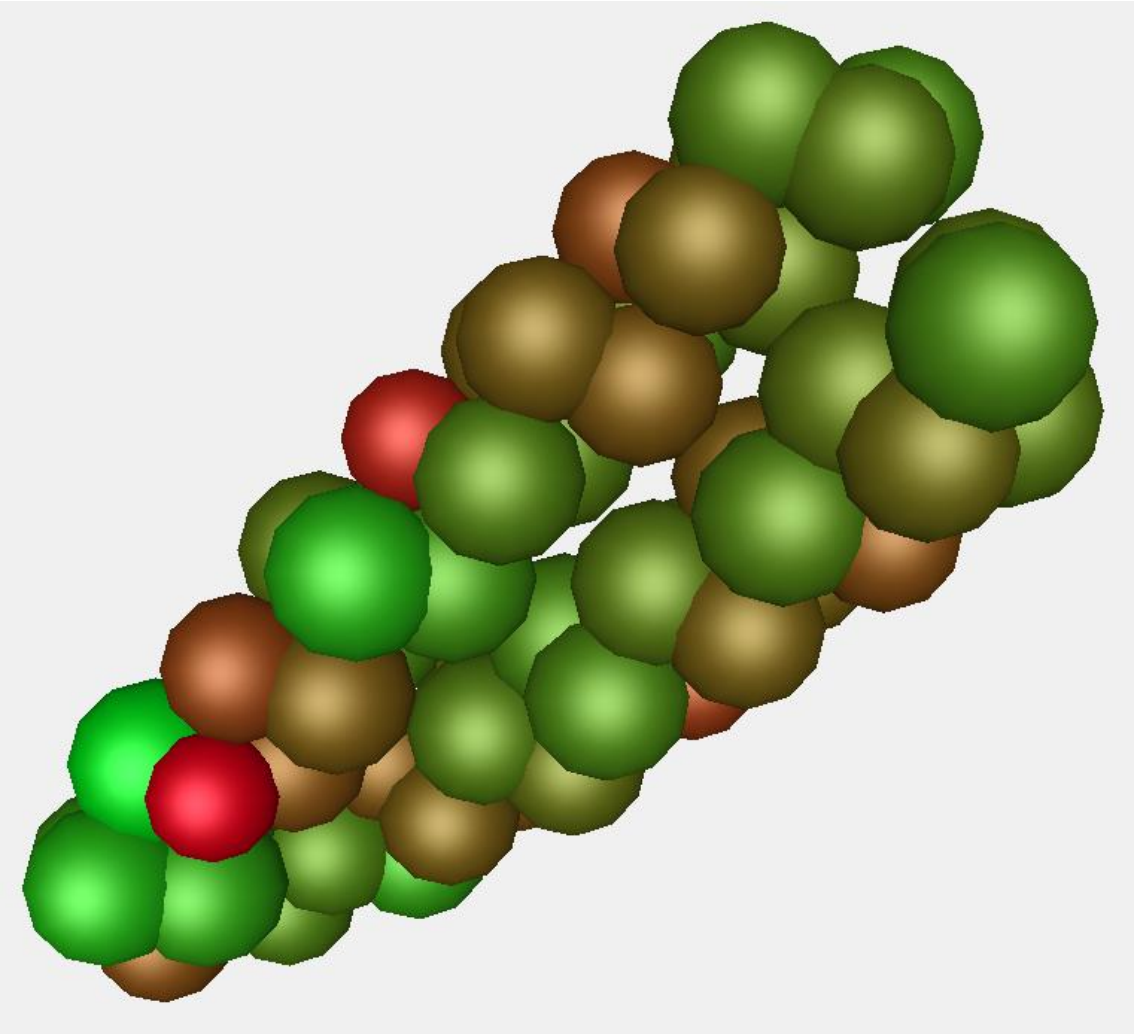

Figure 5.11 : The previous method where we only draw VDW radii of C-alpha atoms.

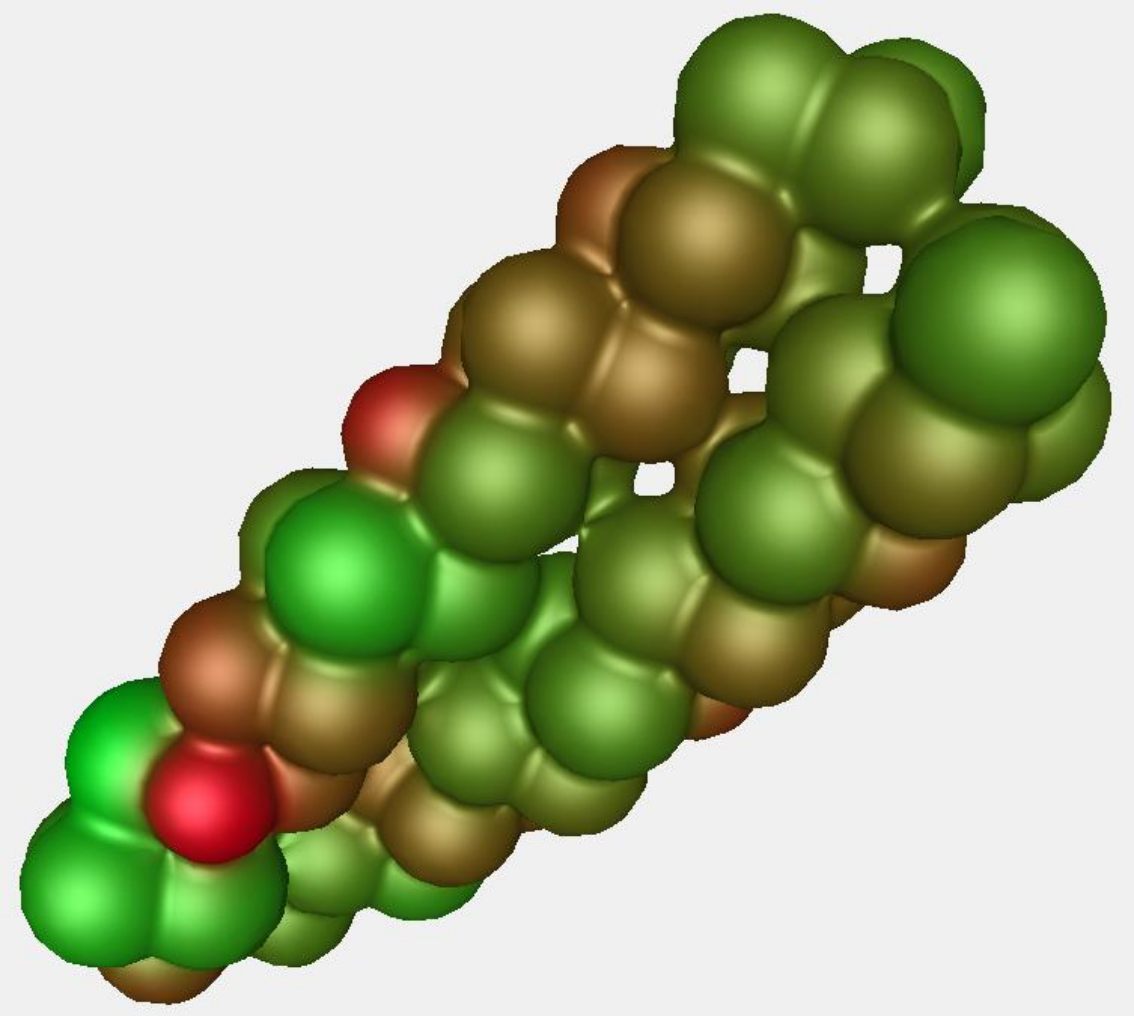

Figure 5.12 : Skin surface obtained by using CGAL. The surface is tangent continuous.

In order to speed up energy function calculations, we use Intel MKL (Math Kernel Library) [14]. This library provides highly efficient calculation methods for math routines. In our application, matrix operations are done via Intel MKL.

Input devices have their own libraries in our application. We have libraries for keyboard, for gloves, and for trackers. Keyboard is used during testing phase, so having a keyboard library is handy. Any combination of these devices can be used while running the application.

Unlike the dummy scoring function in phase 1, we use a modified version of the empirical energy function of Launay et al. [16]. In their paper, they provide several groupings of amino acids. For each grouping, there is a corresponding lookup table. The impact of interaction of any two amino acids is predefined.

$$E(C) = \sum_{i,j} C_{i,j} U(S_i, S_j)$$

(5.1)

In Equation 5.1, “i” and “j” stand for amino acids. $U(S_i,S_j)$ is the interaction energy between “i” and “j” that we check from the look-up table. $C_{i,j}$ is a constant which equals to 1 when “i” and “j” are closer than 4.5 Ångstroms. If the distance in-between is more than 4.5 Å, then $C_{i,j}$ equals to 0. What this energy function does is that it adds up all the interaction energies among each amino acid in a protein and each amino acid in another protein (Of course, if any amino acid couple is further than 4.5 Å, their interaction energy counts as 0 in the summation).

In our application, we use Launay et al.’s look-up tables and a derivation of their energy function. Rather than having $C_{i,j}$ equal to 1 at any distance closer than 4.5 Å, we gradually increase the contact score from 0 to 1. That is, a given amino acid couple reaches their contact score, which is written in the look-up table, they surfaces touch each other; so, $C_{i,j}$ equals to 1 only when they are touching each other. Between 4.5 Å and 0 Å, there is inverse proportion between $C_{i,j}$ and the distance between two amino acids.

# Chapter 6

# DISCUSSION AND CONCLUSION

## 6.1 Interaction Feedback

### 6.1.1 Haptics

Haptic devices have been used for ligand-protein docking problem [21; 25] to simulate electrostatic potential energy that plays an important role during the process. In haptic applications of ligand-protein docking problem, user moves the ligand with a haptic device in 3D space, and tries to position it on the protein. The force on the ligand is calculated and rendered to the haptic device as if it is emitted from one point. However, every atom of the ligand has a force interaction with every atom of the protein and hence each atom of the ligand has its own force. Since ligands are very small molecules, this approach does not give rise to any drastic errors.

While such aggregation is tolerable in the case of ligand-protein docking problem, it is inapplicable to protein-protein docking problem. It is not sufficient to calculate the total force between two proteins since there can be several amino acid groups that we should give feedback on their force relations. Consequently in this case force aggregation for the whole protein is not possible. Hence, it is better to calculate all force relations between each atom of two proteins, and render them in a visually appealing way.

### 6.1.2 DockPro Visual Feedback

Since we cannot use haptics for protein-protein docking problem, we proposed the following solution in the first phase of the system:

$$F_{p_i^1} = \sum_{j=1}^{n} f(p_j^2 p_i^1) \tag{6.1}$$

Total force exerted on $i^{th}$ atom of protein $p^1$ by atoms in protein $p^2$ is shown in Equation 6.1. Expert can define and adjust the scoring function *f*.

To provide visual feedback, we use the method of dynamic color coding. Colors stand for charges and magnitudes of forces. Color of a given surface indicates the magnitude and charge of total force exerted on that surface (Figure 5.8).

User can assign scoring functions (physicochemical forces) between each group. In DockPro, user creates her own intergroup score table (i.e. by creating groups and assigning scores accordingly). Usage of expert's own knowledge enables her to favor interactions observed between the examined proteins.

In the second phase of the system, we implemented Launay et al.'s [16] empirical energy function with a modification. We also presented another visualization method called 3D rose glyph. Both 3D rose glyph method and dynamic color coding method can be used in the system with the energy function defined.

## 6.2 Workspace

### 6.2.1 Haptics

An important problem with haptic devices is their highly restricted workspace due to hardware constraints. In ligand protein docking problem, there is a large molecule (protein) and a relatively small one (ligand). For this reason, one must "zoom" into the

area of interest on the protein by a factor that is enough to fit the ligand easily. As it has been mentioned in Section 3.2, Subasi and Basdogan [25] developed a technique called Active Haptic Workspace, which enables zooming and panning. However, this method is inapplicable to protein-protein docking since both are large molecules requiring large workspace.

### 6.2.2 DockPro Environment

In our application, we are using magnetic trackers rather than haptic devices, and our workspace is only limited with the reach of our arms if we stand still. Moreover, the user can move back and forth to zoom in or zoom out further. Hence, our system does not suffer from workspace limitations on protein-protein docking.

## 6.3 Future Work

By using the RMSD method mentioned in Section 2.2.5, we are planning to evaluate our application. We will try to dock several protein-protein couples of which experimental data are already available. From this point on, there are two different ways to be followed:

- Firstly, we are going to get the output conformation data and compare it directly with experimental data. The aim is to see how close we were able to get by solely using DockPro. We are going to try both 3D rose glyph and dynamic color coding methods.
- As a second step, we will get the output data of DockPro and give it to a fully automated docking algorithm with the expectation of obtaining "fine-tuned" results. We will then compare these new results to the experimental data and see whether this operation leads to better docking or not. Of course, the automated docking algorithm should be one that makes local searches around the input conformation. Otherwise, our specific input conformation to the automated program would have no specific meaning.

## 6.4 Conclusion

In this thesis, we have presented an interactive protein-protein docking application, DockPro. With its use of magnetic trackers and gloves, and use of large display, it provides an easy-to-use system. If we sum up the things implemented throughout out the first and second phases of the system, the application addresses several aspects of protein-protein docking:

- It enables an expert to create her own combination of amino acid groups. In addition, expert can define force relations among groups. Unlike common practices of force calculations, this method allows problem specific force adjustments.
- An empirical energy function is implemented.
- At the time of a collision between two proteins, the collision surface is extracted and one of the proteins is rendered semitransparent. This helps the user to analyze the collision area, which would most probably be occluded and unable to be seen due to the dense formation of atoms.
- We propose a force aggregation scheme and render its results color coded on both molecules. We also proposed another visualization method, 3D rose glyph. These methods provide efficient force representations alternative to haptics.
- Tangent continuous skin surfaces are implemented.
- For each protein, there are six different orthogonal views which helps user see colliding atoms with the aim of making the docking process easier. They are dynamic, and we are able to see each possible collision which may not be seen from the default view. These views enhance user's understanding of the process along with collision surface view.

## BIBLIOGRAPHY


[1] N. Akkiraju, H. Edelsbrunner, P. Fu and J. Qian. Viewing geometric protein structures from inside a CAVE. IEEE Comput. Graphics Applications 16, 58–61, 1996.

[2] A. Anderson and Z. Weng. VRDD: Applying virtual reality visualization to protein docking and design. Journal of Molecular Graphics and Modelling 17, 3, 180–186, 1999.

[3] H. M. Berman, J. Westbrook, Z. Feng, G. Gilliland, T. N. Bhat, H. Weissig, I. N. Shindyalov and P. E. Bourne. The protein data bank. Nucleic Acids Research 28, 235–242, 2000.

[4] CGAL. http://www.cgal.org/

[5] R. Chen, J. Mintseris, J. Janin and Z. Weng. A protein-protein docking benchmark. Proteins 52, 88–91, 2003.

[6] CMake. http://www.cmake.org/

[7] Code::Blocks. http://www.codeblocks.org/

[8] Connexions, Dimensionality Reduction Methods for Molecular Motion. http://cnx.org/content/m11461/latest/

[9] Cyberglove. http://www.cyberglovesystems.com/

[10] H. Edelsbrunner and P. Koehl. The geometry of biomolecular solvation. Combinatorial and Computational Geometry, 52, 243-275, 2005.

[11] N. Ferey, G. Bouyer, C. Martin, P. Bourdot, J. Nelson and J. M. Burkhardt. User needs analysis to design a 3d multimodal protein-docking interface. IEEE Symposium on 3D User Interfaces, 125–132, 2008.

[12] Flock of Birds. http://www.ascension-tech.com/realtime/RTflockofBIRDS.php

[13] T. N. Hart and R. J. Read. A multiple-start Monte Carlo docking method. Proteins: Structure, Function, and Genetics 13, 3, 206–222, 2004.

[14] Intel MKL. http://software.intel.com/en-us/intel-mkl/

[15] E. Katchalski-Katzir, I. Shariv, M. Eisenstein, A. A. Friesem, C. Aflalo and I. A. Vakser. Molecular surface recognition: Determination of geometric fit between proteins and their ligands by correlation techniques. Proceedings of the National Academy of Sciences 89, 2195–2199, 1992.

[16] G. Launay, R. Mendez, S. Wodak and T. Simonson. Recognizing protein-protein interfaces with empirical potentials and reduced amino acid alphabets. BMC Bioinformatics, 8, 270, 2007

[17] D. Levine, M. Facello, P. Hallstrom, G. Reeder, B. Walenz and F. Stevens. Stalk: An interactive system for virtual molecular docking. Proceedings of IEEE Computational Science and Engineering 4, 2, 55–65, 1997.

[18] H. A. Meier, M. Schlemmer, C. Wagner, A. Kerren, H. Hagen, E. Kuhl and P. Steinmann. IEEE Transactions on Visualization and Computer Graphics, 14, 5, Sept.-Oct. 2008

[19] Microsoft Visual Studio. http://msdn.microsoft.com/en-us/vstudio/default.aspx

[20] G. M. Morris, D. S. Goodsell, R. S. Halliday, R. Huey, W. E. Hart, R. K. Belew and A. J. Olson. Automated docking using a Lamarckian genetic algorithm and an empirical binding free energy function. Journal of Computational Chemistry 19, 1639–1662, 1999.

[21] H. Nagata, H. Mizushima and H. Tanaka. Concept and prototype of protein-ligand docking simulator with force feedback technology. Bioinformatics 18, 140–146, 2001.

[22] F. Nightingale. Notes on Matters Affecting the Health, Efficiency, and Hospital Administration of the British Army. Harrison & Sons, 1858.

[23] OpenGL. http://www.opengl.org/

[24] Peggleston-Bioreview, Life Molecules. https://peggleston-bioreview.wikispaces.com/Life+Molecules?f=print

[25] E. Subasi and C. Basdogan. A new haptic interaction and visualization approach for rigid molecular docking in virtual environments. Presence: Teleoperators and Virtual Environments, MIT Press 17, 73–90, 2008.

[26] Structural Biology Labs, Biomedical Centre, Uppsala, Sweden. Introduction to Swiss Pdb Viewer: The structure levels in proteins. http://xray.bmc.uu.se/~kurs/BiostrukfunkX2/practicals/practical_1/practical_1.html