Structural Prediction of Flexible Molecular Interactions

THESIS SUBMITTED FOR THE DEGREE OF
“DOCTOR OF PHILOSOPHY”

by

Efrat Farkash

The research work in this thesis has been carried out under the supervision of

Prof. Haim J. Wolfson and Prof. Ruth Nussinov.

Submitted to the Senate of Tel Aviv University

2012
Abstract

Most of the activities of living cells are performed by protein-protein interactions, which form molecular complexes. Structural details of molecular interactions are invaluable for understanding and deciphering biological mechanisms. Computational docking methods aim to predict the structure of such complexes given the structures of their single components. Dealing with the flexibility of proteins poses a great challenge in the docking field. In this thesis we have developed new methods that face this challenge.

FiberDock is a new method for flexible refinement of rigid docking solutions. The method models both backbone and side-chain flexibility and refines the interaction between the proteins in each docking solution. The backbone flexibility is modeled by a new normal-mode based approach which uses both low and high frequency modes and therefore is able to model both global and local movements. The side-chain movements are modeled by a linear programming approach which chooses the optimal conformation for each interface residue from a rotamer library. The FiberDock method also re-ranks the refined solutions by an energy based scoring function, for identifying the near native models.

Symmetric protein complexes are abundant in the living cell. The SymmRef method was developed to refine and re-rank docking solutions of symmetric multimers. The method uses symmetry constraints that reduce the search space and thus improve the accuracy and ranking of the results. Both FiberDock and SymmRef were tested on a benchmark of unbound docking challenges. The results show that the methods significantly improve the accuracy and the ranking of rigid docking solutions. Moreover, they outperform existing state-of-the-art methods.

FiberDock and SymmRef were incorporated into our full docking protocol which combines many methods which were developed in our lab over the years. We have analyzed and tested the full docking protocol on a large benchmark and on recent targets of the CAPRI competition, which simulates realistic and diverse blind docking challenges. Our analysis has demonstrated once again, the significant contribution of the refinement and re-ranking stage in the docking protocol.
# Table of Contents

Abstract .................................................................................................................................................. 3  
Acknowledgements .......................................................................................................................... 8  
Preface .................................................................................................................................................. 9  
Chapter 1: Introduction .......................................................................................................................... 10  
  1.1 Biological Background and Motivation ....................................................................................... 10  
  1.2 Challenges of flexible docking .................................................................................................... 11  
  1.3 General Scheme of Flexible Docking .......................................................................................... 12  
Chapter 2: Existing Methods for Flexible Docking.............................................................................. 14  
  2.1 Protein Flexibility Analysis ......................................................................................................... 14  
     2.1.1 Conformational ensemble analysis ....................................................................................... 14  
     2.1.2 Molecular dynamics .............................................................................................................. 15  
     2.1.3 Normal modes ....................................................................................................................... 16  
     2.1.4 Essential dynamics ................................................................................................................. 18  
     2.1.5 Rigidity theory ........................................................................................................................ 19  
  2.2 Handling Backbone Flexibility in Docking Methods ..................................................................... 19  
     2.2.1 Soft interface ........................................................................................................................ 20  
     2.2.2 Ensemble docking ................................................................................................................. 20  
     2.2.3 Modeling hinge motion ......................................................................................................... 21  
     2.2.4 Refinement and minimization methods for treating backbone flexibility ......................... 22  
  2.3 Handling Side-Chain Flexibility in Docking Methods .................................................................... 23  
     2.3.1 Global optimization algorithms for side-chain refinement ................................................. 24  
     2.3.2 Heuristic methods for treating side-chain flexibility .............................................................. 27  
  2.4 Handling Both Backbone and Side-Chain Flexibility in Recent Capri Challenges .................... 28  
  2.4 Discussion .................................................................................................................................... 29  
Chapter 3: FiberDock - Flexible induced-fit backbone refinement in molecular docking ............... 34  
  3.1 Methods ....................................................................................................................................... 34  
     3.1.1 Normal mode analysis .......................................................................................................... 36  
     3.1.2 Correlation measurement ..................................................................................................... 37  
     3.1.3 Minimization according to normal modes ............................................................................ 38  
     3.1.4 Applying a normal mode on a protein .................................................................................. 39
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1.5 The scoring function of the backbone refinement stage</td>
<td>40</td>
</tr>
<tr>
<td>3.1.6 Ranking according to an approximation of the energy function</td>
<td>40</td>
</tr>
<tr>
<td>3.1.7 RMSD calculations</td>
<td>40</td>
</tr>
<tr>
<td>3.1.8 Test cases</td>
<td>41</td>
</tr>
<tr>
<td>3.2 Results</td>
<td>41</td>
</tr>
<tr>
<td>3.2.1 Docking refinement starting from known binding orientation and</td>
<td>42</td>
</tr>
<tr>
<td>unbound conformation of the proteins</td>
<td></td>
</tr>
<tr>
<td>3.2.2 Docking refinement starting from random orientations of the</td>
<td>45</td>
</tr>
<tr>
<td>ligand around the native binding orientation</td>
<td></td>
</tr>
<tr>
<td>3.2.3 Local docking by FiberDock produces more accurate results than</td>
<td>49</td>
</tr>
<tr>
<td>RosettaDock</td>
<td></td>
</tr>
<tr>
<td>3.2.4 FiberDock improves the shape of energetic funnels around near-</td>
<td>50</td>
</tr>
<tr>
<td>native results</td>
<td></td>
</tr>
<tr>
<td>3.2.5 Docking refinement starting from rigid-body docking candidates</td>
<td>52</td>
</tr>
<tr>
<td>3.3 Discussion and Conclusions</td>
<td>55</td>
</tr>
<tr>
<td>3.4 FiberDock Web Server</td>
<td>56</td>
</tr>
<tr>
<td>3.4.1 Input</td>
<td>57</td>
</tr>
<tr>
<td>3.4.2 Output</td>
<td>58</td>
</tr>
<tr>
<td>Chapter 4: SymmRef - a Flexible Refinement Method for Symmetric</td>
<td>60</td>
</tr>
<tr>
<td>Multimers</td>
<td></td>
</tr>
<tr>
<td>4.1 Introduction to Symmetric Docking</td>
<td>60</td>
</tr>
<tr>
<td>4.2 Methods</td>
<td>63</td>
</tr>
<tr>
<td>4.2.1 Side-chain optimization</td>
<td>64</td>
</tr>
<tr>
<td>4.2.2 Rigid-body Monte-Carlo minimization</td>
<td>65</td>
</tr>
<tr>
<td>4.2.3 Backbone refinement</td>
<td>67</td>
</tr>
<tr>
<td>4.2.4 Ranking</td>
<td>68</td>
</tr>
<tr>
<td>4.2.5 Dataset</td>
<td>68</td>
</tr>
<tr>
<td>4.2.6 Docking Evaluation</td>
<td>70</td>
</tr>
<tr>
<td>4.3 Results</td>
<td>70</td>
</tr>
<tr>
<td>4.3.1 Analysis of backbone and side-chain conformations of symmetric</td>
<td>70</td>
</tr>
<tr>
<td>complexes</td>
<td></td>
</tr>
<tr>
<td>4.3.2 The importance of symmetry constraints in docking refinement of</td>
<td>74</td>
</tr>
<tr>
<td>symmetric multimers</td>
<td></td>
</tr>
<tr>
<td>4.3.3 Bound Docking Experiments</td>
<td>75</td>
</tr>
<tr>
<td>4.3.4 Unbound Docking Experiments</td>
<td>80</td>
</tr>
<tr>
<td>4.3.5 Comparison of SymmRef to the refinement and rescoring by</td>
<td>82</td>
</tr>
<tr>
<td>RosettaDock</td>
<td></td>
</tr>
</tbody>
</table>
4.3.6 Comparison to other symmetric docking methods .......................................... 82
4.4 Summary .............................................................................................................. 85
Chapter 5 : Performance Evaluation of Our Full Docking Protocol ......................... 87
5.1 METHODS .......................................................................................................... 87
  5.1.1 Biological and bioinformatics research of the interacting proteins ............... 87
  5.1.2 Rigid or hinge bent flexible docking .............................................................. 88
  5.1.3 Flexible refinement and re-ranking ................................................................. 88
  5.1.4 Clustering and filtering .................................................................................. 89
  5.1.5 CAPRI participation ..................................................................................... 89
5.2 RESULTS ............................................................................................................ 89
  5.2.1 Target 29: Trm8/Trm82 tRNA guanin-N(7)-methyltransferase .................... 91
  5.2.2 Target 30: Rnd1-GTP bound to RBD dimer ................................................... 91
  5.2.3 Target 32: Protease savinase bound to Bi-functional inhibitor BASI ............ 91
  5.2.4 Targets 33-34: Rlma2 methyltransferase bound to its RNA substrate .......... 92
  5.2.5 Targets 35-36: Xylanase Xyn10B .................................................................. 92
  5.2.6 Target 37: G-protein Arf6 bound to Leucine zipper of JIP4 ......................... 92
  5.2.7 Targets 38-39: Centaurin-α1 bound to FHA domain of KIF13B ................. 93
  5.2.8 Target 40: A complex of Trypsin and protease inhibitor ............................. 93
  5.2.9 Target 41: Colicin E9 bound to Im2 .............................................................. 93
  5.2.10 Target 42: TPR repeat dimer ..................................................................... 94
  5.2.11 Blind docking experiment ......................................................................... 94
5.3 DISCUSSION ...................................................................................................... 95
Chapter 6 : Conclusions ......................................................................................... 97
References ............................................................................................................... 99
Acknowledgements

During my PhD studies I had the privilege to work with several gifted and inspiring scientists whom I would like to thank. First, I want to thank my thesis advisors, Haim Wolfson and Ruth Nussinov, for their guidance and support. They have exposed me to the interesting and challenging world of protein-protein docking, believed in me and guided me along my research. I also wish to thank the friends and colleagues from the structural bioinformatics group of the Tel Aviv University who were always there to teach, help and support me. I would like to especially thank Dina Schneidman and Nelly Andrusier who were always happy to share their docking expertise, to guide and to help me in the early stages of my research.

I wish to emphasize my appreciation for the support of my research to the Edmond J. Safra Bioinformatics Program at Tel-Aviv University, the Adams fellowship of the Israel Academy of Sciences and Humanities, Wolf Foundation and the Google Anita Borg Memorial Scholarship.

Finally, I would like to thank my wonderful husband, Michael Farkash, for supporting and encouraging me throughout my PhD studies.
Preface

This thesis is based on the following collection of papers that were published in scientific journals during my PhD studies.


   * - Equal contribution.


Chapter 1: Introduction

1.1 Biological Background and Motivation

Most cellular processes are carried out by protein-protein interactions. Revealing the 3D structures of protein–protein complexes (docking) can shed light on their functional mechanisms and roles in the cell. It is important for understanding signaling pathways and for evaluating the affinity of protein-protein interactions. Furthermore, the structures of the complexes provide information regarding the interfaces of the proteins and can assist in drug design, enabling us to discover small molecules that inhibit or induce protein interactions.

In some cases, the 3D structure of protein–protein complexes can be determined experimentally by X-ray crystallography or NMR spectroscopy. However, it is an extremely difficult and time-consuming task. Therefore, the ability to predict the structure of complexes by computational means is essential.

Proteins are made of polypeptide chains. Each chain consists of a sequence of amino acids. The amino acid sequence is unique to a protein, and defines its structure and function. Each amino acid type has unique physical and chemical properties and a set of statistically common 3D conformations that it tends to adopt, called rotamers. Rotamer libraries contain the rotamers of each amino acid and are often used for predicting protein structure or interaction between proteins.

Each amino acid is composed of a backbone segment, common to all amino acid types, and a side-chain, also known as residue or R group, which distinguishes between amino acid types. Peptide bonds link the backbone segments of the amino acids within a protein and form the protein backbone (see Figure 1.1). The peptide bond is rigid and planar, with a dihedral angle (ω) that is close to 180°. The ϕ and ψ dihedral angles can have a certain range of possible values which determine the possible 3D conformations that the protein can adopt.

![Figure 1.1](https://i.imgur.com/3J2Q5.png)

**Figure 1.1.** (a) Two separate amino acids. The circled atoms form a water molecule when a peptide bond is formed. (b) A peptide bond. The ω dihedral angle is rigid and planar and the ϕ and ψ angles are flexible. The figure was taken from wikipedia.org
Proteins are flexible entities. This flexibility is reflected in the conformational variation shown in different crystallized 3D structures of the same protein. Many proteins were crystallized when interacting with another protein in a bound conformation, and by themselves in an unbound conformation. By comparing the 3D structures of a protein in its bound and unbound conformations, one can see conformational changes in both the side chains and the backbone [1]. Backbone flexibility can be divided into two major types: large-scale domain motions, such as shear and hinge-bending motion [Figure 1.2(a–d)], and disordered regions such as flexible loops [Figure 1.2(e)]. There are two main biological models that explain the structural differences between bound and unbound conformations of proteins. The first is called the conformational selection model [2,3,4,5,6]. According to this model, proteins constantly change conformations, and when, by chance, a protein in its bound conformational state, encounters a complementary molecule, they interact and create a complex. The second model is called the induced-fit model [7,8]. This model postulates that the structures of the receptor and the ligand are partially compatible, and when they come into proximity of each other, the chemical forces created during their interaction induce their conformational changes. In nature, both models are likely to hold [9]. The binding process begins with conformational selection, followed by an induced fit, which likely plays a role in local side chain and relatively minor backbone changes to optimize the association [10].

1.2 Challenges of flexible docking

In docking, our goal is to predict the structure of a complex of two (or more) biological molecules, often called receptor and ligand, given their unbound conformations. The first docking methods treated proteins as rigid bodies in order to reduce the search space for optimal structures of the complexes [11,12]. However, predicting only the rigid transformation, which places the unbound ligand on the interaction interface of the unbound receptor in the native orientation, is not sufficient. Ignoring flexibility could prevent docking algorithms from...
recovering native associations. The resulting model of the complex will often contain major steric clashes. Consequently, the calculated binding energy value of this near-native model will be very high and it may not be identified among a group of docking solution candidates. Additionally, the accuracy of such a model will often be poor, as without modeling the conformational changes of the proteins, the native chemical interactions, which are important for the complex formation, will not be attained in the model. In addition, flexibility must also be taken into account if the docked structures were determined by homology modeling [13] or if loop conformations were modeled [14]. Therefore, docking methods must model the conformational changes that proteins undergo upon binding, including both backbone and side-chain movements.

Incorporating flexibility in a docking algorithm is much more difficult than performing rigid-body docking. The high number of degrees of freedom not only significantly increases the running time, but also results in a higher rate of false-positive solutions. These must be scored correctly in order to identify near-native results [9]. In order to reduce the number of degrees of freedom, existing docking methods limit the modeled flexibility to certain types of motions. In addition, many of these methods allow only one of the proteins in the complex to be flexible.

### 1.3 General Scheme of Flexible Docking

The general scheme of flexible docking can be divided into four major stages as depicted in Figure 1.3. The first is a preprocessing stage. In this stage the proteins are analyzed in order to define their conformational space. An ensemble of discrete conformations can be generated from this space and used in further cross-docking, where each protein conformation is docked separately. This process simulates the conformational selection model [4,5]. The analysis can also identify possible hinge locations. In this case the proteins can be divided into their rigid parts and be docked separately. The second is a rigid-docking stage. The docking procedure aims to generate a set of solution candidates with at least one near-native structure. The rigid docking should allow some steric clashes because proteins in their unbound conformation can collide when placed in their native interacting position. The next stage, called refinement, models an induced fit [8], resolves the clashes and improves the shape complementarity of the proteins. In this stage, each candidate is optimized by small backbone and side-chain movements and by rigid-body adjustments. It is difficult to simultaneously optimize the side-chain conformations, the backbone structure and the rigid-body orientation. Therefore, the three can be optimized in three separately repeated successive steps. The resulting refined structures have better binding energy and hardly include steric clashes. The final stage is scoring. In this stage the candidate solutions are scored and ranked according to different parameters such as binding energy, agreement with known binding sites, deformation energy of the flexible proteins, and existence of energy funnels [15,16]. The goal of this important stage is to identify the near-native solutions among the candidates.
Figure 1.3. A general scheme of flexible docking procedure. The figure was taken from Andrusier et al. 2008.
Chapter 2:
Existing Methods for Flexible Docking

Treating flexibility in molecular docking is a major challenge in cell biology research. Here we describe the background and the principles of existing flexible protein–protein docking methods, focusing on the algorithms and their rational. We describe how protein flexibility is treated in different stages of the docking process: in the preprocessing stage, rigid and flexible parts are identified and their possible conformations are modeled. This preprocessing provides information for the subsequent docking and refinement stages. In the docking stage, an ensemble of pre-generated conformations or the identified rigid domains may be docked separately. In the refinement stage, small-scale movements of the backbone and side-chains are modeled and the binding orientation is improved by rigid-body adjustments. For clarity of presentation, we divide the different methods into categories. This should allow the reader to focus on the most suitable method for a particular docking problem.

2.1 Protein Flexibility Analysis

Protein flexibility analysis methods, reviewed below, can be classified into three major categories:

1. Methods for generating an ensemble of discrete conformations. Ensembles of conformations are widely used in cross-docking and in the refinement stage of the docking procedure. The different conformations can be created by analyzing different experimentally solved protein structures or by using Molecular Dynamics (MD) simulation snapshots.

2. Methods for determining a continuous protein conformational space. The conformational space can be used as a continuous search space for refinement algorithms. In addition, many flexible docking methods sample this pre-calculated conformational space in order to generate a set of discrete conformations. This group of methods includes Normal Modes Analysis (NMA) and Essential Dynamics.

3. Methods for identifying rigid and flexible regions in the protein. These methods include the rigidity theory and hinge detection algorithms.

2.1.1 Conformational ensemble analysis

Using different solved 3D structures (by X-ray and NMR) of diverse conformations of the same protein, or of homologous proteins, is probably the most convenient way to obtain information relating to protein flexibility. Using such conformers, one can generate new viable conformations which might exist during the transition between one given conformation to another. These new conformations can be generated by ‘morphing’ techniques [17,18] which implement linear interpolations, but have limited biological relevance.
Known structures of homologs or of different conformations of the same protein can also be useful in detecting rigid domains and hinge locations. Boutonnet et al. [19] developed one of the first methods for an automated detection of hinge and shear motions in proteins. The method uses two conformations of the same protein. It identifies structurally similar segments and aligns them. Then, the local alignments are hierarchically clustered to generate a global alignment and a clustering tree. Finally the tree is analyzed to identify the hinge and shear motions. The DynDom method [20] uses a similar clustering approach for identifying hinge points using two protein conformations. Given the set of atom displacement vectors, the rotation vectors are calculated for each short backbone segment. A rotation vector can be represented as a rotation point in a 3D space. A domain that moves as a rigid-body will produce a cluster of rotation points. The method uses the K-means clustering algorithm to determine the clusters and detect the domains. Finally, the hinge axis is calculated and the residues involved in the inter-domain bending are identified.

The HingeFind [21] method can also analyze structures of homolog proteins in different conformations and detect rigid domains, whose superimposition achieves RMSD of less than a given threshold. It requires sequence alignment of two given protein structures. The procedure starts with each pair of aligned Ca atoms and iteratively tries to extend them by adding adjacent Ca atoms as long as the RMSD criterion holds. After all the rigid domains are identified, the rotation axes between them are calculated. Verbitsky et al. [22] used the geometric hashing approach to align two molecules, and detect hinge-bent motifs. The method can match the motifs independently of the order of the amino acids in the chain. The more advanced FlexProt method [23,24] searches for 3D congruent rigid fragment pairs in structures of homolog proteins, by aligning every Ca pair and trying to extend the 3D alignment, in a way similar to HingeFind. Next, an efficient graph-theory method is used for the connection of the rigid parts and the construction of the full solution with the largest correspondence list, which is sequence-order dependent. The construction simultaneously detects the locations of the hinges.

2.1.2 Molecular dynamics

Depending on the time scales and the energy barrier heights, molecular dynamics simulations can provide insight into protein flexibility. Molecular dynamics (MD) simulations are based on a force field that describes the forces created by chemical interactions. Throughout the simulation, the motions of all atoms are modeled by repeatedly calculating the forces on each atom, solving Newton’s equation and moving the atoms accordingly. Di Nola et al. [25,26] were first to incorporate explicit solvent molecules into MD simulations while docking two flexible molecules. Pak et al. [27] applied MD using Tsallis effective potential [28] for the flexible docking of few complexes.

Molecular dynamics simulations require long computational time scales and therefore are limited in the motion amplitudes. For this reason they can be used for modeling only relatively small-scale movements, which take place in nanosecond time scales, while conformational
changes of proteins often occur over a relatively long period of time (~1 ms) [29,30]. One way to speed up the simulations is by restricting the degrees of freedom to the torsional space, which allows larger integration time steps [31]. Another difficulty is that the existence of energy barriers may trap the MD simulation in certain conformations of a protein. This problem can be overcome by using simulated annealing [32] and scaling methods [33] during the simulation. For example, simulated annealing MD is used in the refinement stage of HADDOCK [34,35] in order to refine the conformations of both the side-chains and the backbone. Riemann et al. applied potential scaling during MD simulations to predict side chain conformations [36].

In order to sample a wide conformational space and search for conformations at local minima in the energy landscape, biased methods, which were previously reviewed [37], can be used. The flooding technique [38], which is used in the GROMACS method [39], fills the ‘‘well’’ of the initial conformation in the energy landscape with a Gaussian shape ‘‘flooding’’ potential. Another similar method, called puddle-jumping [40], fills this well up to a flat energy level. These methods accelerate the transition across energy barriers and permit scanning other stable conformations.

### 2.1.3 Normal modes

Normal Modes Analysis (NMA) is a method for calculating a set of basis vectors (normal modes) which describes the flexibility of the analyzed protein [41,42,43]. The length of each vector is 3N, where N is the number of atoms or amino acids in the protein, depending on the resolution of the analysis. Each vector represents a certain movement of the protein such that any conformational change can be expressed as a linear combination of the normal modes. The coefficient of a normal mode represents its amplitude.

A common model used for normal modes calculation is the Anisotropic Network Model (ANM) [42,44]. This is a simplified spring model which relies primarily on the geometry and mass distribution of a protein (Figure 2.1). Every two atoms (or residues) within a distance below a threshold are connected by a spring (usually all springs have a single force constant). The model treats the initial conformation as the equilibrium conformation.

The normal modes describe continuous motions of the flexible protein around a single equilibrium conformation. Theoretically, this model does not apply to proteins which have several conformational states with local free energy minima. However, in practice, normal modes suit very well conformational changes observed between bound and unbound protein structures [45]. Another advantage of the normal modes analysis is that it can discriminate between low and high frequency modes. The low frequency modes usually describe the large scale motions of the protein. It has been shown [45,46] that the first few normal modes, with the lowest frequencies, can describe much of a conformational change. This allows reducing the degrees of freedom considerably while preserving the information about the main characteristics of the protein motion. Therefore, many studies [47,48,49] use a subset of the
lowest frequency modes for analyzing the flexibility of proteins. The normal modes can further be used for predicting hinge-bending movements [50], for generating an ensemble of discrete conformations [51] and for estimating the protein’s deformation energy resulting from a conformational change [52,53].

Tama and Sanejouand [46] showed that normal modes obtained from the open form of a protein correlate better with its known conformational changes, than the ones obtained from its closed form. In a recent work, Petrone and Pande [45] showed that the first 20 modes can improve the RMSD to the bound conformation by only up to 50%. The suggested reason was that while the unbound conformation moves mostly according to low frequency modes, the binding process activates movements related to modes with higher frequencies.

The binding site of proteins often contains loops which undergo relatively small conformational changes triggered by an interaction. This phenomenon is common in protein kinase binding pockets. Loop movements can only be characterized by high-frequency normal modes. Therefore, we would like to identify the modes which influence these loops the most, in order to focus on these in the docking process. For this reason, Cavasotto et al. [54] have introduced a method for measuring the relevance of a mode to a certain loop. This measure of relevance favors modes which bend the loop at its edges, and significantly moves the center of the loop. It excludes modes which distort the loop or move the loop together with its surroundings. This measure was used to isolate the normal modes which are relevant to loops within the binding sites of two cAMP-dependent protein kinases (cAPKs). For each loop less than 10 normal modes were found to be relevant, and they all had relatively high frequencies. These modes were used for generating alternative conformations of these proteins, which were later used for docking. The method succeeded in docking two small ligands which could not be docked to the unbound conformations of the cAPKs due to steric clashes. In addition, binders identification was improved in a small-scale virtual screening.

Since NMA is based on an approximation of a potential energy in a specific starting conformation, its accuracy deteriorates when modeling large conformational changes. Therefore in some studies, the normal modes were recomputed after each small displacement.
This is an accurate but time-consuming method. Kirillova et al. [58] have recently developed an NMA-guided RRT method for exploring the conformational space spanned by 10–30 low frequency normal modes. This efficient method requires a relatively small number of normal modes calculations to compute large conformational changes.

The Gaussian Network Model (GNM) is another simplified version of normal modes analysis [59,60]. The GNM analysis uses the topology of the spring network for calculating the amplitudes of the normal modes and the correlations between the fluctuations of each pair of residues. However, the direction of each fluctuation cannot be found by GNM. This analysis is more efficient both in CPU time and in memory than the ANM analysis and therefore it can be applied on larger systems. The drawback is that the GNM calculates relatively partial information on the protein flexibility.

The HingeProt algorithm [50] analyzes a single conformation of a protein using GNM, and predicts the location of hinges and rigid parts. Using the two slowest modes, it calculates the correlation between the fluctuations of each pair of residues, that is their tendency to move in the same direction. A change in the sign of the correlation value between two consecutive regions in the protein suggests a flexible joint that connects rigid units.

**2.1.4 Essential dynamics**

The Essential dynamics approach aims at capturing the main flexible degrees of freedom of a protein, given a set of its feasible conformations [61]. These degrees of freedom are described by vectors which are often called essential modes, or principal components (PC). The conformation set is used to construct a square covariance matrix (3N X 3N, where N is the number of atoms) of the deviation of each atom coordinates from its unbound position or, alternatively, average position. This matrix is then diagonalized and its eigenvectors and eigenvalues are found. These eigenvectors represent the principal components of the protein flexibility. The bigger the eigenvalues, the larger the amplitude of the fluctuation described by its eigenvector.

Mustard and Ritchie [62] used this essential dynamics approach to generate realistic starting structures for docking, which are called eigenstructures. The covariance matrix was created according to a large number of conformations, generated using the CONCOORD program [63], which randomly generates 3D protein conformations that fulfill distance constraints. The eigenvectors were calculated and it has been shown [62] that the first few of these (with the largest eigenvalues) can account for many of the backbone conformational changes that occurred upon binding in seven different CAPRI targets from rounds 3–5 [64]. Linear combinations of the first eight eigenvectors were later used to generate eigenstructures from each original unbound structure of these CAPRI targets. An experiment that used these eigenstructures in rigid docking showed improvements in the results compared to using the unbound structure or a model-built structure [62]. An ensemble of conformations can be
generated in a similar way by the Dynamite software [65], which also applies the essential dynamics approach on a set of conformations generated by CONCOORD.

Principle component analysis (PCA) can also be based on molecular dynamics simulations. Unlike normal mode analysis, this PCA includes the effect of the surrounding water on the flexibility. However, the results of the analysis strongly depend on the simulation’s length and convergence. It has been shown that most of the conformational fluctuations observed by MD simulations [61] and some known conformational changes between unbound and bound forms [66], can be described with only few PCs.

2.1.5 Rigidity theory

Jacobs et al. [67] developed a graph-theory method which analyzes protein flexibility and identifies rigid and flexible substructures. In this method a network is constructed according to distance and angle constraints, which are derived from covalent bonds, hydrogen bonds and salt bridges within a single conformation of a protein. The vertices of the network represent the atoms and the edges represent the constraints. The analysis of the network resembles a pebble game. At the beginning of the algorithm, each atom (vertex) receives three pebbles which represent three degrees of freedom (translation in 3D). The edges are added one by one and each edge consumes one pebble from one of its vertices, if possible. It is possible to rearrange the pebbles on the graph as long as the following rules hold: (1) Each vertex is always associated with exactly three pebbles which can be consumed by some of its adjacent edges. (2) Once an edge consumes a pebble it must continue holding a pebble from one of its vertices throughout the rest of the algorithm. At the end of the algorithm the remaining pebbles can still be rearranged but the specified rules divide the protein into areas in such a way that the pebbles can not move from one area to another.

The number of remaining degrees of freedom in a certain area of the protein quantifies its flexibility. For example, a rigid area will not possess more than 6 degrees of freedom (which represent translation and rotation in 3D). The algorithm can also identify hinge points, and rigid domains which are stable upon removal of constraints like hydrogen-bonds and salt bridges. This pebbles-game algorithm is implemented in a software called FIRST (Floppy Inclusion and Rigid Substructure Topography) which analyzes protein flexibility in only a few seconds of CPU time. The algorithm was tested on HIV protease, dihydrofolate reductase and adenylate kinase and was able to predict most of their functionally important flexible regions, which were known beforehand by X-ray and NMR experiments.

2.2 Handling Backbone Flexibility in Docking Methods

Treating backbone flexibility in docking methods is still a major challenge. The backbone flexibility adds a huge number of degrees of freedom to the search space and therefore makes the docking problem much more difficult. The docking methods can be divided into four groups according to their treatment of backbone flexibility. The first group uses soft interface during
the docking and allows some steric clashes in the resulting complex models. The second performs an ensemble docking, which uses feasible conformations of the proteins, generated beforehand. The third group deals with hinge bending motions, and the last group heuristically searches for energetically favored conformations in a wide conformational search space.

2.2.1 Soft interface

Docking methods that use soft interface actually perform relatively fast rigid-body docking which allows a certain amount of steric clashes (penetration). These methods can be divided into three major groups: (i) brute force techniques [68,69,70] that can be significantly speeded up by FFT [71,72,73,74,75,76], (ii) randomized methods [15,77] and (iii) shape complementarity methods [78,79,80,81,82]. This approach can only deal with side chain flexibility and small scale backbone movements. It is assumed that the proteins are capable of performing the required conformational changes which avoid the penetrations, although the actual changes are not modeled explicitly. Since the results of this soft docking usually contain steric clashes, a further refinement stage must be used in order to resolve them.

2.2.2 Ensemble docking

In order to avoid the search through the entire flexible conformational space of two proteins during the docking or refinement process, the ensemble docking approach samples an ensemble of different feasible conformations prior to docking. Next, docking of the whole ensemble is performed. The different conformations can be docked one by one (cross-docking), which significantly increases the computational time, or all together using different algorithms such as the mean-field approach presented below.

The ensemble may include different crystal structures and NMR conformers of the protein. Other structures can be calculated using computational sampling methods which are derived from the protein flexibility analysis (molecular dynamics, normal modes, essential dynamics, loop modeling, etc). Feasible structures can also be sampled using random-search methods such as Monte Carlo and genetic algorithms.

The search for an optimal loop conformation can be performed during the docking procedure by the mean-field approach. In this method, a set of loop conformations is sampled in advance and each conformation is initialized by an equal weight. Throughout the docking, in each iteration, the weights of the conformations (copies) change according to the Boltzmann criterion, in a way that a conformation receives a higher weight if it achieves a lower free energy. The partner and the rest of the protein which interact with the loop ‘feel’ the weighted average of the energies of their interactions with each conformation in the set. The algorithm usually converges to a single conformation for each loop, with a high weight [83].

Bastard et al. [83] used the mean-field approach in their MC2 method which is based on multiple copy representation of loops and Monte Carlo (MC) conformational search. Viable loop
conformations were created using a combinatorial approach, which randomly selected common torsional angles for the loop backbone. In each MC step the side-chains dihedral angles and the rotation and translation variables are randomly chosen. Then, the weight of each loop is adjusted according to its Boltzmann probability. The performance of the MC2 algorithm was evaluated [83] on the solved protein-DNA complex of a Drosophila Prd-paired-domain protein, which interacts with its target DNA segment by a loop of seven residues [84]. 23% of the MC2 simulations produced results in which the RMSD was lower than 1.5 Å, and included the selection of a loop conformation which was extremely similar to the native one. Furthermore, these results got much better energy scores than the other 77%, therefore they could be easily identified.

In a later work [85], the mean-field approach was introduced in the ATTRACT software and was tested on a set of eight protein–protein complexes in which the receptor undergoes a large conformational change upon binding or its solved unbound structure has a missing loop at its interaction site. The results showed that the algorithm improved the docking significantly compared to rigid docking methods.

### 2.2.3 Modeling hinge motion

Hinge-bending motions are common during complex creation. Hinges are flexible segments which separate relatively rigid parts of the proteins, such as domains or subdomains.

Sandak et al. [86,87] introduced a method which deals with this type of flexibility. The algorithm allows multiple hinge locations, which are given by the user. Hinges can be given for only one of the interacting proteins (e.g. the ligand). The algorithm docks all the rigid parts of the flexible ligand simultaneously, using the geometric hashing approach.

The FlexDock algorithm [88,89] is a more advanced method for docking with hinge-bending flexibility in one of the interacting proteins. The locations of the hinges are automatically detected by the HingeProt algorithm. The number of hinges is not limited and does not affect the running time complexity. However, the hinges must impose a chain-type topology, that is the subdomains separated by the hinges must form a linear chain. The algorithm divides the flexible protein into subdomains at its hinge points. These subdomains are docked separately to the second protein by the PatchDock algorithm [79]. Then, an assembly graph is constructed. Each node in the graph represents a result of a subdomain rigid docking (a transformation), and the node is assigned a weight according to the docking score. Edges are added between nodes which represent consistent solutions of consecutive rigid subdomains. An edge weight corresponds to the shape complementarity score between the two subdomains represented by its two nodes. Finally, the docking results of the different subdomains are assembled to create full consistent results for the complex using an efficient dynamic programming algorithm that finds high scoring paths in the graph. This approach can cope with very large conformational changes. Among its achievements, it has predicted the bound conformation of calmodulin to a target peptide, the complex of Replication Protein A with a single stranded DNA as shown in
and has created the only acceptable solution for the LicT dimer at the CAPRI challenge (Target 9) [88].

Ben-Zeev et al. [90] have coped with the CAPRI challenges which included domain movements (Target 9, 11, and 13) by a rigid body multi-stage docking procedure. Each of the proteins was partitioned into its domains. Then, the domains of the two proteins were docked to each other in all possible order of steps. In each step, the current domain was docked to the best results from the previous docking step.

This multistage method requires that the native position of a subdomain will be ranked high enough in each step. This restriction is avoided in the FlexDock algorithm which in the assembling stage uses a large number of docking results for each subdomain. Therefore, a full docking result can be found and be highly ranked even if its partial subdomain docking results were poorly ranked.

2.2.4 Refinement and minimization methods for treating backbone flexibility

Fitzjohn and Bates [91] used a guided docking method, which includes a fully flexible refinement stage. In the refinement stage CHARMM22 all-atom force field was used to move the individual atoms of the receptor and the ligand. In addition, the forces acting on each atom were summed and converted into a force on the center-of-mass of each molecule.

Lindahl and Delarue introduced a new refinement method [92] for docking solutions which minimizes the interaction energy in a complex along 5–10 of the lowest frequency normal modes’ directions. The degrees of freedom in the search space are the amplitudes of the
normal modes, and a quasi-Newtonian algorithm is used for the energy minimization. The method was tested on protein-ligand and protein-DNA complexes and was able to reduce the RMSD between the docking model and the true complex by 0.3Å–3.6Å.

In a recent work, May and Zacharias [53] accounted for global conformational changes during a systematic docking procedure. The docking starts by generating many thousands of rigid starting positions of the ligand around the receptor. Then, a minimization procedure is performed on the six rigid degrees of freedom and on five additional degrees of freedom which account for the coefficients of the five, pre-calculated, slowest frequency normal modes. The energy function includes a penalty term that prevents large scale deformations. Applying the method to several test cases showed that it can significantly improve the accuracy and the ranking of the results. However the side-chain conformations must be refined as well. The method was recently incorporated into the ATTRACT docking software.

A new data structure called Flexibility Tree (FT) was recently presented by Zhao et al. [93]. The FT is a hierarchical data structure which represents conformational subspaces of proteins and full flexibility of small ligands. The hierarchical structure of this data structure enables focusing solely on the motions which are relevant to a protein binding site. The representation of protein flexibility by FT combines a variety of motions such as hinge bending, flexible side-chain conformations and loop deformations which can be represented by normal modes or essential dynamics. The FT parameterizes the flexibility subspace by a relatively small number of variables. The values of these variables can be searched, in order to find the minimal energy solution. The FLIPDock [94] method uses two FT data structures, representing the flexibility of both the ligand and the receptor. The right conformations are then searched using a genetic algorithm and a divide and conquer approach, during the docking process.

Many docking methods use Monte Carlo methods in the final minimization step. For example, Monte Carlo minimization (MCM) is used in the refinement stage of RosettaDock [95,96]. Each MCM iteration consists of three steps: (1) random rigid-body movements and backbone perturbation, in certain peptide segments which were chosen to be flexible according to a flexibility analysis performed beforehand; (2) rotamer-based side-chain refinement; (3) quasi-Newton energy minimization for relatively small changes in the backbone and side-chain torsional angles, and for minor rigid-body optimization.

Some docking methods [95] simply ignore flexible loops during the docking and rebuild them afterwards in a loop modeling [14,97] step.

### 2.3 Handling Side-Chain Flexibility in Docking Methods

The majority of the methods handle side-chain flexibility in the refinement stage. Each docking candidate is optimized by side-chain movements. Figure 2.3 shows a non-optimized conformation of a ligand residue, which clashes with the receptor’s interface, and a correct prediction of its bound conformation by a side-chain optimization algorithm. Most
conformational changes occur in the interface between the two binding proteins. Therefore, many methods try to predict side-chain conformational changes for a given backbone structure in the interaction area. The problem has been widely studied in the more general context of side-chain assignment on a fixed backbone in the fields of protein design and homology modeling. Therefore, all the algorithms reviewed in this section apply to side-chain refinement in both folding and docking methods.

To reduce the search space, most of the methods use rotamer discretization. Rotamer libraries are derived from statistical analysis of side-chain conformations in known high-resolution protein structures. Backbone-dependent rotamer libraries contain information on side-chain dihedral angles and rotamer populations dependent on the backbone conformation [98]. Usually, unbound conformations of side-chains are added to the set of conformers for each residue. In this way a side-chain can remain in its original state if the unbound conformer is chosen by the optimization algorithm.

2.3.1 Global optimization algorithms for side-chain refinement

The side-chain prediction problem can be treated as a combinatorial optimization problem. The goal is to find the combination of rotamer assignments for each residue with the global minimal energy, denoted as GMEC (Global Minimal Energy Conformation). The energy value of GMEC is calculated as follows:

\[
E_{\text{GMEC}} = \min_{r,i} \left( \sum_i E(i_r) + \sum_{i,j} E(i_r, j_s) \right)
\]

where \( E(i_r) \) is the self energy of the assignment of rotamer \( r \) for residue \( i \). It includes the interaction energy of the rotamer with a fixed environment. \( E(i_r, j_s) \) is the pair-wise energy between rotamer \( r \) of residue \( i \) and rotamer \( s \) of residue \( j \). For each residue one rotamer should be chosen, and the overall energy should be minimal. This combinatorial optimization problem was proved to be NP-hard [99] and inapproximable [100]. In practice, topological restraints of residues can facilitate the problem solution.

In branch-and-bound algorithms all possible conformations are represented by a tree. Each level of the tree represents a different residue and the order of the nodes at this level is the number of possible residue rotamers. Scanning down the tree and adding self and pairwise energies at each level will sum up to the global energy values at the leaves. A branch-and-bound algorithm can be performed by using a bound function [101,102]. A proposed bound function is defined for a certain level, and yields a lower bound of energy, obtainable from any branch below this.
level. This level bound function is added to the cumulative energy in the current scanned node and the branch can be eliminated if the value is greater than a previously calculated leaf energy.

The dead-end elimination (DEE) method \[103\] is based on pruning the rotamers, which are certain not to participate in GMEC, because better alternatives can be chosen. The Goldstein DEE \[104\] criterion removes a rotamer from further consideration if another rotamer of the same residue has a lower energy for every possible rotamer assignment for the rest of the residues. A more powerful criterion for dead-end elimination is proposed in the “split DEE” \[105,106\] method (Figure 2.4). Many methods use DEE as a first stage in order to reduce a conformational search space.

In addition to the rotamer reduction method by DEE, many methods also use a residue reduction procedure, which eliminates residues with a single rotamer or with up to two interacting residues (neighbors). A residue with a single rotamer can be eliminated from further consideration by incorporating its pairwise energies into the self energies of its neighbors \[102\]. A residue with one neighbor can be reduced by adding its rotamer energies to the self-energies of its neighbor’s rotamers \[102\]. A residue with two neighbors can be eliminated by updating the pair-wise energies of the neighbors \[107\]. The Residue-Rotamer-Reduction (R3) method \[108\] repeatedly performs residue and rotamer reduction. When a reduction is not possible in a certain iteration, the R3 method performs residue unification \[104\]. In this procedure, two residues are unified and a set of all their possible rotamer pairs is generated. The method finds the GMEC in a finite number of elimination iterations, because at least one residue is reduced in each iteration \[108\].

Bahadur et al. \[109\] have defined a weighted graph of non-colliding rotamers. In this graph the vertices are rotamers and two rotamers are connected by an edge if they represent different residues that do not have steric clashes. The weights on the edges correspond to the strength of the interaction between two rotamers. The algorithm searches for the maximum edge-weight clique in the induced graph. If the size of the obtained clique equals the number of residues, then each residue is assigned with exactly one rotamer. Since each two nodes in the clique are connected, none of the chosen rotamers collide. Thus, the obtained clique defines a feasible conformation and the maximum edge-weight clique corresponds to the GMEC.

The SCWRL \[102\] algorithm uses a residue interaction graph in which residues with clashing rotamers are connected. The resulting graph is decomposed to biconnected components [see Figure 2.5(a,b)] and a dynamic programming technique is applied to find a GMEC. Any two components include at most one common residue – the articulation point. It starts by
optimizing the leaf components, which have only one articulation point. The component’s GMEC is calculated for each rotamer of the articulation point and is stored as the energy of the compatible rotamer for further GMEC calculations of adjacent components. Figure 2.5(a,b) demonstrate the decomposition of a residue interaction graph into components. The drawback of the method is that it might include large components, which increase dramatically the CPU time. SCATD [107] proposes an improvement of the SCWRL methodology by using a tree decomposition of the residue interaction graph. This method results in more balanced decomposition and prevents creation of huge components, as opposed to biconnected decomposition. After this decomposition, any two components can share more than one common residue [Figure 2.5(c)]. Therefore, the component GMEC is calculated for every possible combination of these common residues and stored for further calculations.

Recent methods use the mixed-integer linear programming (MILP) framework [110,111,112,113] to find a GMEC. In general, a decision variable is defined for each rotamer and rotamer-rotamer interaction. If a rotamer participates in GMEC, its corresponding decision variables will be equal to 1. Each decision variable is weighted by its score (self and pair-wise energies) and summed in a global linear expression for minimization. Constraints are set in order to guarantee one rotamer choice for each residue, and that only pair-wise energies between the selected rotamers are included in the global minimal energy. Although the MILP algorithm is NP-hard, by relaxation of the integrality condition on the decision variables, the polynomial-complexity linear programming algorithm can be applied to find the minimum. If the solution happens to be integral, the GMEC is found in polynomial time. Otherwise, an integer linear programming algorithm, with significantly longer running time, is applied. The MILP framework allows obtaining successive near-optimal solutions by addition of constraints that exclude the previously found optimal set of rotamers [112]. The FireDock [113] method for refinement and scoring of docking candidates uses the MILP technique for side-chain optimization. An example of a successful rotamer assignment by FireDock is shown in Figure 2.3.

In general, for all methods which use pair-wise energy calculations, a prefix tree data structure (trie) can be used for saving CPU time [114]. In a trie data structure, the inter-atomic energies of rotamers’ parts, which share the same torsion angles, are computed once.

Many of the described methods efficiently find a GMEC due to the use of a simplified energy function, which usually includes only the repulsive van der Waals and rotamer probability terms. The energy function can be extended by additional terms, like the attractive van der Waals, solvation and electrostatics. However, this complicates the problem. The SCWRL/SCATD graph decomposition results in larger components, the number of decision variables in the MILP
technique increases, and so forth. For example, Kingsford et al. [112] use only van der Waals and rotamer probability terms and almost always succeed in finding the optimal solution by polynomial LP. However, when adding electrostatic term, non-polynomial ILP is often required.

A performance comparison of R3 [108], SCWRL [102] and MILP [112] methods was performed. The first test set included 25 proteins [115]. The differences in the prediction ability of the methods were minor, since they all find a GMEC and use a similar energy function. The time efficiency of the R3 and SCWRL methods was better than of MILP for these cases. The second test set of 5 proteins was harder [102] and the R3 method performed significantly faster than SCWRL and MILP. In addition, Xu [107] demonstrated that the SCATD method shows a significant improvement in CPU time compared to SCWRL for the second test set.

2.3.2 Heuristic methods for treating side-chain flexibility

Heuristic algorithms are widely used in side-chain refinement methods because of the following reasons. First, a continuous conformational space can be used during the minimization, as opposed to global optimization algorithms, where the conformational space has to be reduced to a pre-defined discrete set of conformers. Second, different energy functions can be easily incorporated into heuristic algorithms, while global optimization methods usually require a simplified energy function. A third advantage is that heuristic algorithms can provide many low-energy solutions, while most of the global algorithms provide a single one. However, the main drawback of the heuristic methods is that they cannot guarantee finding the GMEC.

Monte Carlo (MC) [116] is an iterative method. At each step it randomly picks a residue and switches its current rotamer by another. The new overall energy is calculated and the conformational change is accepted or rejected by the Metropolis criterion [117]. In simulated annealing MC, the Boltzman temperature is high at the beginning to overcome local minima. Then, it is gradually lowered in order to converge to the global minimum. Finally, a quench step can be performed. The quench step cycles through the residues in a random order, and for each residue, the best rotamer for the overall energy is chosen. RosettaDock [118] uses this rotamer-based MC approach and, in addition, performs gradient-based minimization in torsion space of dihedral angles.

The self-consistent mean-field (SCMF) optimization method [119,120] uses a matrix which contains the probability of each rotamer to be included in the optimal solution. Each rotamer probability is calculated by the sum of its interaction energies with the surrounding rotamers, weighted by their respective probabilities. The method iteratively refines this matrix and converges in a few cycles. The 3D-DOCK [121] package uses the mean-field approach for side-chain optimization with surrounding solvent molecules.

Other optimization techniques like genetic algorithms [122] and neural networks [123] are also applied to predict optimal conformations of side-chains. Several methods do not restrict the conformational search to rotamers [124,125]. Abagyan et al. [125,126] (ICMDISCO [127,128])
apply the biased probability MC method for minimization in the torsion angles space. Molecular
dynamics simulations (described in Section Molecular dynamics) are also used to model
flexibility of sidechains. SmoothDock [16,129] uses short MD simulations to predict
conformations of anchor side-chains [130] at the pre-docking phase. HADDOCK [34,35] uses
restricted MD simulations for final refinement with explicit solvent.

Obviously, an energy function has great influence on side-chain prediction performance.
Yanover et al. [131] showed that finding a GMEC does not significantly improve side-chain
prediction results compared to the heuristic RosettaDock side-chain optimization. They showed
that using an optimized energy function has much greater influence on the performance than
using an improved search strategy.

Recent studies have shown that most of the interface residues do not undergo significant
changes during binding [66,113,130,132]. Therefore, changing unbound conformations should
be performed carefully during the optimization process [113,118]. In addition, when analyzing
the performance of side-chain optimization methods, unbound conformations of side-chains
should be used as a reference [113,118].

2.4 Handling Both Backbone and Side-Chain Flexibility in
Recent Capri Challenges

In recent CAPRI (Critical Assessment of PRediction of Interactions) challenges [133], some of the
participating groups attempted to handle both backbone and sidechain flexibility. Many groups
treated conformational deformations by generating ensembles of conformations, which were
later used for cross-docking. Additionally, some methods, specified below, handle protein
flexibility during the docking process or in a refinement stage.

The group of Bates [134] used MD for generating ensembles of different conformations for the
receptor and the ligand. Then, rigid body cross-docking was performed by the FTdock method
[72]. The best results were minimized by CHARMM [135], and clustered. Finally, a refinement by
MD was performed. It has been shown that the cross-docking produced more near native
results compared to unbound docking only in cases where the proteins undergo large
conformational changes upon binding [134]. Similar conclusions were obtained by Smith et al
[66].

The ATTRACT docking program [53,136] uses a reduced protein model, which represents each
amino acid by up to three pseudo atoms. For each starting orientation, energy minimization is
performed on six rigid-body degrees of freedom and on additional five degrees of freedom
derived from the five lowest frequency normal modes. Finally, the side-chain conformations at
the interface of each docking solution are adjusted using the Swiss-PdbViewer [137], and the
Sander program from the Amber8 package [138] is used for a final minimization.
The RosettaDock method [95,96] performs an initial low-resolution global docking, which includes a Monte Carlo (MC) search with random backbone and rigid-body perturbations. The low energy docking candidates are further refined by Monte Carlo minimization (MCM). Each MCM cycle consists of: (i) backbone and/or rigid-body perturbation, (ii) rotamer-based side-chain optimization and (iii) quasi-Newton minimization on the degrees of freedom of the backbone and/or side-chains and/or rigidbody orientation.

The HADDOCK protocol [35] consists of rigid-body docking followed by a semi-flexible refinement of the interface in torsion angles’ space (of both backbone and side-chains). As a final stage, a Cartesian dynamics refinement in explicit solvent is performed.

To conclude, the treatment of internal flexibility can be performed in different stages of the docking process and in different combinations. In many cases, backbone flexibility is treated before the side-chain flexibility. For example, an ensemble of backbone conformations is often created before the docking procedure. In addition, some methods, like ATTRACT and RosettaDock, perform backbone minimization prior to further refinement. There are two reasons for this order of handling flexibility: (i) the backbone deformations have greater influence on the protein structure than the side-chain movements; (ii) side-chain conformations often depend on the backbone torsion angles. On the other hand, in the final refinement stage, leading docking groups attempt to parallelize the treatment of all the degrees of freedom, including full internal flexibility and rigid-body orientation. CAPRI challenges still show unsatisfying results for cases with significant conformational changes. Therefore the optimal way to combine side-chain and backbone optimization methods is still to be found, and further work in this direction is required.

2.4 Discussion

Protein flexibility presents a great challenge in predicting the structure of complexes. This flexibility includes both backbone and side-chain conformational changes, which increase the size of the search space considerably. In this chapter we reviewed docking methods that handle various flexibility types which are used in different stages of the docking process. These are summarized in the flowchart in Figure 2.6. Table 2.1-Table 2.3 briefly specify the algorithmic approaches of these methods.

The flexible docking process is divided into three major stages. In the first stage the flexibility of the proteins is analyzed. Hinge points can be detected by Ensemble Analysis, GNM or Rigidity Theory. Flexible loops can be identified by MD or Rigidity Theory. Additionally, general conformational space can be defined by NMA, MD or Essential Dynamics. In the second stage the actual docking is performed. If hinges were identified, the subdomains can be docked separately. Furthermore, an ensemble of conformations can be generated, according to the results of the flexible analysis, and docked using cross-docking or the Mean Field approach. The docking candidates generated in this stage are refined in the third stage. This stage refines the backbone, side chains and rigid-body orientation. These three can be refined separately in an
iterative manner or simultaneously. Backbone refinement can be performed by normal modes minimization. Side-chain optimization can be achieved by methods like iterative elimination, graph theory algorithms, MILP, and the Mean Field approach. The refinement of the orientation can be done by a variety of minimization methods such as Steepest Descent [139], Conjugate Gradient [140], Newton-Raphson, quasi-Newton [141] and Simplex [142]. Simultaneous refinement can be performed by methods like MD, MC, and genetic algorithms. The final refined docking candidates are scored and ranked.

In spite of the variety of methods developed for handling protein flexibility during docking, the challenge is yet far from resolved. This can be observed from the CAPRI results [133,143,144] where in cases with significant conformational changes the predictions were dissatisfying. Modeling backbone flexibility is currently the main challenge in the docking field and is addressed by only a few methods. In contrast, side-chain flexibility is easier to model and encouraging results have been achieved. The rigid-body optimization stage plays an important role in flexible docking refinement, and contributes considerably to docking prediction success [113]. However, we believe that in order to achieve the best flexible refinement results, the refinement of the backbone, side-chains and rigid-body orientation need to be parallelized. Parallel refinement will best model the induced fit process that proteins undergo during their interaction.

Another major obstacle in the flexible docking field is the poor ranking ability of the current scoring functions. Adding degrees of freedom of protein flexibility to the search space increases the number of false-positive solutions. Therefore, a reliable energy function is critical for the correct model discrimination. The near-native solutions can be identified not only by their energies, but also by the existence of energy binding funnels [15,16]. Since the ranking ability of the current methods is dissatisfying, further work in this field is required.

Finally, we would like to emphasize that although modeling internal flexibility is essential for general docking predictions, rigid docking is also extremely important. In many known cases the structural changes that occur upon binding are minimal, and rigid-docking is sufficient [145,146]. The benefits of the rigid procedure are its simplicity and relatively low computational time. In addition, a reliable rigid docking algorithm is essential for generating good docking candidates for further flexible refinement.
Figure 2.6. A summary of methods for handling flexibility during docking, which are reviewed in this chapter. The methods handle various flexibility types and are used in different stages of the docking process. Docking applications which implement the algorithmic methods are in brackets. The figure was taken from Andrusier et al. 2008.
<table>
<thead>
<tr>
<th>Method</th>
<th>Flexibility type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HingeFind [21]</td>
<td>Hinge bending</td>
<td>Compares given conformational states using sequence alignment and detects hinge locations.</td>
</tr>
<tr>
<td>FlexProt [23,24]</td>
<td>Hinge bending</td>
<td>Compares given conformational states, preforms structural alignment and detects hinge locations.</td>
</tr>
<tr>
<td>HingeProt [50]</td>
<td>Hinge bending</td>
<td>Detects hinge locations using GNM.</td>
</tr>
<tr>
<td>CONCOORD [63]</td>
<td>General flexibility</td>
<td>Generates conformations that fulfill distance constraints.</td>
</tr>
<tr>
<td>Dynamite [65]</td>
<td>General flexibility</td>
<td>Generates conformations using the essential dynamics approach.</td>
</tr>
</tbody>
</table>

Table 2.1. Some Methods for Flexibility Analysis.

<table>
<thead>
<tr>
<th>Method</th>
<th>Flexibility type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MC2 [83]</td>
<td>Flexible loops</td>
<td>Chooses the best loop conformations from an ensemble using the Mean-Field approach.</td>
</tr>
<tr>
<td>ATTRACT [53,85]</td>
<td>Flexible loops</td>
<td>Chooses the best loop conformations from an ensemble using the Mean-Field approach.</td>
</tr>
<tr>
<td></td>
<td>General flexibility</td>
<td>Energy minimization on degrees of freedom derived from the lowest frequency normal modes.</td>
</tr>
<tr>
<td>FlexDock [88]</td>
<td>Hinge bending</td>
<td>Allows hinge bending in the docking. The rigid sub-domains are docked separately and consistent results are assembled.</td>
</tr>
<tr>
<td>HADDOCK [34,35]</td>
<td>General flexibility</td>
<td>Handles backbone flexibility in the refinement stage, by simulated annealing MD.</td>
</tr>
<tr>
<td>RosettaDock [15,95,118]</td>
<td>General flexibility</td>
<td>Handles backbone flexibility in the refinement stage, by Monte Carlo minimization.</td>
</tr>
</tbody>
</table>

Table 2.2. Some Methods for Docking with Backbone Flexibility.
<table>
<thead>
<tr>
<th>Method</th>
<th>Side-chain flexibility</th>
<th>Rigid-body optimization</th>
<th>Scoring function terms</th>
</tr>
</thead>
<tbody>
<tr>
<td>RosettaDock</td>
<td>MC on rotamers and minimization of rotamer</td>
<td>MC with DFP quasi-Newton minimization</td>
<td>Linear repulsive van der Waals (vdW), attractive vdW, surface area, rotamer probability, hydrogen bonds, residue pair potentials, and electrostatics.</td>
</tr>
<tr>
<td>[15,95,118]</td>
<td>torsion angles</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ICM-DISCO</td>
<td>Biased probability MC on internal</td>
<td>Biased probability MC on internal</td>
<td>Truncated vdW, electrostatics, solvation, hydrogen bonds, and hydrophobicity.</td>
</tr>
<tr>
<td>[128]</td>
<td>coordinates</td>
<td>coordinates</td>
<td></td>
</tr>
<tr>
<td>[121]</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SmoothDock</td>
<td>Pre-docking MD and adopted basis Newton-</td>
<td>Simplex [142] and ABNR minimization</td>
<td>VdW, electrostatics, and atomic contact energy (ACE).</td>
</tr>
<tr>
<td>[16,129]</td>
<td>Raphson (ABNR) minimization in the</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>refinement</td>
<td></td>
<td></td>
</tr>
<tr>
<td>HADDOCK</td>
<td>Simulated annealing MD</td>
<td>Steepest-descent minimization [139]</td>
<td>VdW, electrostatics, binding site restriction, and buried surface area.</td>
</tr>
<tr>
<td>[34,35]</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RDOCK</td>
<td>ABNR minimization</td>
<td>ABNR minimization</td>
<td>Electrostatics and ACE.</td>
</tr>
<tr>
<td>[149]</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FireDock</td>
<td>MILP</td>
<td>MC with BFGS quasi-Newton minimization</td>
<td>Linear repulsive vdW, attractive vdW, ACE, electrostatics, π-stacking and aliphatic interactions, hydrogen and disulfide bonds, and insideness measure.</td>
</tr>
<tr>
<td>[113]</td>
<td></td>
<td>[150,151]</td>
<td></td>
</tr>
</tbody>
</table>

Table 2.3. Some Docking and Refinement Methods with Side-Chain and Rigid-Body Optimization.
Chapter 3:
FiberDock - Flexible induced-fit backbone refinement in molecular docking

Upon binding, proteins undergo conformational changes. These changes often prevent rigid-body docking methods from predicting the 3D structure of a complex from the unbound conformations of its proteins. Handling protein backbone flexibility is a major challenge for docking methodologies, as backbone flexibility adds a huge number of degrees of freedom to the search space, and therefore considerably increases the running time of docking algorithms. Normal mode analysis permits description of protein flexibility as a linear combination of discrete movements (modes). Low-frequency modes usually describe the large-scale conformational changes of the protein. Therefore, many docking methods [53,92] model backbone flexibility by using only few modes, which have the lowest frequencies. However, studies show [45,54] that due to molecular interactions, many proteins also undergo local and small-scale conformational changes, which are described by high-frequency normal modes. Here we present a new method, FiberDock, for docking refinement. The method allows both backbone and side-chain flexibility. It minimizes the structural conformations of the interacting proteins and optimizes their rigid-body orientation. The side-chain flexibility is modeled by a rotamer library, and the backbone flexibility is modeled by an a priori unlimited number of normal modes. The method iteratively minimizes the structure of the flexible protein along the most relevant modes. The relevance of a mode is calculated according to the correlation between the chemical forces, applied on each atom, and the translation vector of each atom, according to the normal mode. The results show that the method successfully models backbone movements that occur during molecular interactions and considerably improves the accuracy and the ranking of rigid-docking models of protein–protein complexes. In addition, we compared FiberDock to our previously developed refinement method FireDock [113] and to the RosettaDock method [15]. Both model only side-chain movements and keep the backbone rigid. This comparison showed that the modeling of backbone flexibility in the refinement process is often critical for creating near-native models with low energy values. A web server for the FiberDock method is available at: http://bioinfo3d.cs.tau.ac.il/FiberDock.

3.1 Methods

Docking refinement aims to refine docking solution candidates and re-rank them to identify near-native models. The refinement has to take into account both backbone and side-chain flexibility. The new method, FiberDock (flexible induced-fit backbone refinement in molecular docking), presented here combines a novel NMA-based backbone flexibility treatment with our previously developed flexible side-chain refinement technique, FireDock [113]. The algorithm, described in the flowchart in Figure 3.1 contains the following steps:
1. **Preprocessing**: Normal mode analysis of the flexible proteins using the anisotropic network model (ANM) [42].

2. For each docking solution candidate do:
   a. **Side-chain optimization**: Side-chain flexibility of interface residues is modeled by a rotamer library. The optimal combination of rotamers is found by an integer linear programming (ILP) technique [110]. At the end of this stage, a rigid body minimization is performed by the BFGS quasi-Newton algorithm [150,151].
   b. **NMA-based backbone refinement**: The backbone refinement performs up to N iterations which consist of the following steps:
      i. The van der Waals (vdW) forces that the proteins apply on each other are calculated.
      ii. The 10 normal modes with the best correlation to these forces are identified, and the backbone conformation is minimized along these modes.
      iii. Ten Monte-Karlo (MC) iterations of rigid-body minimization are performed (as described in item 2c).
      iv. A score is calculated for the current result and the result is saved if it is superior to the results from previous iterations.
         The iterative process of the backbone refinement step stops if the repulsive van der Waals (repVdW) energy value of the current result is below a threshold (no significant steric clashes) or if there was no improvement in the result in the last five iterations.
   c. **Rigid body MC minimization**: The rigid-body orientation of the ligand is optimized by a MC technique (50 iterations), and a BFGS quasi-Newton minimization is performed in each MC cycle [150,151].

3. **Ranking according to an approximation of the energy function**: This stage attempts to identify near-native solutions among the entire set of refined complexes.

In the evaluation experiments, detailed in the Results section, FiberDock was used to model backbone flexibility in one of the proteins (the receptor) and side-chain flexibility in both of them. Up to 20 iterations of backbone refinement were performed (N=20), and the normal modes with the best correlation to the repVdW forces were chosen out of the first 200 modes, with the lowest frequencies. The running time of the refinement algorithm on a single docking
solution varies between 1 and 50 seconds (average of 14 seconds) depending on the size of the receptor.

The implementation of the side-chain optimization (item 2a) and the rigid-body MC minimizations (item 2c) steps were adopted from the FireDock method [113]. The number of iterations performed in the rigid-body MC minimization step was chosen according to convergence rate of the minimization. The rest of the steps are detailed later.

3.1.1 Normal mode analysis

NMA enables us to describe protein flexibility as a linear combination of discrete movements [41,42,43]. Given a single conformation of a protein, the analysis provides a set of vectors (normal modes) that describe typical motions of the analyzed protein. Each normal mode vector contains $3N$ entries, where $N$ is the number of atoms or $C\alpha$ atoms in the protein, depending on the resolution of the analysis. The entire set of normal modes spans the conformational space of the protein, that is, any conformation can be expressed as a linear combination of normal modes. The coefficient of a normal mode represents its amplitude. In addition, the analysis provides the vibration frequency of each mode. The low-frequency modes usually describe the large scale motions of the protein.

In general, normal modes are calculated as follows. First, the Hessian matrix ($K$) of the second derivative of the potential energy ($U$) of each atom in each axis is calculated as follows:

$$K_{ij} = \frac{\partial^2 U}{\partial r_i \partial r_j}$$

The size of the matrix is $3N \times 3N$, where $N$ is the number of atoms. $r_i$ denotes the position of atom $i$ in the minimal energy conformation. Then, the matrix is converted to mass-weighted coordinates according to the formula:

$$\tilde{K} = \sqrt{M}^{-1} \cdot K \cdot \sqrt{M}^{-1}$$

where $M$ is a diagonal $3N \times 3N$ matrix containing the atomic masses. The normal modes are the eigenvectors of this matrix ($\tilde{K}$). The vibration frequencies are the square roots of the corresponding eigenvalues.

The Anisotropic Network Model (ANM) is a simplified spring model of a protein, which is commonly used for NMA [42,44]. This model is based on a pairwise harmonic potential function, which is calculated for atom pairs whose distance from each other is below a threshold. The model treats the analyzed structure as the equilibrium conformation, as opposed to the original all-atom-based NMA which requires prior energy minimization. The harmonic potential function is detailed in the following formula:

$$U(R_1, ..., R_N) = \sum_{i,j} U_{ij}(R_i - R_j)$$
where $R_i$ is the position of atom $i$, and $R_i^0$ is the position of atom $i$ in the equilibrium conformation. $k_{ij}$ denotes the force coefficient of the spring, which connects atoms $i$ and $j$ in the model.

In this work, we used the NMA software developed by Lindahl et al. [92]. The software uses the ANM with force coefficients that decay exponentially with distance, as detailed below:

$$k_{ij} = \exp\left(-\frac{|R_i^0 - R_j^0|^2}{r_0^2}\right)$$

The analysis was performed on the Cα atoms with screening length ($r_0$) of 3Å and a distance cutoff of 10Å.

### 3.1.2 Correlation measurement

In each iteration of the backbone refinement procedure, the normal modes with the best correlation to the repVdW forces are applied on the flexible protein (or proteins). The application of the repVdW forces only produced better results (i.e., more accurate backbone movements and more accurate docking results) when compared with the application of the full van der Waals forces. This is probably due to the fact that the correlation to the repVdW forces helps us choose normal modes that describe backbone movements that resolve existing steric clashes, which often prevent docking methods from succeeding. Resolving steric clashes in near-native rigid-docking solutions drastically improves the energy score of the model, enabling it to be highly ranked and therefore identified among a group of hundreds or thousands solution candidates. Normal modes that correlate well with attractive vDW forces often describe unrealistic closing motions of the receptor around the ligand. We believe that attractive vDW forces can still improve the results when used with certain regularization factors (yet to be found) and plan to continue investigating the optimal application of these forces along with other attractive forces in the future.

The vDW forces are calculated according to the derivative of the modified Lennard-Jones 6–12 potential with linear short-range repulsive score [15]. Specifically, the value of the vDW force between atom $a_i$ and $a_j$ is calculated as follows:

$$F_{\text{vdW}}(a_i, a_j) = \begin{cases} 
-12\varepsilon_{ij} \frac{\sigma_{ij}^6}{r_{ij}^6} \left(\frac{\sigma_{ij}^6}{r_{ij}^6} - 1\right) & \text{for } r_{ij} > 0.6\sigma_{ij} \\
-\varepsilon_{ij} B & \text{otherwise}
\end{cases}$$

where $B = \frac{12}{0.6^7} \left(\frac{1}{\sigma_{ij}} - \frac{1}{0.6^6}\right)$.
The parameter $r_{ij}$ is the sum of the radii of the two atoms. The parameter $\varepsilon_{ij}$ is the energy well depth, and its value was taken from CHARMM22 force field parameters [152].

A vdW force that a ligand atom applies on a receptor atom is considered to be repulsive if it pushes the receptor atom in a direction opposite to the ligand atom. The repVdW forces that are applied on the atoms of a certain amino acid are summed, and the resulting force vector is assigned to the C\textalpha atom of that amino acid.

The correlation between the forces ($F$) that are applied on the C\textalpha atoms and a certain normal mode ($V_i$) is calculated as follows:

$$
corr(F, V_i) = \frac{1}{m \cdot (V_i^{freq})^2} \sum_{j=0}^{m} \overline{v}_{ij} \cdot \overline{f}_j
$$

where $m$ is the number of C\textalpha atoms in the receptor, and $\tilde{m}$ is the number of C\textalpha atoms on which a vdW force is applied ($\tilde{m} = |\left\{ f_j : \overline{f}_j > \varepsilon, j = 1 \ldots m \right\}|$). $F = (\overline{f}_1, \ldots, \overline{f}_m)$ are the repVdW forces applied on the C\textalpha atoms, and $V_i = (\overline{v}_{i1}, \ldots, \overline{v}_{im})$ are the displacement vectors of each C\textalpha atom according to the $i^{th}$ normal mode. $V_i^{freq}$ denotes the frequency value of the $i^{th}$ normal mode.

The absolute value of the dot product $\overline{v}_{ij} \cdot \overline{f}_i$ is higher when the angle between the force vector and the normal mode vector (or its inverse vector) is smaller. Therefore, high correlation indicates that the forces and the normal mode vectors are in similar directions. Additionally, the absolute value of the dot product is higher when the force is stronger (the vector’s $l_2$ norm is bigger). Hence, the correlation measurement gives higher weight to an agreement with the direction of strong vdW forces. Moreover, the division by $(V_i^{freq})^2$ increases the correlation value of the lowest frequency normal modes and therefore gives them a higher priority.

### 3.1.3 Minimization according to normal modes

In each iteration of the backbone refinement procedure, the 10 most relevant normal modes, chosen by the correlation measurement described earlier, and the six rigid-body degrees of freedom, represented as six modes that describe translation and rotation movement along the three axes, are used for minimizing the structure of the complex (overall, 16 degrees of freedom). The energy function of the structure minimization procedure is composed of the attractive and repulsive van der Waals energy and a penalty deformation energy term that prevents the minimization from over-distorting the structure. The energy function formula is:

$$
E = KE_{attrVDW} + E_{repVDW} + \lambda \sum_{i=1}^{M} (V_i^{freq})^2 |V_i^{amp}|
$$
where $K$ is the weight of the attractive van der Waals term in the energy function. The choice of $K = 5$ yields the best performance results. $M$ denotes the number of normal modes, the parameter $\lambda$ is a scaling factor which was set to 0.05. $V_i^{freq}$ denotes the frequency of the $i^{th}$ normal mode (the frequency of the rigid-body modes is zero), and $V_i^{amp}$ denotes its amplitude. The vdw energy values are calculated according to the modified Lennard-Jones 6–12 potential with linear short-range repulsive score [15]. If the vdw energy between two atoms is a positive number, then it is added to the repulsive van der Waals term $E_{repvdW}$, otherwise to the attractive van der Waals term $E_{attrvdW}$.

Using BFGS quasi-Newton algorithm [150,151] we find the optimal amplitudes of the 10 minimized normal modes and the six rigid body degrees of freedom, which result in the nearest local energy minimum. The algorithm uses the gradient of the energy function above. The gradient in the direction of normal mode $V_i$ is:

$$\frac{dE}{dV_i^{amp}} = \begin{cases} \sum_{j=0}^{m} \overline{v}_i \cdot \overline{f}_j + \lambda (V_i^{freq})^2, & \text{if } V_i^{amp} > 0 \\ \sum_{j=0}^{m} \overline{v}_i \cdot \overline{f}_j - \lambda (V_i^{freq})^2, & \text{if } V_i^{amp} < 0 \end{cases}$$

where $\overline{V}_i^{amp}$ is the normal mode $V_i$ multiplied by its amplitude, $m$ is the number of Cα atoms in the receptor, $\overline{f}_1, \ldots, \overline{f}_m$ are the vdw forces applied on the Cα atoms (the attractive forces are multiplied by $K$), and $\overline{v}_{i1}, \ldots, \overline{v}_{im}$ are the displacement vectors of each Cα according to the $i^{th}$ normal mode. At the end of each structure minimization step, we apply the normal modes on the flexible protein (or proteins) with the optimal amplitudes found, as described later.

### 3.1.4 Applying a normal mode on a protein

A normal mode $V_i = (\overline{v}_{i1}, \ldots, \overline{v}_{im})$ is composed of displacement vectors for each Cα atom in a protein. When applying normal mode movements on a protein in a naïve manner, that is, by adding the displacement vectors, multiplied by an amplitude value, to the points of the Cα atom, the protein structure often distorts. We would like to change the conformation of the protein according to certain normal modes while preserving the bond lengths and angles, that is, by allowing a change only in the backbone dihedral angles, $\phi$ and $\psi$.

To overcome this problem, we use a modification of the CCD algorithm [153], a robotics algorithm which was adapted for protein loop closure. First, we add the displacement vectors of the normal modes to the centers of the Cα atoms and get the desired positions of the atoms, denoted by $(a_1, \ldots, a_m)$. Then we start from the Cα atom that moves the least ($C\alpha_j$, where $j = \text{argmin}_i |\overline{v}_{ij}|$) and change the values of the backbone dihedral angles $\phi$ and $\psi$ in a sequential order in both directions of the backbone chain. For each dihedral angle $(\theta)$ of $C\alpha_k$, we choose the value that minimizes the sum of the squared distances between the next three moving Cα atoms $(c_{k+1}, c_{k+2}, c_{k+3})$ and their desired positions $(a_{k+1}, a_{k+2}, a_{k+3})$ [see the formula of $S(\theta)$ below]. The value of each angle is calculated by setting the first-order derivative
of the sum of the square distances to zero \( \frac{dS}{d\theta} = 0 \), as described by Canutescu and Dunbrack [153].

\[
S(\theta) = [d(c_{k \pm 1}, a_{k \pm 1})]^2 + [d(c_{k \pm 2}, a_{k \pm 2})]^2 + [d(c_{k \pm 3}, a_{k \pm 3})]^2
\]

### 3.1.5 The scoring function of the backbone refinement stage

At the end of each iteration in the backbone refinement procedure, a score is calculated for the current solution, and the solution with the best score is returned from the procedure. This scoring function is identical to the energy function of the normal modes minimization specified above.

### 3.1.6 Ranking according to an approximation of the energy function

This stage attempts to identify near-native solutions among the entire set of refined complexes. The calculated energy score is an approximation of the binding free energy function. It includes an interface energy score, adopted from the FireDock method [113], and an energy term that approximates the internal deformation energy of the flexible proteins. The interface energy score includes a variety of energy terms, such as desolvation energy (ACE), van der Waals interactions, partial electrostatics, hydrogen and disulfide bonds, \( p \)-stacking, aliphatic interactions, and more. These terms are described in detail in the FireDock paper [113]. The added deformation energy term, specified below, approximates the energy required for deforming the unbound backbone structure according to the calculated linear combination of the chosen relevant normal modes.

\[
E_{\text{deform}} = \sum_{i=1}^{M} (V_{i}^{\text{freq}})^2 |V_{i}^{\text{amp}}|
\]

where \( V_{i}^{\text{freq}} \) denotes the frequency of the \( i^{\text{th}} \) normal mode and \( V_{i}^{\text{amp}} \) denotes its amplitude. The deformation energy term, \( E_{\text{deform}} \), is added to the interface energy function with a weight of \( \lambda = 0.05 \).

### 3.1.7 RMSD calculations

The root mean square deviation (RMSD) is a common measure of the difference between structures of two proteins (or complexes). The RMSD is calculated according to the following equation:

\[
\text{RMSD} = \sqrt{\frac{1}{n} \sum_{i=1}^{n} ||v_i - u_i||^2}
\]
where \( n \) is the number of atoms in the compared molecules, \( v_i \) is the position of the \( i^{th} \) atom of the first molecule, and \( u_i \) is the position of the corresponding atom in the second molecule. In this work, we evaluated the results by three types of RMSD measurements:

**LRMSD**: The RMSD between the predicted location of the ligand and its location in the native complex. The calculation was performed on Cα atoms of the ligand after superimposing the receptor molecules in the native complex and in the predicted complex.

**IRMSD**: The RMSD between the interface Cα atoms in the predicted complex structure and in the native complex structure after superimposing the two interfaces. The interface includes all the residues that contain an atom within 1Å of the other interacting protein in the structure of the native complex, as defined in the evaluation protocol of the CAPRI experiment [154].

**Rec-IRMSD**: The RMSD between the interface Cα atoms in a certain structure of the receptor and in the structure of its bound conformation (as in the native complex), after superimposing the two interfaces.

### 3.1.8 Test cases

We used 20 test systems in which the conformation of the receptor’s backbone changes upon interaction with the ligand. The test cases are detailed in Table 3.1. The interface RMSD between the bound and the unbound conformations of the receptor (Rec-IRMSD) in this data set varies in the range of 0.59–6.08Å. We classified the motions of the receptors into three types: (1) opening motion, where the conformation of the unbound receptor partially blocks the binding site of the ligand (nine cases); (2) closing motion, where the receptor closes around the ligand and increases the contact area (three cases); and (3) other motions, where some of the interface suits the opening criterion and some suits the closing criterion (eight cases). In most of the cases, an unbound structure of the ligand was available. In these cases, unbound–unbound docking experiments were performed.

### 3.2 Results

To evaluate the contribution of the backbone flexibility modeling within the docking refinement process, we compared the performance of the new FiberDock method to the performance of our previously developed flexible side chain refinement technique, FireDock [113]. The only difference between the two methods is the addition of the novel NMA-based backbone refinement procedure in FiberDock (step 2b in the algorithm, described in “Methods” section).

We performed three main experiments. In the first experiment, we tested the performance of the method on refining a complex structure, in which the ligand, in its unbound conformation, is placed in its native binding orientation and the receptor is in its unbound conformation. In the second experiment, we refined, for each test case, 500 randomly generated near-native docking solutions. Here, we compared FiberDock with both FireDock and RosettaDock [15], and we investigated the influence of the backbone refinement procedure on the shape of the energy
funnels created around the native binding orientation of the ligand. Finally, in the last experiment, we refined the best 500 results of the PatchDock rigid-body docking method [79,155] and rescored the results. Ranking was identified as a major bottleneck in the CAPRI challenge [64,133,156]. Therefore, in this last experiment, we aim to test to what extent FiberDock improves the ranking of the docking procedure.

<table>
<thead>
<tr>
<th>No.</th>
<th>Complex ID</th>
<th>Unbound receptor</th>
<th>Unbound ligand</th>
<th>Complex description</th>
<th>Rec-IRMSD</th>
<th>Motion type</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1AO0</td>
<td>1CHN</td>
<td>1FWP</td>
<td>CheY-binding domain of CheA in complex with CheY</td>
<td>2.12</td>
<td>Closing</td>
</tr>
<tr>
<td>2</td>
<td>1ACB</td>
<td>2CGA</td>
<td>1EGL</td>
<td>Bovine alpha-chymotrypsin-Eglin C complex</td>
<td>2.68</td>
<td>Opening</td>
</tr>
<tr>
<td>3</td>
<td>1AY7</td>
<td>1RGH</td>
<td>1A19</td>
<td>Ribonuclease Sa complex with Barstar</td>
<td>0.59</td>
<td>Closing</td>
</tr>
<tr>
<td>4</td>
<td>1BTH</td>
<td>2HNT</td>
<td>6PTI</td>
<td>Thrombin complexed with bovine pancreatic trypsin inhibitor</td>
<td>1.31</td>
<td>Opening</td>
</tr>
<tr>
<td>5</td>
<td>1CGI</td>
<td>2CGA</td>
<td>1HPT</td>
<td>Bovine chymotrypsinogen A and pancreatic secretory trypsin inhibitor</td>
<td>2.26</td>
<td>Opening</td>
</tr>
<tr>
<td>6</td>
<td>1DFJ</td>
<td>2BNH</td>
<td>7RSA</td>
<td>Ribonuclease inhibitor complexed with ribonuclease A</td>
<td>1.18</td>
<td>Opening</td>
</tr>
<tr>
<td>7</td>
<td>1E6E</td>
<td>1E1N</td>
<td>1CJE</td>
<td>Adrenodoxin reductase-adrenodoxin complex</td>
<td>0.62</td>
<td>Other</td>
</tr>
<tr>
<td>8</td>
<td>1FIN</td>
<td>1HCL</td>
<td>1VIN</td>
<td>Cyclina-CDK2 complex</td>
<td>6.08</td>
<td>Other</td>
</tr>
<tr>
<td>9</td>
<td>1GGI</td>
<td>1GGC</td>
<td>-</td>
<td>HIV-1 neutralizing antibody in complex with its V3 loop peptide antigen</td>
<td>1.67</td>
<td>Opening</td>
</tr>
<tr>
<td>10</td>
<td>1GOT</td>
<td>1TAG</td>
<td>1TBG</td>
<td>Heterotrimeric G protein</td>
<td>3.72</td>
<td>Opening</td>
</tr>
<tr>
<td>11</td>
<td>1IBR</td>
<td>1FS9</td>
<td>1FS9</td>
<td>Complex of Ran with Importin beta</td>
<td>2.62</td>
<td>Opening</td>
</tr>
<tr>
<td>12</td>
<td>1OAZ</td>
<td>1OAQ</td>
<td>-</td>
<td>Immunoglobulin E complexed with a Thioredoxin 1</td>
<td>1.07</td>
<td>Other</td>
</tr>
<tr>
<td>13</td>
<td>1PXV</td>
<td>1X9Y</td>
<td>1NYC</td>
<td>Staphostatin–Staphopain complex</td>
<td>3.48</td>
<td>Other</td>
</tr>
<tr>
<td>14</td>
<td>1T6G</td>
<td>1UKR</td>
<td>1T6E</td>
<td>Complex of endo-1,4 beta xylanase I and xylanase inhibitor</td>
<td>0.87</td>
<td>Opening</td>
</tr>
<tr>
<td>15</td>
<td>1TGS</td>
<td>2PTN</td>
<td>1HPT</td>
<td>Complex of trypsinogen and pancreatic secretory trypsin inhibitor</td>
<td>1.54</td>
<td>Closing</td>
</tr>
<tr>
<td>16</td>
<td>1WQ1</td>
<td>6Q21</td>
<td>6Q21</td>
<td>Ras-RasGAP complex</td>
<td>0.93</td>
<td>Other</td>
</tr>
<tr>
<td>17</td>
<td>1ZHI</td>
<td>1M4Z</td>
<td>1ZIA</td>
<td>Complex of Orc1 and Sir1 interacting domains</td>
<td>0.74</td>
<td>Closing</td>
</tr>
<tr>
<td>18</td>
<td>2BUO</td>
<td>1A43</td>
<td>-</td>
<td>HIV-1 capsid C-terminal domain with an inhibitor of particle assembly</td>
<td>4.15</td>
<td>Opening</td>
</tr>
<tr>
<td>19</td>
<td>2KAI</td>
<td>2PKA</td>
<td>6PTI</td>
<td>Complex of porcine kallikrein A and the bovine pancreatic trypsin inhibitor</td>
<td>0.72</td>
<td>Other</td>
</tr>
<tr>
<td>20</td>
<td>3HHR</td>
<td>1HGU</td>
<td>-</td>
<td>Complex of a human growth hormone and extracellular domain of its receptor</td>
<td>2.62</td>
<td>Opening</td>
</tr>
</tbody>
</table>

Table 3.1. Flexible Protein–Protein Docking Data Set

3.2.1 Docking refinement starting from known binding orientation and unbound conformation of the proteins

In this experiment, we check the performance of the refinement method on the native complex structures after replacing the bound conformation of each protein with the superimposed unbound conformation. These complexes contain steric clashes due to the wrong conformation of the proteins. Therefore, their initial energy score is high. The refinement of the complex
attempts to find a structurally close complex structure with minimal energy score. The results of the refinement are detailed in Table 3.2.

The results show that in many cases, FiberDock produced a near-native model with a much lower energy value when compared with FireDock. In five of the cases, the energy difference was very significant. These include case numbers 8, 9, 10, 11, and 20. In all of these cases, the receptor opens its binding site upon interaction with the ligand. Modeling these opening movements by FiberDock resolved the steric clashes between the proteins in their unbound conformations and therefore significantly improved the energy score of these complexes.

<table>
<thead>
<tr>
<th>Complex ID</th>
<th>FireDock (rigid backbone)</th>
<th>FiberDock (flexible backbone)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>IRMSD</td>
<td>recIRMSD</td>
</tr>
<tr>
<td>1. 1A0O</td>
<td>2.44</td>
<td>2.12</td>
</tr>
<tr>
<td>2. 1ACB</td>
<td>2.58</td>
<td>2.58</td>
</tr>
<tr>
<td>3. 1AY7</td>
<td>1.30</td>
<td>0.59</td>
</tr>
<tr>
<td>4. 1BTH</td>
<td>1.16</td>
<td>1.31</td>
</tr>
<tr>
<td>5. 1CGI</td>
<td>2.08</td>
<td>2.26</td>
</tr>
<tr>
<td>6. 1DFJ</td>
<td>1.41</td>
<td>1.18</td>
</tr>
<tr>
<td>7. 1E6E</td>
<td>1.21</td>
<td>0.62</td>
</tr>
<tr>
<td>8. 1FIN</td>
<td>5.17</td>
<td>6.08</td>
</tr>
<tr>
<td>9. 1GGI*</td>
<td>2.68</td>
<td>1.67</td>
</tr>
<tr>
<td>10. 1GOT</td>
<td>3.02</td>
<td>3.72</td>
</tr>
<tr>
<td>11. 1BIR</td>
<td>2.78</td>
<td>2.62</td>
</tr>
<tr>
<td>12. 1OAZ*</td>
<td>1.00</td>
<td>1.07</td>
</tr>
<tr>
<td>13. 1PXF</td>
<td>3.54</td>
<td>3.48</td>
</tr>
<tr>
<td>14. 1T66</td>
<td>0.99</td>
<td>0.87</td>
</tr>
<tr>
<td>15. 1T65</td>
<td>1.57</td>
<td>1.54</td>
</tr>
<tr>
<td>16. 1WQ1</td>
<td>1.50</td>
<td>0.93</td>
</tr>
<tr>
<td>17. 1ZHI</td>
<td>1.24</td>
<td>0.74</td>
</tr>
<tr>
<td>18. 2BUO*</td>
<td>3.92</td>
<td>4.15</td>
</tr>
<tr>
<td>19. 2KAI</td>
<td>0.74</td>
<td>0.72</td>
</tr>
<tr>
<td>20. 3HHR*</td>
<td>2.46</td>
<td>2.62</td>
</tr>
</tbody>
</table>

Table 3.2. Refinement of the unbound receptor and unbound ligand in their native binding orientation.

*For these cases, a structure of the ligand in its unbound conformation was not available. Therefore, the bound conformation of the ligand was used.

Inspection of the results in Table 3.2 also reveals that in 7 of the 20 cases, the refinement by FiberDock resulted in a model of a complex in which the conformation of the interface deviates less from the bound structure when compared with the model created by FireDock (lower IRMSD). In all of these cases, FiberDock also created a conformation of the receptor interface, which was closer to the bound conformation than with the unbound conformation (lower Rec-IRMSD). In three cases, the receptor’s interface RMSD (Rec-IRMSD) and the total interface RMSD (IRMSD) got worse, and in the rest of the cases (10), the Rec-IRMSD and IRMSD remained unchanged. The best improvements in the Rec-IRMSD were in case numbers 9 (1GGI) and 14 (1T6G), where the improvement in the Rec-IRMSD was around 25%, when compared with the unbound conformation.

Case number 9 is an antibody–antigen complex, with a flexible loop in the binding site of the antibody. The loop movement, which is essential for the interaction, is modeled correctly by
FiberDock (Figure 3.2). The refined structure of the antibody was created by a linear combination of low- and high-frequency modes, as described by the formula: 

\[ R' = R - 2.80V_1 - 2.13V_3 + 1.45V_4 + 1.34V_6 - 0.95V_8 - 3.90V_9 - 5.00V_{10} - 6.09V_{11} + 2.38V_{15} - 7.92V_{16} - 6.82V_{17} + 4.24V_{21} + 0.82V_{27} + 4.38V_{32} + 0.37V_{40} + 0.60V_{130}, \]

where \( R \) is the unbound structure, \( V_i \) is the \( i \)th normal mode, and \( R' \) is the modeled structure of the receptor. The normal mode that has the highest amplitude in this linear combination is mode number 16 (amplitude of 7.92). Figure 3.3 shows that this normal mode describes local deformation of the flexible region that indeed moves upon interaction with the antigen. On the other hand, the figure shows that the first normal mode, which has a lower amplitude in the linear combination (-2.8), describes a collective movement that is not specifically relevant to the flexibility induced by the interaction with the antigen. The peak that exists around residue 29 (in both modes) is due to a missing segment in the unbound structure that is interpreted as a flanking end by the NMA.

**Figure 3.2.** The FiberDock refinement result of test case number 9 (1GGI), HIV-1 neutralizing antibody in complex with its V3 loop peptide antigen, starting from the known binding orientation of the ligand (the antigen) and unbound conformation of the receptor (the antibody). The unbound structure of the receptor (the starting conformation of the refinement) is colored in blue and the bound structure of the receptor is in green. The bound ligand in the native orientation is presented in gray. The refined structure of the receptor, which was created by FiberDock, is in red. The refinement predicted accurately the loop movement in the binding site of the antibody that occurs during the interaction with the antigen (marked by an arrow). The figure was taken from Mashiach et al. 2010.
Although the improvement in the IRMSD was modest in this experiment, FiberDock results achieved much better energy values. According to the last CAPRI Assessment Meeting [144], one of the current major challenges in docking is ranking docking solutions and sorting out false positives. The energy value is a crucial factor in the final ranking. A relatively accurate model (with low IRMSD) which has a high energy value will not be ranked high among a group of docking solution candidates. Therefore, the improvement in the energy of the refined models is very important in the docking scheme.

### 3.2.2 Docking refinement starting from random orientations of the ligand around the native binding orientation

In this experiment, we used FiberDock for local-docking around the native binding orientation of the ligand. We created 500 random transformations of the ligand around the native orientation and refined each of them. To create the random transformations, we sampled the three translation variables (in X,Y,Z axes) from a Gaussian distribution with mean 0 Å and standard deviation 3 Å. The three rotation variables (along the X,Y,Z axes) were sampled from a Gaussian distribution with mean 0° and standard deviation 8°. These selected values of standard
deviations are similar to the values used in the perturbation studies of Gray et al. [15] (standard deviation of 8° for rotation, 3Å for translation along the line of protein centers, and 8Å for translation in the two perpendicular directions). By applying these 500 transformations on the ligand, we created 500 starting docking models for refinement.

In almost all of the test cases, the FiberDock refinement protocol produced many more near-native results with low-energy values than the FireDock method. We defined a good solution as a solution in which the energy score is negative and the IRMSD is lower than 4Å, which is an acceptable solution according to the CAPRI contest [154]. The number of good solutions of FireDock and FiberDock, for each test case, is presented in Figure 3.4.

![Figure 3.4](image-url)  
**Figure 3.4.** Results of refining 500 random transformations of the ligand around the true binding orientation by using the two methods, FireDock with rigid backbone (cyan and blue bars) and FiberDock with flexible backbone (orange and red bars). The cyan and orange bars show the results of the experiment with both the receptors and the ligands in their unbound conformation (UU). The blue and red bars show the results of the experiment with unbound structures of the receptors and bound structures of the ligands (UB). The histogram shows the ratio of good solutions out of the 500 refined models. A good solution is defined as a solution in which the IRMSD is lower than 4Å and the energy value is negative. The histogram is sorted according to the ratio of good solutions of FireDock with unbound structures of the receptors and bound structures of the ligands (FireDock-UB). For cases that are marked by stars (*), a structure of the ligand in its unbound conformation was not available. The figure was taken from Mashiach et al. 2010.

In 17 of the 20 test cases, the number of good solutions was higher in the results of FiberDock when compared with FireDock, in both the unbound–unbound (UU) experiment and the unbound–bound (UB) experiment. In eight cases, this number was higher by more than 40% (in
both experiments). These eight cases include six cases where the binding site of the receptor opens upon binding. In one case (1FIN), none of the methods produced any good solutions. This is the most difficult case in the data set, with recIRMSD of 6.08Å.

Figure 3.5 shows the best IRMSD solution out of the group of good solutions (in the UU experiment) of two test case numbers 11 and 18. In both cases, the refinement correctly modeled backbone movements, which are necessary for solving steric clashes of the receptor and the ligand in near-native orientations. In case number 11 (1IBR), the refinement moved a loop that blocks the binding site in the unbound conformation, allowing the ligand to enter the binding site in a near-native orientation without any clashes and with a low energy value.

The refined structure of the receptor was created by a linear combination of both low- and high-frequency normal modes, described by the formula:

\[ R' = R + 9.84V_1 + 4.31V_2 - 0.83V_3 - 10.63V_4 - 7.43V_5 - 1.84V_6 - 4.70V_7 + 9.21V_8 - 2.78V_9 - 5.10V_{10} - 5.00V_{11} - 3.93V_{12} - 1.16V_{13} + 2.50V_{16} + 1.78V_{17} - 3.49V_{22} - 0.78V_{23} - 2.34V_{41}, \]

where \( R \) is the structure of the unbound receptor, \( V_i \) is the \( i \)th normal mode, and \( R' \) is the modeled structure of the receptor. The five normal modes that have the highest amplitude in this linear combination are mode numbers 1, 4, 5, 8, and 11. The influence of each normal mode on each residue is shown in Figure 3.6. The lower frequency modes (numbers 1, 4 and 5) describe a collective deformation of the protein, whereas the higher frequency modes (numbers 8 and 11) describe local deformation of the loop in the interface (residues 332–344). Figure 3.6 also shows the distance between the positions of each residue in the bound and unbound conformation. These distances have four high peaks (marked A–D in Figure 3.6). The highest peak (C) is between residues 288 and 316. However, these residues are located on the opposite side of the ligand, and their movement is not important for correct modeling of the interaction. The most important movement is of the interface loop (peak D), which is modeled by modes 8 and 11 during the backbone refinement of FiberDock.
Figure 3.6. The influence of the highest amplitude modes used by FiberDock to model the backbone movement of the receptor in test case number 11 (1IBR). The upper graph shows the three low-frequency modes (1,4,5) which describe collective deformations of the protein. The two higher frequency modes (8,11), shown in the bottom graph, describe local deformation of the loop in the interface (residues 332–344). The dashed black line shows the distance between the positions of each residue in the bound and unbound conformation. These distances have four high peaks (marked A–D) in the most flexible positions. On the right, the structure of the bound (blue) and unbound (green) conformations are shown. The flexible regions, which correspond to the peaks, are marked by orange circles. The interface residues are marked by red lines in the x-axis. The figure was taken from Mashiach et al. 2010.
In case number 18 (2BUO), the refinement moved a helix and opened the binding site. Figure 3.5(A) clearly shows that the ligand in its native orientation has a major steric clash with the receptor in the unbound conformation. Therefore, without modeling the backbone movement of the receptor, a low-energy near-native solution cannot be achieved. In this case, the structure of the receptor was also created by a linear combination of both low- and high-frequency normal modes: \[ R' = R - 1.99V_1 - 0.36V_2 - 3.45V_3 + 4.22V_4 + 0.16V_9 + 3.08V_{10} - 0.07V_{14} - 2.18V_{21} + 3.51V_{23} + 2.58V_{26}. \]

3.2.3 Local docking by FiberDock produces more accurate results than RosettaDock

The local docking results of FiberDock were compared with the local docking results of RosettaDock3.0, which keeps the backbone rigid and models only side-chain flexibility. For both methods, we randomly sampled 500 rigid-body perturbations of the ligand, in the bound conformation, from a similar distribution (Gaussian distribution with standard deviation of 3Å for translation and standard deviation of 8° for rotation). For each test case, we compared the accuracy of the lowest IRMSD result in the top 10 solutions (with the lowest energy) of the two methods. The comparison is detailed in Table 3.3.

In 11 of the 20 test cases, FiberDock produced more accurate results than RosettaDock (with \( \Delta \text{IRMSD} = \text{IRMSD}_{\text{FiberDock}} - \text{IRMSD}_{\text{RosettaDock}} < -0.2\text{Å} \)). These cases include most of the test cases where the receptor undergoes an opening motion upon binding. Only in two cases RosettaDock produced better results (\( \Delta \text{IRMSD} > 0.2\text{Å} \)), and in seven cases, the accuracy of the

<table>
<thead>
<tr>
<th>Complex ID</th>
<th>FiberDock</th>
<th>RosettaDock3.0</th>
<th>( \Delta \text{IRMSD}^* )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. 1A0O</td>
<td>1.8</td>
<td>3.11</td>
<td>-1.31</td>
</tr>
<tr>
<td>2. 1ACB</td>
<td>2.21</td>
<td>2.49</td>
<td>-0.28</td>
</tr>
<tr>
<td>3. 1AY7†</td>
<td>0.89</td>
<td>0.72</td>
<td>0.17</td>
</tr>
<tr>
<td>4. 1BTH</td>
<td>1.24</td>
<td>1.24</td>
<td>0.00</td>
</tr>
<tr>
<td>5. 1CGI</td>
<td>2.00</td>
<td>2.04</td>
<td>-0.04</td>
</tr>
<tr>
<td>6. 1DFJ†</td>
<td>1.11</td>
<td>5.80</td>
<td>-4.69</td>
</tr>
<tr>
<td>7. 1E6E</td>
<td>0.63</td>
<td>1.71</td>
<td>-1.08</td>
</tr>
<tr>
<td>8. 1FIN†</td>
<td>5.90</td>
<td>5.93</td>
<td>-0.03</td>
</tr>
<tr>
<td>9. 1GG1†</td>
<td>1.70</td>
<td>2.58</td>
<td>-1.88</td>
</tr>
<tr>
<td>10. 1GOT†</td>
<td>2.59</td>
<td>3.89</td>
<td>-1.3</td>
</tr>
<tr>
<td>11. 1IBRy</td>
<td>1.98</td>
<td>9.01</td>
<td>-7.03</td>
</tr>
<tr>
<td>12. 1OAZ</td>
<td>2.62</td>
<td>1.55</td>
<td>1.07</td>
</tr>
<tr>
<td>13. 1PXV</td>
<td>3.23</td>
<td>3.34</td>
<td>-0.11</td>
</tr>
<tr>
<td>14. 1T6G†</td>
<td>0.77</td>
<td>2.34</td>
<td>-1.57</td>
</tr>
<tr>
<td>15. 1TGS</td>
<td>1.38</td>
<td>1.31</td>
<td>0.07</td>
</tr>
<tr>
<td>16. 1WQ1</td>
<td>1.41</td>
<td>5.06</td>
<td>-3.65</td>
</tr>
<tr>
<td>17. 1ZHI</td>
<td>1.12</td>
<td>0.9</td>
<td>0.22</td>
</tr>
<tr>
<td>18. 2BUO†</td>
<td>3.62</td>
<td>4.24</td>
<td>-0.62</td>
</tr>
<tr>
<td>19. 2KAI</td>
<td>0.75</td>
<td>0.67</td>
<td>0.08</td>
</tr>
<tr>
<td>20. 3HHR†</td>
<td>1.89</td>
<td>4.30</td>
<td>-2.41</td>
</tr>
</tbody>
</table>

Table 3.3. Local Docking Results of FiberDock and RosettaDock.

*\( \Delta \text{IRMSD} \) is the difference between the best IRMSD in the top 10 solutions of FiberDock and RosettaDock.

† Cases where the conformation of the unbound receptor partially blocks the binding site of the ligand and the receptors undergoes an opening motion upon binding.
results were about the same (20.2 Å < ΔIRMSD < 0.2 Å). This comparison shows the ability of FiberDock in modeling opening motions of binding sites and its contribution in producing accurate models of protein–protein complexes.

Wang et al. [95] have recently incorporated explicit backbone flexibility into RosettaDock. During each MC iteration of RosettaDock, a random backbone perturbation is performed together with a rigid-body perturbation. Although the method enables modeling full backbone flexibility for both proteins, in practice, this is extremely computationally demanding because of the high number of degrees of freedom. Therefore, it is feasible only for very small proteins or very subtle backbone perturbations. A more practical use of this method is to predefined the flexible segments of the protein (by a “fold tree” [95]) and perturb backbone conformational changes only in these regions. This, however, requires prior knowledge of the flexible regions. FiberDock, on the other hand, minimizes the backbone conformation along few degrees of freedom, which are carefully picked by NMA. Therefore, the backbone refinement is much faster. In addition, as FiberDock considers both low- and high-frequency normal modes, both global and local conformational changes are modeled, and no prior knowledge of the flexible regions is required. By the time this work was published, backbone flexibility was not yet included in the latest RosettaDock release (version 3.0). Hence, we did not compare the performance of the backbone refinement of RosettaDock to FiberDock.

3.2.4 FiberDock improves the shape of energetic funnels around near-native results

The formation of energy funnels is known to be a relatively reliable indicator for identifying near-native models of protein–protein complexes among a group of solution candidates [157,158]. We used the 500 refined near-native complexes of FiberDock, FireDock, and RosettaDock, generated in the local docking experiments described earlier, to draw energy funnels around the native orientation of the ligands. In many cases, the shapes of the energetic funnels, which were created by FiberDock, were significantly better than the ones created by RosettaDock and FireDock. These funnels usually included many more near-native complex models and reached lower energy values. The energetic funnels of four of these cases (1CGI, 1IBR, 1T6G, 2BUO), using unbound receptors and bound ligands, are shown in Figure 3.7.

The improvement in the shape of the funnels generated by FiberDock when compared with FireDock, shown in Figure 3.7, is clearly due to the backbone refinement procedure, which is the only difference between the two methods. However, the figure also shows that FireDock generates better looking funnels when compared with RosettaDock, although both methods model side-chain flexibility by the same rotamer library and they both optimize the relative rigid-body orientation by a similar technique. There are two possible explanations for these differences in the created energy funnels: (1) The energy function of RosettaDock might be more sensitive to steric clashes than the energy function of FireDock. In these test cases, all of which include backbone conformational changes, most of the near-native (rigid backbone)
results contain a certain amount of steric clashes. Energy functions that are too sensitive to clashes would not show a funnel-shaped energy landscape around the native ligand orientation. (2) The side-chain optimization technique is different in these two methods. FireDock optimizes the rotamers selection by the ILP approach, which guarantees to find the combination of rotamers that globally minimizes the repulsive vdW interface energy. RosettaDock, on the other hand, uses the heuristic MC technique for side-chain repacking. To fully understand the true reason for these differences in the shape of energy funnels, further research should be performed, which is out of the scope of this work.

Figure 3.7. Funnels created by the three refinement methods: RosettaDock, FireDock, and FiberDock, using unbound structure of the receptor and bound structure of the ligand. Each row compares the funnels created for a certain test case (pdb-id is specified on the left). The x-axis denotes the IRMSD of the refined complex, and the y-axis denotes its energy score value. The figure was taken from Mashiach et al. 2010.
3.2.5 Docking refinement starting from rigid-body docking candidates

In this experiment, we test the contribution of the backbone refinement procedure to the refinement and ranking of rigid-body docking solutions. For each test case, we identified the interacting amino acids (residues which contain an atom within 6 Å from the interacting protein). Then, we ran the PatchDock [79,155] method given the information on the location of the binding site.

Some of the proteins in our data set undergo significant conformational changes upon binding. Therefore, a completely blind rigid-docking run might not have had a near-native solution in its first 500 solution candidates. As we test the refinement and re-ranking abilities of our method, we used the binding site information, which is often known from experimental data.

The solutions of PatchDock are ranked by a shape complementarity score. We refined and re-ranked the best 500 solution candidates by FireDock and FiberDock and compared the results of the three methods (PatchDock, FireDock, and FiberDock). We performed this experiment on the unbound conformation of the receptors and the bound conformation of the ligands. The results are presented in Table 3.4.

Table 3.4 shows the rank of the first acceptable solution (IRMSD < 4.0 Å or RMSD < 10.0 Å) and the number of acceptable solutions in the top 20 solutions for each of the methods. The results show a gradual improvement in these criteria. The rank of the first acceptable solution was the best in the results of PatchDock in 6 cases and in the results of FireDock and FiberDock in 8 and 14 cases, respectively. The number of acceptable solutions in the top 20 solutions also increased gradually. In 4 cases, this number was the highest for PatchDock results, and in 7 and 11 cases, it was the highest for the results of FireDock and FiberDock, respectively. These results show that backbone refinement can significantly improve the ranking of near-native docking solutions, as it often solves steric clashes between the interacting proteins that prevent the docking solution from getting a low energy value and good ranking (shown earlier).

In case number 14 (1T6G), the first solution of FiberDock was of medium accuracy according to CAPRI criteria. However, the second solution was highly accurate, with IRMSD of 0.92 Å and RMSD of 3.04 Å (not shown in the table). By examining the structure of the refined model, depicted in Figure 3.8, one can see that FiberDock automatically identified the single loop that slightly moves during the interaction to open the binding site and enable the ligand to enter it in the correct orientation. FiberDock moved this flexible loop in the right direction and kept the other parts of the protein rigid.

The table also provides details relating to the original PatchDock solution of the first acceptable solution of FireDock and FiberDock before the refinement. The refinement by both FireDock and FiberDock significantly improves the ranking and the accuracy of the rigid-docking results. For example, in case number 4 (1BTH), an inaccurate PatchDock solution with RMSD of 14.8 Å and IRMSD of 3.55 Å which was ranked in place 403, was refined by FiberDock to a more accurate
model (RMSD of 7.98Å and IRMSD of 1.97Å ) which was ranked in first place by its energy function. The refinement by FireDock, on the other hand, resulted in a worse model (RMSD of 18.9 Å and IRMSD of 4.64 Å ). In this case, FiberDock hardly changed the backbone conformation of the receptor (RMSD of 0.15Å between the modeled and the unbound conformation of the receptor). However, this case shows that even a slight movement of the backbone, which resolve steric clashes, may enable the rigid-body optimization stage to converge to a near-native position. The results of the refinement by the two methods are shown in Figure 3.9.

Figure 3.8. FiberDock’s predicted model of test case number 14 (complex of Endo-1,4-beta-xylanase I and xylanase inhibitor, pdb-id: 1T6G). This model was ranked in second place after refining and reranking the 500 top solutions of PatchDock. The unbound structure of the receptor (the starting conformations of the refinement) is colored in blue and the bound structure of the receptor is in green. The bound ligand in the native orientation is presented in gray. The refinement solution, which was created by FiberDock, is in red. The refinement accurately predicted the loop movement that occurs in the receptor during the interaction with the ligand. The figure was taken from Mashiach et al. 2010.

Figure 3.9. Refinement of a rigid-docking solution of case number 5 (Thrombin complexed with bovine pancreatic trypsin inhibitor, pdb-id: 1BTH) by FireDock (A) and FiberDock (B). The receptor in its bound conformation is presented in green. The ligand in the native orientation is colored in gray. The original rigid-docking solution (generated by PatchDock), on which the refinement was applied, is colored in blue. The position of the ligand after the refinement by FireDock is presented in orange, and the position of the ligand after the refinement by FiberDock is in red. This case shows a drastic improvement of the docking solution due to the flexible refinement by FiberDock. The figure was taken from Mashiach et al. 2010.
<table>
<thead>
<tr>
<th>Complex ID</th>
<th>PatchDock</th>
<th>FireDock</th>
<th>FiberDock</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>First acceptable(^a)</td>
<td>Acceptables in top 20(^b)</td>
<td>First acceptable(^a)</td>
</tr>
<tr>
<td>1A00</td>
<td>1(^*) (7.66, 3.79)</td>
<td>3(^†)</td>
<td>7 (6.34, 2.30)</td>
</tr>
<tr>
<td>1ACB</td>
<td>3 (6.17, 3.12)</td>
<td>2</td>
<td>3 (8.24, 4.31)</td>
</tr>
<tr>
<td>1AY7</td>
<td>14 (9.78, 5.27)</td>
<td>3</td>
<td>5(^*) (1.37, 0.77)</td>
</tr>
<tr>
<td>1BTH</td>
<td>1(^*) (12.10, 3.65)</td>
<td>1</td>
<td>2 (10.11, 3.28)</td>
</tr>
<tr>
<td>1CGI</td>
<td>2 (3.82, 2.31)</td>
<td>1</td>
<td>1(^*) (2.82, 2.25)</td>
</tr>
<tr>
<td>1DFJ</td>
<td>1(^*) (6.84, 2.76)</td>
<td>4</td>
<td>1(^*) (5.55, 2.03)</td>
</tr>
<tr>
<td>1E6E</td>
<td>None</td>
<td>0</td>
<td>474 (6.01, 3.32)</td>
</tr>
<tr>
<td>1FIN</td>
<td>None</td>
<td>0</td>
<td>None</td>
</tr>
<tr>
<td>1G1I</td>
<td>3 (6.06, 3.24)</td>
<td>6(^†)</td>
<td>25 (6.83, 3.37)</td>
</tr>
<tr>
<td>1GOT</td>
<td>None</td>
<td>0</td>
<td>None</td>
</tr>
<tr>
<td>1IBR</td>
<td>32 (6.99, 2.78)</td>
<td>0</td>
<td>2(^*) (5.01, 2.50)</td>
</tr>
<tr>
<td>1OAZ</td>
<td>58 (18.47, 3.84)</td>
<td>0</td>
<td>9(^*) (14.41, 3.27)</td>
</tr>
<tr>
<td>1PXV</td>
<td>51 (8.54, 4.03)</td>
<td>0</td>
<td>17 (6.94, 3.49)</td>
</tr>
<tr>
<td>1TEG</td>
<td>4 (8.10, 1.75)</td>
<td>1</td>
<td>1(^*) (6.83, 1.33)</td>
</tr>
<tr>
<td>1TGS</td>
<td>15 (2.69, 1.54)</td>
<td>1</td>
<td>1(^*) (1.94, 1.43)</td>
</tr>
<tr>
<td>1WQ1</td>
<td>6(^*) (2.24, 1.42)</td>
<td>1(^†)</td>
<td>20 (5.64, 2.35)</td>
</tr>
<tr>
<td>1ZHI</td>
<td>134 (13.44, 2.81)</td>
<td>0</td>
<td>10 (7.52, 2.73)</td>
</tr>
<tr>
<td>2BUO</td>
<td>1(^*) (9.38, 5.39)</td>
<td>9(^†)</td>
<td>3 (5.05, 3.91)</td>
</tr>
<tr>
<td>2KAI</td>
<td>17 (12.46, 3.23)</td>
<td>1</td>
<td>1(^*) (1.94, 0.84)</td>
</tr>
<tr>
<td>3HHR</td>
<td>214(^*) (11.61, 3.27)</td>
<td>0</td>
<td>497 (9.19, 5.51)</td>
</tr>
<tr>
<td>wins(^‡)</td>
<td>6</td>
<td>4</td>
<td>8</td>
</tr>
</tbody>
</table>

Table 3.4. Refinement of Right-Body Docking Solution Candidates (for Unbound Receptors and Bound Ligands).

\(^a\) The rank of the first acceptable solution. The RMSD and the iRMSD of this solution are in brackets in the corresponding order.

\(^b\) The number of acceptable solutions in the top 20 solutions.

\(^c\) The rank of the original PatchDock solution of the first acceptable solution, before the refinement, according to the shape complementarity score of PatchDock. The RMSD and the iRMSD of this solution are in brackets in the corresponding order.

\(^*\) The best rank of the first acceptable solution among the three methods (PatchDock, FireDock, and FiberDock).

\(^†\) The highest number of acceptable solutions in the top 20 solutions of the three methods.

\(^‡\) The wins row summarizes the number of cases that a method achieved the best rank of the first acceptable solution or the highest number of acceptable solutions in the top 20 solutions.
3.3 Discussion and Conclusions

The structure prediction of protein–protein complexes usually consists of two major stages: soft rigid-docking, which allows a certain amount of steric clashes, followed by flexible refinement. CAPRI challenges [64,133] showed that in many cases the rigid-docking stage succeeds in producing a near-native result. However, this result often contains steric clashes, and therefore it is ranked low in the list of solution candidates. The goal of the flexible refinement stage is to refine thousands of rigid-docking solutions, resolve their steric clashes, and evaluate their binding energies which are used for reranking. This is an extremely important stage that is necessary for identifying near-native models among a group of docking candidates and to create even more accurate models which will help scientists study and understand the chemical mechanism of molecular complexes.

In this chapter, we presented a new method for flexible refinement of docking solution candidates, called FiberDock. The method models both side-chain and backbone flexibility and performs rigid body optimization on the ligand orientation. The refinement algorithm mimics an induced-fit process. The backbone and side-chain movements are modeled according to the vdW forces between the receptor and ligand. The backbone movements are modeled using the NMA approach. Unlike previous methods [53,92] FiberDock uses both low- and high-frequency normal modes, and therefore it is able to model both global and local conformational changes such as opening of binding sites and loop movement. The results show that the method successfully models backbone movements that occur during molecular interactions. The inclusion of the backbone refinement procedure in the refinement process was shown to improve both the accuracy and the ranking of near-native docking solution candidates. Moreover, accounting for backbone flexibility improves the shape of energy funnels around the native docking orientation. These energy funnels can assist in identifying near-native solutions among a group of solution candidates.

Modeling backbone flexibility is necessary not only in cases where the proteins change conformation upon binding but also in cases where the 3D structures of the interacting proteins are not available and models are used. Backbone refinement might be able to deal with the inaccuracy of the models in these cases. In addition, we expect FiberDock to be helpful in predicting antibody–antigen complexes. Docking in this field is known to be difficult due to the flexible CDR loops. Our dataset contained a single antibody–antigen case (HIV-1 neutralizing antibody in complex with its V3 loop peptide antigen, pdb-id: 1GGI) in which a CDR loop moves upon binding. FiberDock improves the refinement of this complex, when compared with FireDock (see Figure 3.2). We plan to further investigate the performance of the FiberDock method on antibody–antigen complexes.

Currently, the FiberDock method is particularly helpful in cases where the receptor undergoes an opening conformational change induced by steric clashes. In general, opening movements are easier to handle in docking as modeling the precise alternate conformation is not essential
for generating an accurate model of the molecular complex with low-energy score. The current version of FiberDock will not model movements induced by attractive forces, such as closing of a binding site around the ligand, as the correlation measurement, used for selecting the relevant modes, uses only the repVdW forces. In the future, we plan to incorporate additional chemical forces (e.g., attractive vDW forces, electrostatic forces, and hydrogen bonds) in the normal modes selection step of the backbone refinement procedure.

FiberDock deals with relatively subtle backbone conformational changes that occur upon binding. It achieves good refinement results in cases where the receptor interface RMSD (reclRMSD) is below 5Å. In cases with larger conformational changes, an initial near-native rigid-docking solution cannot be generated, and therefore other approaches should be considered. An analysis should be performed prior to the docking to assess the level of flexibility of the interacting proteins. One of the common types of backbone flexibility is a hinge bending motion. Hinge locations can be predicted by the HingeProt method [50], which analyzes the two lowest frequency normal modes. Hinge motions usually result in a large conformational change that prevents any rigid-docking method from generating a near-native model. In these cases, one can perform flexible docking by the FlexDock method [89]. This method divides the flexible protein into its rigid parts, dock each part separately and then assemble the partial docking solutions into consistent flexible docking models. Hinge-bending motions are often coupled with other types of backbone flexibility (e.g. flexible loops). These can be handled by refining FlexDock solutions using the FiberDock method.

In other cases where a high level of backbone flexibility is predicted, cross-docking of pre-generated conformations should be performed, followed by flexible refinement of the solutions. This will mimic both the conformational selection process and the induced-fit process. A similar approach was recently tested by Chadhury and Gray [159] with promising results. However, cross-docking might produce many more solutions with good energy values. Therefore the identification of near-native solutions among them will be more difficult. To correctly rank the solution candidates, a more accurate and robust energy function should be developed, and energy funnels should be searched around the lowest energy solutions.

3.4 FiberDock Web Server

Today, there are many freely available web servers that deal with different aspects of the docking field. Rigid-body docking can be performed by PatchDock [155], ZDOCK [75], GRAMM-X [160], Hex [161] and SymmDock [155]. ClusPro [162] filters, clusters and ranks docking solution candidates. The RosettaDock web server [163] performs local search in the vicinity of a single given input complex structure by optimizing rigid-body orientation and side-chain conformations. The NOMAD-Ref server [164] uses normal mode analysis to refine one of the molecules in a single-docking model. The FireDock web server [165], refines the rigid-body orientation and side-chain conformations of up to 1000 rigid-body solution candidates and re-
scores the refined structures according to a binding energy function. The HADDOCK web server [34] performs experimental data-driven docking followed by a semi-flexible refinement.

The FiberDock web server, presented here, is the first docking refinement web server that handles both backbone and side-chain flexibility and optimizes the relative rigid-body orientation of the proteins. It has a clear and user-friendly user interface and it requires no previous knowledge in docking algorithms. The web server is free and available with no login requirement at http://bioinfo3d.cs.tau.ac.il/FiberDock/. It refines a single rigid-body docking solution in an average time of 14s. Therefore, it can be used for refining and re-ranking of up to 100 solutions in a reasonable time. The FiberDock software (for Linux users) can also be downloaded from the web site. The downloaded version does not restrict the amount of refined docking solutions. We believe that this server will be very useful to the biological community. It can help model new structures of protein–protein complexes and as such improve our understanding of protein functions in the living cell.

### 3.4.1 Input

The FiberDock server can refine up to 100 rigid-docking solution candidates. The user uploads or specifies codes of two PDB files (from the Protein Data Bank [166]), receptor and ligand, and provides a list of up to 100 transformations. Each transformation, when applied on the ligand, produces a candidate docking solution. If no transformation file is uploaded the identity transformation is used. Alternatively, the user can upload a PDB file that contains the rigid-docking solutions as a set of models. The candidate solutions for FiberDock can be generated by any rigid-body docking methods favored by the user (such as PatchDock [79,155], ZDOCK [74,75], GRAMM-X [160], Hex [161], etc.). In addition, the user can choose whether to model backbone movements or not. The user can also specify an e-mail address to which a link to the output web page, containing the results, will be sent when the refinement process is finished.

The server also includes optional advanced parameters for adjusting the refinement and scoring parameters for a specific biological system. These parameters are divided into four groups according to the refinement stage they affect.

For the side-chain optimization stage, the user can decide if the optimization will be preformed on both proteins, one of them or none. In addition, the user can specify the level of side-chain optimization: restricted or full. When the restricted level is chosen, only the side chains that form steric clashes will be allowed to move. The full side-chain optimization level will allow all the side chains in the protein–protein interface to be flexible. By default, the restricted level is chosen, because studies have shown that many of the side chains in the interface keep their unbound conformation within a complex [66,130,132].

The parameters of the backbone refinement stage include the number of lowest frequency normal modes that will be considered in the refinement. By specifying a small number (10 for example), the user restricts the backbone movements to be relatively global, whereas a high
number of normal modes will allow the algorithm to use high-frequency modes, which describe local movements (if they correlate well with the chemical forces that the proteins apply on each other). In addition, the user can set the level of backbone flexibility. In order to prevent the backbone from over distorting, a penalty term is introduced into the backbone minimization step. The level of backbone flexibility determines the weight of this penalty term. The higher the level, the lower the weight. A value of 0.95 (the default value) was found to suit most of our test cases.

For the rigid-body optimization stage, the user can set the number of MC iterations. In general, increasing this value improves the search for a local minima in the vicinity of the ligand’s current position. However, according to our experience, the optimization usually converges after 50 iterations.

The complex type parameter (Default, Antibody-Antigen or Enzyme-Inhibitor), is used for adjusting the weights of the scoring function for a specific biological system. The parameter of atomic radius scale influences the extent of acceptable steric clashes in the final refined solutions. This parameter scales down the radius of the atoms, affecting the VdW terms that are used in all of the three refinement stages and the final calculated binding energy.

3.4.2 Output

When the refinement is finished, a web page with the results is generated and a link to it is sent to the e-mail address specified by the user. This web page (Figure 3.10) contains a table in which each row corresponds to a single refined solution. Each row specifies the rank of the solution according to the binding energy value, its original number (according to the given transformation file), the global binding energy value and the values of four of the energy terms (Attractive VdW, repulsive VdW, ACE and hydrogen bonds). The table is sorted by the binding energy of the refined solution. The user can view the 3D structure of each refined complex in a Jmol applet window [167]. The different structures can be viewed simultaneously, allowing the user to easily compare different models. The PDB files of the refined solutions can be downloaded, and so can the full results table that details the values of all the energy terms, for each solution. This table also specifies the linear combination of normal modes that generates the refined backbone conformation of the receptor and the ligand.
**Figure 3.10.** The output of FiberDock web server. The results are presented in a table, sorted by the binding energy value. The user can view the 3D structure of each of the refined complex in a Jmol applet window. The figure was taken from Mashiach et al. 2010.
Chapter 4 :
SymmRef - a Flexible Refinement Method for Symmetric Multimers

4.1 Introduction to Symmetric Docking

The majority of protein-protein complexes are symmetric oligomers [168,169]. Symmetric complexes have many functional, genetic and physicochemical advantages, which can explain their abundance in the living cell [169]. Symmetric protein complexes often form structural arrangements with specific functions, such as channels, containers, and filaments.

Monod et al. [170] suggested that cyclic symmetric complexes are common because they occupy all of the available binding sites, which reduces aggregation. Moreover, once initially formed, the impact of mutations during evolution is doubled compared to hetero-oligomers, which might increase the evolutionary rate of improving the binding affinity. However, André et al. [171] clarified that although the influence of each mutation is doubled, the number of unique amino acids in homo-oligomers is half of that in hetero-oligomers and therefore the mutation rate is halved. They suggested an alternative theory that explains the high frequency of symmetric complexes. When the mutation rate is halved but the impact of each mutation is doubled, the mean of the energetic contributions does not change, but the variance doubles. Increasing the variance drastically improves the probability of symmetric protein complexes with very-low energy to initially be formed by evolution.

In nature, one can find few types of symmetry. The most common is the cyclic symmetry (C_n), which contains a single axis of rotational symmetry [169]. C_2 symmetry (homodimers) is the most frequent among the various C_n symmetries. Dihedral symmetry (D_n) is another common type. These complexes combine an axis of rotational symmetry and a perpendicular axis of two-fold symmetry. Icosahedral symmetry produces spherical assemblies and helical symmetry is produced by rotation and translation along a single symmetry axis.

Despite their high frequency, the large size of symmetric complexes makes them hard to be structurally characterized using experimental means. However, computational prediction is significantly facilitated when symmetry constraints are applied. In docking, symmetry provides powerful selectivity that considerably improves the success rate compared to general protein-protein docking. The number of degrees of freedom in the rigid-docking search-space of symmetric complexes is four, compared to six in heterodimers. André et al. [171] have demonstrated the power of the symmetric selectivity by showing that only one out of 5,000 random orientations of two identical monomers is near-symmetric (with a symmetry measure that is typical for known symmetric dimers).
In practice, docking methods should attempt to predict the structure of oligomers given the unbound conformation of the monomers. Since proteins often undergo conformational changes upon binding which include both backbone and side-chain movements, this makes the docking problem much more difficult.

To date there are six major methods for docking prediction of symmetric multimers: MolFit [172,173], SymmDock [88,155], ClusPro [174], M-ZDOCK [175], ROSETTA [176] and HADDOCK [177]. These methods can handle multimers with different types of symmetry and some of them model conformational changes of the monomers, as detailed in Table 4.1.

Berchanski and Eisenstein were the first to tackle the challenge of symmetric docking by developing a method for predicting the structures of homo-multimers with $D_2$ symmetry [173]. First, structures of dimers are predicted by an FFT-based docking method [71]. Then, the solutions are filtered and only the symmetry-related homodimers are selected. Finally, the full $D_2$ complex is assembled by re-docking each dimer solution from the previous step to itself or by superimposing the monomers of two different dimer solutions. This method was later extended to handle $C_n$ and $D_n$ symmetries [172].

SymmDock is a geometry-based rigid-docking method for protein complexes with cyclic symmetry [88,155]. The method calculates matching pairs of surface points across the interface, computes the symmetry axes consistent with large numbers of such matching pairs, and then ranks the resulting candidate symmetric solutions by a shape complementarity score. SymmDock is extremely fast. It is able to successfully predict the structure of symmetric complexes with $C_n$ symmetry in a few minutes on a standard desktop PC, for any $n$. The efficiency of the method results from the fact that it a-priori limits the search space to symmetric transformations.

M-ZDOCK performs symmetric rigid-docking by an FFT approach that explores only the search space of $C_n$ symmetric rigid-transformation (four degrees of freedom) [175]. The method searches for symmetric complexes that optimize not only a surface complementarity metric but also desolvation and electrostatics energy terms.

The ClusPro symmetric docking method attempts to predict both the symmetry type (cyclic or dihedral) and the structure of the complex [174]. It starts by generating a large set of rigid body docking solutions of a monomer with itself, using the DOT program [73]. It identifies the 2000 energetically favorable solutions and scores all the solutions of DOT according to the number of their energetically favorable neighbor solutions. Then, the algorithm generates all possible symmetric complexes that are induced by the rigid-docking solutions (cyclic and dihedral symmetry). Finally, it clusters and ranks them according to the scores of the docking solutions, from which they were built.

The ROSETTA symmetric docking method optimizes the rigid-body orientations of the monomers and their backbone and side-chain conformations, while restricting the search space
to symmetrical models [176]. The optimizations are performed by a Monte-Carlo-Minimization (MCM) protocol, starting from random symmetrical configurations.

The HADDOCK multi-body docking method [177] requires experimental data (and/or bioinformatics data) for docking. The method minimizes a function which includes: various energetic terms, a term of ambiguous interaction restraints (AIRs), determined by the available experimental data, and terms of symmetry restraints. The minimization accounts for torsion angle flexibility in flexible segments, which are typically defined automatically by intermolecular contacts.

The ROSETTA and the HADDOCK symmetric docking methods restrict all the monomers to be identical. However, most of the crystal structures of symmetric protein complexes show local differences in side-chain conformations and small variations in backbone conformation[169]. An example of such local differences is shown in Figure 4.1.

In this chapter we present SymmRef, a novel docking refinement method for symmetric protein complexes. The method refines a set of rigid docking solution candidates which can be generated by any symmetric docking method (e.g. SymmDock [88] or M-ZDOCK [175]). It models both side-chains and backbone movements and re-ranks the refined models by an energy scoring function. The optimization of side-chain conformations is performed by an Integer Linear-Programming approach. This approach is very efficient and promises to assign the optimal rotamers to the interface side-chains. Unlike other methods, SymmRef does not apply symmetry constraints at the side-chain optimization level. Backbone refinement by SymmRef is performed by normal modes minimization. While previous methods model backbone movements only in pre-selected backbone segments [176,177], SymmRef is able to model both local and global conformational changes, which move the entire backbone simultaneously. The results show that the method improves both the accuracy and the ranking of rigid-docking solution candidates, and outperforms existing state-of-the-art symmetric docking methods. SymmRef is available for download at http://bioinfo3d.cs.tau.ac.il/SymmRef/download.html.

<table>
<thead>
<tr>
<th>Method</th>
<th>Symmetry type</th>
<th>Flexibility</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>M-ZDOCK</td>
<td>Cyclic</td>
<td>None</td>
<td>Ref. [175]</td>
</tr>
<tr>
<td>SymmDock</td>
<td>Cyclic</td>
<td>None</td>
<td>Ref. [88,155]</td>
</tr>
<tr>
<td>MolFit</td>
<td>Cyclic, dihedral</td>
<td>None</td>
<td>Ref. [172,173]</td>
</tr>
<tr>
<td>ClusPro</td>
<td>Cyclic, dihedral</td>
<td>None</td>
<td>Ref. [174]</td>
</tr>
<tr>
<td>ROSETTA</td>
<td>Cyclic, dihedral, helical, icosahedral</td>
<td>Full interface flexibility</td>
<td>Ref. [176]</td>
</tr>
<tr>
<td>HADDOCK</td>
<td>Oligomers with arbitrary symmetry</td>
<td>Full interface flexibility</td>
<td>Ref. [177]</td>
</tr>
</tbody>
</table>

*Table 4.1.* Existing docking methods for symmetric multimers.
Figure 4.1. Local differences in side-chain conformations of the monomers of the Flavivirus envelope glycoprotein (PDB-ID: 1URZ). (A) Superimposition of the three monomers. The interface residues are shown in sticks presentation. (B) The interface between two monomers. The native residue conformations are in red, a duplication of the rotamers of one of the monomers (chain A) is in blue and the refined conformations by the side-chain optimization of SymmRef is in green. In the square, one can see that enforcing side-chain conformations to be identical in all the monomers results in steric clashes and high repulsive electrostatic energy (the squared area show a steric clash between Arg217 in monomers A and B). By allowing local differences between the monomers, SymmRef was able to find an alternative rotamer (similar to the native one) that resolved the clash. The figure was taken from Mashiach-Farkash et al. 2011.

4.2 Methods

The SymmRef method refines a set of rigid docking solutions of protein complexes with cyclic symmetry and improves the accuracy and the ranking of the near native models. The method optimizes the rigid-body orientation of the monomers, assigns optimal rotamers to interface side-chains and minimizes the backbone conformation.
The input to the algorithm is a set of hundreds or thousands docking solutions produced by any docking method of choice (e.g. SymmDock [88] or M-ZDOCK [175]). For each docking solution, the refinement includes the following steps:

1. Repeat N times:
   1.1. Fast rigid-body Monte-Carlo minimization.
   1.2. Restricted side-chain optimization.
   1.3. Backbone refinement.
   1.4. Extensive rigid-body Monte-Carlo minimization.
   1.5. Full side-chain optimization
   1.6. Scoring.
2. Return the solution with the best score.

In this study we repeated stage 1 of the algorithm 10 times (N=10). Each iteration results in a different refined solution due to the random Monte-Carlo process. The method is expected to produce better results when more iterations are performed. However, there is a tradeoff between performance and running-time that the user can control. The running time of a single refinement iteration depends on the size of the interface and the extent of steric clashes in the starting docking model. The average running time of a single iteration on our data set was around 23 seconds on a 2.33GHz Intel(R) Xeon(R) CPU with 16G of memory. The refinement can be performed on a set of hundreds or thousands of rigid docking solutions and it can be easily parallelized and be run on a cluster of computers, where each process refines a subset of the rigid-docking solution candidates.

4.2.1 Side-chain optimization

The side-chain optimization stage assigns the optimal conformation to each interface residue, from a set consisting of common rotamers (backbone dependent) and the unbound conformation. In a restricted side-chain optimization (step 1.2 in the SymmRef algorithm) only residues that are in a steric clash are allowed to be moved, and in the full optimization (step 1.5 in the SymmRef algorithm) the movements of all interface residues are modeled. SymmRef optimizes the side-chain conformations in the interface of a single monomer and its neighboring monomers. We do not restrict the side-chain conformations to be identical in each monomer, since solved crystal structures of homo-oligomers show that the conformation of side chains often vary between different monomers in a symmetric complex. For example, local differences in side-chain conformations of symmetric complexes can be seen in the crystal structure of Insulin [178], HIV-1 protease [169] and the Flavivirus envelope glycoprotein (Figure 4.1A). An extensive analysis of side-chain conformations in symmetric complexes is detailed in the Results.

The optimization is performed by the integer linear programming approach described by Andrusier et al. [113] with a modification in the scoring function that is being minimized.
Minimize:

\[
\sum_{(i,r)} E_{ISCO}(i_r) \cdot y_{i_r} + \sum_{(i,r,s); i<j, j\in N(i)} E_{ISCO}(i_r, j_s) \cdot x_{i_r, j_s}
\]

Subject to:

\[
\sum_{r} y_{i_r} = 1, \quad i \in V
\]

\[
\sum_{r} x_{i_r, j_s} = y_{j_s}, \quad i \in V, j \in N(i)
\]

\[
y_{i_r}, x_{i_r, j_s} \in \{0,1\}
\]

Where \(y_{i_r}\) and \(x_{i_r, j_s}\) are decision variables. Setting \(y_{i_r}\) to 1 corresponds to choosing rotamer \(r\) for residue \(i\), and similarly setting \(x_{i_r, j_s}\) to 1 corresponds to choosing to include the energy between rotamers \(r\) of residue \(i\) and rotamer \(s\) of residue \(j\), in the minimized energy function.

\(E_{ISCO}(i_r, j_s)\) and \(E_{ISCO}(i_r)\) are the pairwise and self energy terms used for interface side-chain optimization. \(E_{ISCO}(i_r, j_s)\) is the energy between two rotamers \((r\) and \(s\)) of two movable residues \((i\) and \(j\)), and \(E_{ISCO}(i_r)\) is the energy between rotamer \(r\) of residue \(i\) and its static environment (backbone and other fixed atoms). These energy terms include repulsive van der Waals energy \(E_{\text{rep,vdW}}\), repulsive electrostatic energy \(E_{\text{rep,Elec}}\) and internal energy score of the chosen rotamers \(E_{\text{rot}}\), as described below:

\[
E_{ISCO} = E_{\text{rep,vdW}} + E_{\text{rep,Elec}} + 6 \cdot E_{\text{rot}}
\]

For the detailed formulas of these energy terms the reader is referred to Andrusier et al. [113] While in our previous docking refinement methods [113,179] the side chain optimization minimized a scoring function which was composed only of a repulsive van der Waals energy term and rotamer internal energy score, here we added electrostatics. We found the electrostatics to be particularly important in symmetric complexes since the symmetry often causes a residue to interact with the same residue in the neighboring monomer. This means that positive residues often interact with positive residues and negative residues interact with negative residues. Accounting for electrostatics in the side-chain optimization can resolve these electrostatic repulsive forces by choosing optimal rotamers.

### 4.2.2 Rigid-body Monte-Carlo minimization

Given a symmetric transformation of a docking solution candidate, the rigid-body minimization procedure aims to optimize the van der Waals binding energy of the complex by minimizing the energy in the four-dimensional space of symmetric rigid movements, one translational (the distance from the symmetry axis) and three rotational degrees of freedom. In this stage we perform Monte-Carlo (MC) sampling with local minimizations while restricting all transformations to be symmetric.
In a symmetric complex the interface between any monomer and its neighboring monomers is very similar (not identical since we assume that side-chain conformations may vary). Therefore, we optimize the orientation of a single monomer by minimizing the binding energy of its interaction with both its neighbors. In the case of a symmetric homodimer ($C_2$ symmetry), we minimize the binding energy of the single interface.

Given a model of a $C_k$ symmetric complex, let $M_1$ be the coordinates of one of the monomers, and let $T$ be the symmetric transformation that transforms the monomer to the orientation of its neighboring monomer in the model. We can calculate the coordinates of the neighboring monomers, $M_2$ and $M_k$, by the following formulas:

$$M_2 = T \cdot M_1$$
$$M_k = T^{-1} \cdot M_1$$

Let $P$ be a subtle rigid-body perturbation, sampled randomly from the four rigid-body degrees of freedom discussed above, and let $M'_i$ be the coordinates of monomer $i$ after the perturbation $P$, as described by the following formula:

$$M'_i = T^{i-1} \cdot P \cdot M_1$$

By applying the $P^{-1}$ transformation on all the monomers $M'_i$, we will get a new symmetric complex which is slightly different from the original one, and in which the orientation of $M_1$ remains the same. The new symmetric transformation is $T(P) = P^{-1} \cdot T \cdot P$. It is still a $C_k$ symmetric transformation, for any $P$, since $T(P)^k = (P^{-1} \cdot T \cdot P)^k = P^{-1} \cdot T^k \cdot P = P^{-1} \cdot P = I$. The rigid body minimization procedure aims to find a local perturbation $P^*$ which minimizes the binding van der Waals energy of $M_1$ and its two neighbors: $E_{RBM}(T,P) = E_{RBM}(M_2,T(P) \cdot M_1) + E_{RBM}(M_k,T(P)^{-1} \cdot M_1)$. The van der Waals energy between two atoms is defined as the modified Lennard-Jones 6–12 potential with linear short-range repulsive score, as detailed in Mashiach et al [179]. The energy calculation accounts for different side-chain conformations of the neighboring monomers, as could be predicted by a previous run of the side-chain optimization procedure.

The rigid-body optimization procedure starts with a given $C_k$ symmetric transformation $T$, and performs the following repeated steps:

1. Generate a random perturbation $P$.
2. Local minimization of $E_{RBM}(T,P)$ (in 4 degrees of freedom), by the BFGS quasi-Newton algorithm [151,180].
3. The obtained position is accepted or rejected by the Metropolis criterion: If the new position results in a lower energy score, the move is unconditionally accepted. Otherwise, it is accepted with some probability. If the position is accepted: $T = P^{-1} \cdot T \cdot P$
4. Go to 1.
The rigid-body MC Minimization procedure is performed twice in the SymmRef algorithm. At the beginning of the refinement (step 1.1 in the algorithm) a short MC minimization, with 10 iterations, is performed. The goal of this step is to remove major steric clashes that may occur in the initial docking model that is being refined. Later in the algorithm, after refining the side-chains and backbone conformations, a second minimization of the rigid-body orientation is performed (step 1.4). In the second MC minimization 50 iterations are performed. The goal here is to optimize the packing of monomers in the complex. 50 iterations were found to be adequate for finding a satisfactory local minimum in the energy landscape.

4.2.3 Backbone refinement

Backbone refinement is performed by a normal mode based approach which was developed for general protein-protein docking refinement in the FiberDock method [179]. Normal modes are a set of predicted movements that a protein is likely to undergo. The conformational change that a protein undergoes upon binding (from the unbound conformation to the bound conformation) can be described as a linear combination of normal modes. In the backbone refinement procedure of SymmRef we attempt to predict the coefficients of this linear combination. We adjusted the backbone refinement algorithm of FiberDock to restrict the backbone conformations of all the monomers in the complex to be identical, in order to preserve the symmetry of the complex. This was done by applying the same linear combination of normal modes on all the monomers simultaneously during the refinement. The algorithm is briefly described below. For more details the reader is directed to the FiberDock paper [179].

In a pre-processing stage, the normal modes of the unbound structure of the monomer are calculated by using the anisotropic network model (ANM) [42]. During the backbone refinement procedure, the following steps are performed:

1. Rigid-body minimization by the BFGS quasi-Newton algorithm [151,180].
2. Repeat the following steps until the energy score converges:
   2.1. Calculate the van der Waals forces between two neighboring monomers.
   2.2. Identify the 10 normal modes with the best correlation to the calculated vdW forces and minimize the backbone conformation of the monomers along them and along the rigid-body degrees of freedom,
   2.3. Fast rigid-body Monte-Carlo minimization.

Step 2 is repeated until the energy score converges to a local minimum (5 iterations without energy improvement). In order to shorten the running time of the algorithm we restrict the number of iterations to be below 20, a sufficient number for most of the cases for producing satisfactory results. Additionally, we stop the backbone refinement procedure when all the steric clashes in the complex are solved, i.e. if the repulsive vdW energy is below a threshold.

In step 2.3 we perform 10 iterations of rigid-body MC minimization in order to optimize the packing of the monomers in the complex after the backbone movement.
4.2.4 Ranking

The ranking of the refined docking solutions is performed by an energy scoring function. We used the default energy score from our previous refinement methods, FireDock [113] and FiberDock [179]. The weights of this energy score were optimized by using a machine learning algorithm and heuristic optimization on a test set of asymmetric protein-protein complexes from a docking benchmark [181], as described in the FireDock paper [113]. The performance of this energy score was shown to be successful on a docking benchmark and in the CAPRI experiments [182]. Kastritis et al have recently shown that this energy score has the highest correlation to experimental binding affinities compared to the energy scores of other docking methods [183]. However, this energy score is still not sufficiently accurate for predicting binding affinities.

We slightly adapted the weights of the energy terms to symmetric complexes. We could not use a machine learning algorithm to optimize the weights for symmetric multimers due to the small number of cases of symmetric complexes with solved unbound structures. Consequently, this was done manually based on specific properties of symmetric complexes as described below. Symmetric complexes often include charged residues which interact with the same residue in neighboring monomers. A slightly inaccurate modeling of the interface side-chain conformations may cause a high calculated repulsive electrostatic energy. Therefore, in order to improve the robustness of the energy score we penalize less for electrostatic repulsion. We have reduced the short range repulsive electrostatics weight from 0.21 to 0.15 and the long-range repulsive electrostatics weight from 0.69 to 0.1. These values were manually adjusted by a process of trial and error. In addition, we removed the heuristic insideness energy term which is not relevant for symmetric complexes. The other weights were unchanged. The full energy score is detailed in the following formula:

\[ E_{symm} = 1.5E_{attrVdW} + 0.8E_{repVdW} + 1.6E_{ACE} + 0.21E_{attrEl} + 0.15E_{repEl} + 0.1E_{repEl} \\
+ 1.2E_{HB} + E_{pipl} + 0.7E_{catpl} + 2.5E_{aliph} + 0.05E_{deform} \]

The adjustment of the energy scoring function was tested on an independent dataset of arbitrarily selected bound docking cases in order to rule out over-fitting (see Results section).

4.2.5 Dataset

In order to test the performance of our refinement method we created a dataset of 16 unbound docking cases. There are very few cases in the Protein Data Bank in which the three dimensional structure of a monomer was solved both in a symmetric complex and alone in the unbound conformation. We identified 7 such cases and included them in our dataset. In addition we added 9 other cases in which the unbound structure of the monomer was not experimentally solved but a structure of a homologue protein was available. In these cases we modeled the structure of the monomer according to the homologue and used this model as the unbound structure in our unbound docking experiments. The homology modeling was performed by the
MODELLER [13] method using sequence alignment that was generated by STACCATO [184]. We trimmed edges and loops that had no template according to the sequence alignment. The details of the dataset are in Table 4.2.

Some of the symmetric complexes in our dataset have dihedral symmetry. The current version of SymmRef can handle only cyclic symmetry. Therefore, in these cases we aimed to predict the structure of the dimers that compose the full dihedral complexes.

Our dataset includes two targets of the CAPRI experiment, target 10 (1URZ) [64] and 42 (2WQH) [185]. The structure of the TBEV envelope protein of the tick-borne encephalitis virus (1URZ) includes a flexible C-terminal domain (from residue 299). In the CAPRI challenge, the predictors were advised to ignore this domain; therefore we removed it from the structure in our dataset as well.

The trimeric form of bovine pancreatic RNase (1JS0) reveals a domain-swap of a beta strand. This conformational change increases drastically the binding energy of the complex. However, we do not expect SymmRef, or any other existing docking method, to predict this conformational change without prior knowledge of the domain swap. Therefore we ignored the residues of this beta strand in the RMSD calculations.

<table>
<thead>
<tr>
<th>System</th>
<th>PDB(a)</th>
<th>Unbound(b)</th>
<th>RMSD(c)</th>
<th>IRMSD(d)</th>
<th>C_n/D_n(e)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Flavivirus envelope glycoprotein*</td>
<td>1URZ</td>
<td>1SVB</td>
<td>4.33</td>
<td>4.97</td>
<td>C_3</td>
</tr>
<tr>
<td>Hypothetical protein*</td>
<td>1VIM</td>
<td>1V1Y(HU)</td>
<td>3.10</td>
<td>3.53</td>
<td>D_2</td>
</tr>
<tr>
<td>HIV gp41 core*</td>
<td>1F23</td>
<td>3CP1(HB)</td>
<td>0.63</td>
<td>0.63</td>
<td>C_3</td>
</tr>
<tr>
<td>TR-beta variant*</td>
<td>3D57</td>
<td>1NAX(HU)</td>
<td>0.88</td>
<td>0.78</td>
<td>C_2</td>
</tr>
<tr>
<td>A boiling stable protein SP1*</td>
<td>1TR0</td>
<td>1Q4R(HB)</td>
<td>1.00</td>
<td>1.05</td>
<td>D_6</td>
</tr>
<tr>
<td>N-ethylmaleimide-sensitive fusion protein*</td>
<td>1D2N</td>
<td>1Y2(HU)</td>
<td>5.29</td>
<td>5.35</td>
<td>C_6</td>
</tr>
<tr>
<td>Domain-swatched RNase*</td>
<td>1JS0</td>
<td>9RAT</td>
<td>6.97</td>
<td>8.69</td>
<td>C_3</td>
</tr>
<tr>
<td>Half of BPTI decamer</td>
<td>1BOC</td>
<td>3PTI</td>
<td>0.51</td>
<td>0.54</td>
<td>C_5</td>
</tr>
<tr>
<td>Phospholipase A2</td>
<td>1A3F</td>
<td>1POA</td>
<td>0.77</td>
<td>0.82</td>
<td>C_3</td>
</tr>
<tr>
<td>Hexamerical Capsomer of HIV-1*</td>
<td>3MGE</td>
<td>1E6J</td>
<td>10.05</td>
<td>10.12</td>
<td>C_6</td>
</tr>
<tr>
<td>Self-association of TPR domains*</td>
<td>2WQH</td>
<td>1NA0</td>
<td>0.86</td>
<td>0.16</td>
<td>C_2</td>
</tr>
<tr>
<td>Bacteriorhodopsin</td>
<td>1AP9</td>
<td>3HAR</td>
<td>1.66</td>
<td>1.39</td>
<td>C_3</td>
</tr>
<tr>
<td>Karyopherin α</td>
<td>1BK6</td>
<td>1IAL(HU)</td>
<td>2.81</td>
<td>2.18</td>
<td>C_2</td>
</tr>
<tr>
<td>TNF Receptor Associated Factor 2</td>
<td>1D01</td>
<td>1LB4(HU)</td>
<td>2.68</td>
<td>1.23</td>
<td>C_3</td>
</tr>
<tr>
<td>Smad4 Active Fragment</td>
<td>1DD1</td>
<td>1M5(HU)</td>
<td>5.14</td>
<td>2.59</td>
<td>C_3</td>
</tr>
<tr>
<td>Yeast cuzznos exposed to nitric oxide</td>
<td>1F1G</td>
<td>1MF3(HU)</td>
<td>1.15</td>
<td>1.67</td>
<td>C_3</td>
</tr>
</tbody>
</table>

Table 4.2. The dataset used for evaluating SymmRef performance.

(a) The PDB ID of the symmetric complex.
(b) The PDB ID of the unbound monomer or the homologue of the monomer (marked by (HU) if the homologue is in an unbound conformation and by (HB) if the homologue is in a bound conformation).
(c) The full backbone RMSD of the bound and unbound structure of the monomer.
(d) The backbone RMSD of the interface residues of the bound and unbound structure of the monomer.
(e) The Symmetry type.

* The cases that were used to manually adjust the electrostatics weights on the energy scoring function.
4.2.6 Docking Evaluation

Following the CAPRI challenge evaluation protocol [64] we used several evaluation criteria for assessing the performance of docking methods, as detailed below. Let $S$ be the native structure of a symmetric complex and let $M$ be a predicted model of the same complex which we would like to evaluate. Let $S_1$ and $S_2$ be two adjacent monomers in the native structure and let $M_1$ and $M_2$ be the corresponding monomers in the evaluated model. Let $I(S_1,S_2)$ be the amino acids of the interface between the adjacent monomers $S_1$ and $S_2$. An amino acid is considered to be in the interface if it has at least one atom within 10Å of the neighboring monomer.

RMSD – The root mean square deviation between the backbone atoms of $S_2$ and $M_2$ after superimposing the monomers $S_1$ and $M_1$.

IRMSD – The root mean square deviation between the backbone atoms of $I(S_1,S_2)$ and $I(M_1,M_2)$ after superimposition of the two interfaces.

$F_{\text{nat}}$ – The fraction of native contacts. A pair of residues from two adjacent monomers is considered to be in contact if any of their atoms are within 5Å.

We classified the accuracy of the predicted models into four categories according to the CAPRI criteria [64]: (1) Incorrect. (2) Acceptable accuracy (marked by one star - *). (3) Medium accuracy (**). (4) High accuracy (***)

4.3 Results

4.3.1 Analysis of backbone and side-chain conformations of symmetric complexes

Previous docking methods [176,177] assume that the conformations of monomers in a symmetric complex are identical. However, most of the crystal structures of symmetric protein complexes show local differences in side-chain conformations and occasional minor differences in backbone conformations. We analyzed the backbone and side chain conformations of 65 crystal structures of symmetric complexes (Table 4.3).

This analysis shows that the backbone conformation in the interface of symmetric complexes is almost identical in different monomers of a symmetric complex. In 82% of the cases the Cα interface RMSD was below 0.5Å, and in 95% of the cases it was below 1Å (Figure 4.3). Due to these findings we decided to use symmetry constraints on the backbone conformations of the monomers during the docking refinement process.

The analysis of the side-chain conformations showed that in 44 (68%) of the cases there was a variation in the side-chain conformation between two arbitrary monomers of the symmetric complex. Two side-chain conformations were considered to be different if the atomic RMSD exceeds 1Å, after superimposing the two residues based on their backbone atoms.
There are few factors that can break the symmetry of side-chain conformations in a symmetric complex. The first is self-interaction of an amino-acid with itself in an adjacent monomer. We define a self-interaction to be a case in which two corresponding residues in adjacent monomers have atoms within a distance of 4Å. For example, in the case of the Flavivirus envelope glycoprotein (Figure 4.1A), Arg217 has a self-interaction in the complex. One of the Arg217 residues (in chain A) adopts a conformation that enables it to form a favorable hydrogen-bond with a water molecule placed inside the complex. The other Arginines cannot adopt the same conformation without forming a steric clash with the first Arginine. In order to examine the impact of the self-interaction property on the symmetry of the side-chain conformations we compared the percentage of complexes with identical side-chain conformations in a group of symmetric complexes with and without self-interactions. Our dataset of 65 crystal structures of symmetric complexes included 28 cases without self-interactions. In this group 12 (43%) cases had identical side-chain conformations between two arbitrary monomers. In contrast, in the group of 37 structures with self-interaction, only 9 (24%) cases had identical side-chain conformations (Figure 4.2). Other factors that may break the symmetry of the side-chain conformations include side-chain flexibility on the surface of the protein due to interactions with the surrounding water molecules, backbone flexibility in flexible loops, asymmetric interactions with small ligands and interaction of a symmetric complex with another protein.

Due to these results we decided not to restrict the side-chain conformations of all the monomers to be identical during the side-chain optimization procedure. Figure 4.1B shows the results of the side-chain optimization procedure on the Flavivirus envelope glycoprotein. We started the optimization by duplicating one of the monomers (chain A) and superimposing it on the other monomers in the complex. This initial complex had a high binding energy score of 782.27, mostly due to high repulsive vdw energy of 1045.99 and high repulsive electrostatic energy of 223.23. The figure focuses on a steric clash between Arginine 217 from two
monomers in the complex. The side-chain optimization method rearranged the side-chain conformations and resulted in a model with a much lower energy score of -72.48 (repulsive vdW energy of 0.07 and repulsive electrostatic energy of 43.23). In this experiment we did not perform rigid-body minimization or backbone refinement.

<table>
<thead>
<tr>
<th>PDB</th>
<th>Symmetric Order</th>
<th>Self interaction</th>
<th>Different side-chain conformations</th>
<th>Backbone IRMSD</th>
</tr>
</thead>
<tbody>
<tr>
<td>1A3F</td>
<td>3</td>
<td>No</td>
<td>No</td>
<td>0.04</td>
</tr>
<tr>
<td>1EK9</td>
<td>3</td>
<td>No</td>
<td>No</td>
<td>0.16</td>
</tr>
<tr>
<td>1QU9</td>
<td>3</td>
<td>No</td>
<td>No</td>
<td>0.00</td>
</tr>
<tr>
<td>1PVN</td>
<td>4</td>
<td>No</td>
<td>No</td>
<td>0.03</td>
</tr>
<tr>
<td>1B0C</td>
<td>5</td>
<td>No</td>
<td>No</td>
<td>0.04</td>
</tr>
<tr>
<td>1B2S</td>
<td>5</td>
<td>No</td>
<td>No</td>
<td>0.02</td>
</tr>
<tr>
<td>1C4Q</td>
<td>5</td>
<td>No</td>
<td>No</td>
<td>0.11</td>
</tr>
<tr>
<td>1EQ2</td>
<td>5</td>
<td>No</td>
<td>No</td>
<td>0.15</td>
</tr>
<tr>
<td>1D2N</td>
<td>6</td>
<td>No</td>
<td>No</td>
<td>0.00</td>
</tr>
<tr>
<td>1AON</td>
<td>7</td>
<td>No</td>
<td>No</td>
<td>0.14</td>
</tr>
<tr>
<td>1HX5</td>
<td>7</td>
<td>No</td>
<td>No</td>
<td>0.13</td>
</tr>
<tr>
<td>1TZO</td>
<td>7</td>
<td>No</td>
<td>No</td>
<td>0.00</td>
</tr>
<tr>
<td>1FXZ</td>
<td>3</td>
<td>No</td>
<td>Yes</td>
<td>0.28</td>
</tr>
<tr>
<td>1P32</td>
<td>3</td>
<td>No</td>
<td>Yes</td>
<td>0.63</td>
</tr>
<tr>
<td>1MIF</td>
<td>3</td>
<td>No</td>
<td>Yes</td>
<td>0.34</td>
</tr>
<tr>
<td>2PHL</td>
<td>3</td>
<td>No</td>
<td>Yes</td>
<td>0.35</td>
</tr>
<tr>
<td>1PQH</td>
<td>4</td>
<td>No</td>
<td>Yes</td>
<td>0.23</td>
</tr>
<tr>
<td>1DJR</td>
<td>5</td>
<td>No</td>
<td>Yes</td>
<td>0.35</td>
</tr>
<tr>
<td>1SAC</td>
<td>5</td>
<td>No</td>
<td>Yes</td>
<td>0.16</td>
</tr>
<tr>
<td>1PV5</td>
<td>5</td>
<td>No</td>
<td>Yes</td>
<td>0.29</td>
</tr>
<tr>
<td>1Q85</td>
<td>5</td>
<td>No</td>
<td>Yes</td>
<td>0.26</td>
</tr>
<tr>
<td>1XTF</td>
<td>5</td>
<td>No</td>
<td>Yes</td>
<td>0.58</td>
</tr>
<tr>
<td>1GEO</td>
<td>6</td>
<td>No</td>
<td>Yes</td>
<td>0.15</td>
</tr>
<tr>
<td>1RRE</td>
<td>6</td>
<td>No</td>
<td>Yes</td>
<td>0.87</td>
</tr>
<tr>
<td>1AVO</td>
<td>7</td>
<td>No</td>
<td>Yes</td>
<td>0.13</td>
</tr>
<tr>
<td>7AHL</td>
<td>7</td>
<td>No</td>
<td>Yes</td>
<td>0.22</td>
</tr>
<tr>
<td>1H64</td>
<td>7</td>
<td>No</td>
<td>Yes</td>
<td>0.28</td>
</tr>
<tr>
<td>1NY6</td>
<td>7</td>
<td>No</td>
<td>Yes</td>
<td>0.99</td>
</tr>
<tr>
<td>2WQH</td>
<td>2</td>
<td>Yes</td>
<td>No</td>
<td>0.00</td>
</tr>
<tr>
<td>1AF6</td>
<td>3</td>
<td>Yes</td>
<td>No</td>
<td>0.03</td>
</tr>
<tr>
<td>1BVP</td>
<td>3</td>
<td>Yes</td>
<td>No</td>
<td>0.00</td>
</tr>
<tr>
<td>PDB</td>
<td>Symmetry Order</td>
<td>Self-Interaction</td>
<td>Interface Difference</td>
<td>Interface Cα RMSD</td>
</tr>
<tr>
<td>---------</td>
<td>----------------</td>
<td>------------------</td>
<td>----------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>1G2O</td>
<td>3</td>
<td>Yes</td>
<td>No</td>
<td>0.11</td>
</tr>
<tr>
<td>1KHT</td>
<td>3</td>
<td>Yes</td>
<td>No</td>
<td>1.13</td>
</tr>
<tr>
<td>1MPM</td>
<td>3</td>
<td>Yes</td>
<td>No</td>
<td>0.00</td>
</tr>
<tr>
<td>1QY7</td>
<td>3</td>
<td>Yes</td>
<td>No</td>
<td>0.01</td>
</tr>
<tr>
<td>1E2Y</td>
<td>5</td>
<td>Yes</td>
<td>No</td>
<td>0.10</td>
</tr>
<tr>
<td>3MGE</td>
<td>6</td>
<td>Yes</td>
<td>No</td>
<td>0.00</td>
</tr>
<tr>
<td>1TR0</td>
<td>2</td>
<td>Yes</td>
<td>No</td>
<td>0.08</td>
</tr>
<tr>
<td>1VIM</td>
<td>2</td>
<td>Yes</td>
<td>No</td>
<td>0.26</td>
</tr>
<tr>
<td>1OUS</td>
<td>2</td>
<td>Yes</td>
<td>Yes</td>
<td>0.60</td>
</tr>
<tr>
<td>3D57</td>
<td>2</td>
<td>Yes</td>
<td>Yes</td>
<td>0.18</td>
</tr>
<tr>
<td>1EUA</td>
<td>3</td>
<td>Yes</td>
<td>Yes</td>
<td>0.25</td>
</tr>
<tr>
<td>1DBF</td>
<td>3</td>
<td>Yes</td>
<td>Yes</td>
<td>0.25</td>
</tr>
<tr>
<td>1URZ</td>
<td>3</td>
<td>Yes</td>
<td>Yes</td>
<td>0.47</td>
</tr>
<tr>
<td>1HB7</td>
<td>3</td>
<td>Yes</td>
<td>Yes</td>
<td>0.13</td>
</tr>
<tr>
<td>1JS0</td>
<td>3</td>
<td>Yes</td>
<td>Yes</td>
<td>0.32</td>
</tr>
<tr>
<td>1KRR</td>
<td>3</td>
<td>Yes</td>
<td>Yes</td>
<td>0.46</td>
</tr>
<tr>
<td>1KKE</td>
<td>3</td>
<td>Yes</td>
<td>Yes</td>
<td>3.20</td>
</tr>
<tr>
<td>1O91</td>
<td>3</td>
<td>Yes</td>
<td>Yes</td>
<td>0.22</td>
</tr>
<tr>
<td>1V6H</td>
<td>3</td>
<td>Yes</td>
<td>Yes</td>
<td>0.47</td>
</tr>
<tr>
<td>2EBO</td>
<td>3</td>
<td>Yes</td>
<td>Yes</td>
<td>0.54</td>
</tr>
<tr>
<td>1QSH</td>
<td>3</td>
<td>Yes</td>
<td>Yes</td>
<td>1.48</td>
</tr>
<tr>
<td>1F23</td>
<td>3</td>
<td>Yes</td>
<td>Yes</td>
<td>0.39</td>
</tr>
<tr>
<td>1TNF</td>
<td>3</td>
<td>Yes</td>
<td>Yes</td>
<td>0.83</td>
</tr>
<tr>
<td>1KBJ</td>
<td>4</td>
<td>Yes</td>
<td>Yes</td>
<td>0.09</td>
</tr>
<tr>
<td>1P7B</td>
<td>4</td>
<td>Yes</td>
<td>Yes</td>
<td>0.44</td>
</tr>
<tr>
<td>1NC7</td>
<td>4</td>
<td>Yes</td>
<td>Yes</td>
<td>0.52</td>
</tr>
<tr>
<td>1NSC</td>
<td>4</td>
<td>Yes</td>
<td>Yes</td>
<td>0.11</td>
</tr>
<tr>
<td>1KZ1</td>
<td>5</td>
<td>Yes</td>
<td>Yes</td>
<td>0.11</td>
</tr>
<tr>
<td>3CHB</td>
<td>5</td>
<td>Yes</td>
<td>Yes</td>
<td>0.10</td>
</tr>
<tr>
<td>1NLQ</td>
<td>5</td>
<td>Yes</td>
<td>Yes</td>
<td>0.36</td>
</tr>
<tr>
<td>1JG5</td>
<td>5</td>
<td>Yes</td>
<td>Yes</td>
<td>0.26</td>
</tr>
<tr>
<td>1FB1</td>
<td>5</td>
<td>Yes</td>
<td>Yes</td>
<td>0.43</td>
</tr>
<tr>
<td>1JZN</td>
<td>5</td>
<td>Yes</td>
<td>Yes</td>
<td>0.31</td>
</tr>
<tr>
<td>1LTA</td>
<td>5</td>
<td>Yes</td>
<td>Yes</td>
<td>0.23</td>
</tr>
</tbody>
</table>

Table 4.3. Analysis of backbone and side-chain conformations of 67 symmetric complexes. For each PDB of symmetric complex the table shows (1) The symmetry order. (2) Whether there is an amino acid that interacts with itself in the adjacent monomer. (3) Whether there are amino-acids in the interface which have different conformations in two adjacent monomers. Two side-chain conformations were considered to be different if the atomic RMSD exceeds 1Å, after superimposing the two residues according to their backbone atoms. (4) The interface Cα RMSD (IRMSD) between two arbitrary monomers.
4.3.2 The importance of symmetry constraints in docking refinement of symmetric multimers

Protein complexes with $C_n$ symmetry have strict constraints on the rigid-body transformation between adjacent monomers. The transformation must be around a symmetry axis and the rotation angle must be exactly $360°/n$. This constraint restricts the search space of the refinement algorithm and therefore increases the chance of finding a highly accurate solution. In order to demonstrate the importance of using these constraints we used SymmDock to predict the hexameric capsomer of HIV-1 (PDB ID: 3MGE) given a monomer in the bound conformation. Then, we refined the first acceptable solution of SymmDock (which was ranked in the 4th place) by SymmRef and by FiberDock. FiberDock [179] is a general docking refinement method that uses the same minimization techniques as SymmRef but with no symmetry constraints. The original SymmDock solution was highly accurate, with RMSD of 1.70Å and IRMSD of 0.69Å. SymmRef improved the accuracy of the docking solution to RMSD of 0.21Å and IRMSD of 0.09Å. FiberDock refinement resulted in a less accurate solution with RMSD of 2.89Å and IRMSD of 1.03Å.

Both refinement methods minimize the rigid-body transformation between two adjacent monomers. The refined transformation can be used for building the full hexameric complex by applying it five times on the structure of the first monomer. Since FiberDock does not restrict the final transformation to be symmetric, using it to build the complete complex often results in an open complex, where the last monomer does not interact with the first monomer, or in a severe clash between the last monomer and the first one. In the case of the hexameric capsomer of HIV-1 the refinement by FiberDock resulted in an open complex (Figure 4.4).

Next, we used FiberDock to refine the top 1000 docking solutions of SymmDock, given the bound structure of one of the monomers (chosen arbitrarily) for each case in our dataset. In cases where SymmDock generated less than 1000 solutions, we refined all of them. The results show that FiberDock improved SymmDock results in only half of the cases (see Table 4.4 in the supplementary material). SymmRef however, improved both the ranking and the accuracy of SymmDock results in 12 out of the 16 cases, and overall produce significantly better results than FiberDock. SymmRef results for bound docking are described in details in the next section and in Table 4.5.
The results of refining SymmDock solutions by FiberDock, with no symmetry constraints.

### 4.3.3 Bound Docking Experiments

In order to test the performance of SymmRef we first used it to refine the top 1000 docking solutions of SymmDock [88], given the bound structure of one of the monomers (chosen arbitrarily). In cases where SymmDock generated less than 1000 solutions, we refined all of them. In this experiment we used SymmRef without the backbone refinement procedure (step 1.3 in the algorithm described in the Methods section), and then compared the performance of the algorithm with and without the restriction of identical side-chain conformations assigned to each monomer.

The results are presented in Table 4.5. In 12 out of the 16 cases, SymmRef significantly improved both the ranking and the accuracy of the SymmDock results, and ranked a medium or a high quality model in the 1st or 2nd place.

In many of the cases, the first acceptable solution of SymmRef originated from a SymmDock solution which was poorly ranked, even though near native results existed in the top 10 results of SymmDock. For example, in the case of 3MGE the first acceptable solution of SymmDock was ranked in the 4th place with RMSD of 1.70Å and IRMSD of 0.69Å. Refinement and re-ranking of this solution resulted in a high accuracy solution with RMSD of 0.21Å and IRMSD of 0.09Å, which was still ranked in the 4th place by SymmRef. The top solution by SymmRef had a similar accuracy but a slightly better energy and it was created by refining a less accurate SymmDock solution, with RMSD of 8.84Å which was ranked 691. This and other results demonstrate that during the refinement by SymmRef, solutions with RMSD in the range of 0Å – 10Å converge to a highly accurate solution with low energy score. Only in rare cases in which the initial model
before the refinement is relatively inaccurate, the accuracy of the refined model might get slightly worse (e.g. 1B0C, 1BK6, 1D01).

In the cases of 1B0C and 1DD1, the top near native results were ranked 188 and 597 (respectively) by SymmRef, which is a poor ranking for bound docking experiments. In these cases the size of the interface of the native structure are very small, with ΔASA (Accessible Surface Area) of 238.4Å² for 1B0C and 601Å² for 1D01 as calculated by the PROTORP server [186]. The standard average interface surface [187] is approximately 1000Å² and complexes with ΔASA < 1400Å² are considered to be very difficult for docking [188]. Additionally, in solution, the 1B0C complex actually forms a decamer (dimer of pentamers) [189]. The interaction between two pentamers stabilizes the binding of the monomers within a pentamer. We believe that this could be the reason for the relatively poor ranking and high binding energy of the first acceptable model that SymmRef predicted for this pentamer. SymmDock ranked in the 7th place an acceptable solution for the 1B0C case, according to the CAPRI criteria. However, detailed examination of this solution showed that the solution is quite far from the native structure, with RMSD of 15.13Å. This solution was considered to be acceptable due to the IRMSD which is slightly below 4Å. However, since the interface in this case is very small, the IRMSD by itself is not a sufficient measure for the quality of the model. SymmRef ranked a more accurate solution with RMSD of 11.61Å in the 10th place but this solution is not considered to be acceptable according to CAPRI criteria and thus it does not appear in Table 4.5.

**Evaluation of the use of side-chains symmetry constraints.**

A comparison of the performance of SymmRef with and without symmetry constraints on the side-chain conformations shows that in specific cases these constraints prevent the method from creating high accuracy results with low energy score. This is shown most significantly in the case of 1URZ in which a comparison between monomers in the crystal structure of the complex reveals interface side-chains with different conformations. In this case, SymmRef without symmetric side-chains restriction ranked 4 high accuracy and two medium accuracy results in the top 10 solutions, while the refinement with this symmetric restriction didn’t generate a high accuracy result at any rank (among all the 1000 results), and only one medium accuracy model was generated, but it was ranked in place 771. In addition to the 1URZ case, an improvement in the results of SymmRef without restricting the side-chain conformations can also be seen in cases 3D57, 1A3F, 3MGE and 1AP9. However, the restriction of the side-chain conformation slightly improved the results in cases 2WQH and 1BK6.

**Evaluation of the energy scoring function adjustments.**

In this study we slightly adjusted the weights of the electrostatics terms of the energy scoring function that was used in our previously developed FireDock and FiberDock methods, as described in Methods section. These values were manually adjusted by a process of trial and error on a subset of our dataset which includes the first 11 cases in Table 4.2 (marked with starts). For these 11 cases the adjustment of the weights improved the ranking in 2 cases (1D2N
and 1A3F). In the rest of the cases the ranking have hardly changed (see Table 4.6). We also noticed that there was a greater improvement in the ranking when the constraint of symmetric side-chain conformation was enforced. As shown before, this constraint causes SymmRef to generate less accurate models. For these models the adjusted scoring function was able to drastically improve the ranking, compared to the original scoring function. The adjusted scoring function reduces the influence of the repulsive electrostatic forces. In symmetric complexes two residues with the same charge often interact with each other. Therefore, inaccurate modeling of their conformation may cause a high calculated repulsive electrostatics energy. Reducing the weights of the repulsive electrostatics terms improves the robustness of the scoring function, which is especially important for unbound docking cases.

In order to rule out over fitting to the data we examined the affect of this adjustment on the other 5 cases in our dataset and on additional 10 independent cases of symmetric complexes. For all these cases we ran SymmDock, using the bound conformation of one of the monomers, and then we refined and re-ranked the solutions by the original and the adjusted energy scoring function. The results are detailed in Table 4.7. In 3 out of the 21 test cases (1A3F, 1EUA and 1RRE) the ranking has improved when the adjusted scoring function was used, and in the other cases the ranking remained almost identical.
<table>
<thead>
<tr>
<th>PDB</th>
<th>SymmDock</th>
<th>SymmRef</th>
<th>SymmRef with symmetric side-chain optimization</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>First Acceptable(a)</td>
<td>Top 10 quality <em><strong>/</strong>/</em></td>
<td>First acceptable(a)</td>
</tr>
<tr>
<td>1URZ (C3)</td>
<td>1 (3.46,1.77,0.73)</td>
<td>0/3/0</td>
<td>1 (2.03,1.09,0.73)</td>
</tr>
<tr>
<td>1VIM (D2)</td>
<td>1 (4.97,1.88,0.56)</td>
<td>1/1/4</td>
<td>1 (0.79,0.45,0.90)</td>
</tr>
<tr>
<td>1F23 (C3)</td>
<td>1 (4.08,1.98,0.81)</td>
<td>0/3/0</td>
<td>1 (1.96,1.13,0.80)</td>
</tr>
<tr>
<td>3D57 (C2)</td>
<td>2 (7.49,2.87,0.59)</td>
<td>0/0/2</td>
<td>1 (3.01,0.92,0.88)</td>
</tr>
<tr>
<td>1TR0 (D6)</td>
<td>1 (1.86,0.71,0.95)</td>
<td>3/4/2</td>
<td>1 (0.58,0.28,0.91)</td>
</tr>
<tr>
<td>1D2N (C6)</td>
<td>1 (1.78,0.55,0.91)</td>
<td>1/3/0</td>
<td>1 (0.26,0.11,0.98)</td>
</tr>
<tr>
<td>1JS0 (C3)</td>
<td>9 (4.83,1.72,0.72)</td>
<td>0/1/0</td>
<td>1 (0.68,0.33,1.00)</td>
</tr>
<tr>
<td>1B0C (C5)</td>
<td>7 (15.13,3.70,0.22)</td>
<td>0/0/1</td>
<td>188 (14.42,3.43,0.22)</td>
</tr>
<tr>
<td>1A3F (C3)</td>
<td>9 (2.24,0.10,0.36)</td>
<td>0/0/1</td>
<td>1 (0.47,0.19,0.92)</td>
</tr>
<tr>
<td>3MGE (C6)</td>
<td>4 (1.70,0.69,0.88)</td>
<td>3/1/1</td>
<td>1 (0.89,0.27,0.89)</td>
</tr>
<tr>
<td>2WQH (C2)</td>
<td>105 (7.6,1.9,0.91)</td>
<td>0/0/0</td>
<td>1 (0.90,0.34,1.00)</td>
</tr>
<tr>
<td>1AP9 (C3)</td>
<td>1 (9.00,4.31,0.27)</td>
<td>0/0/3</td>
<td>3 (8.74,4.21,0.20)</td>
</tr>
<tr>
<td>1BK6 (C2)</td>
<td>13 (3.58,1.71,0.76)</td>
<td>0/0/0</td>
<td>6 (13.65,3.87,0.41)</td>
</tr>
<tr>
<td>1D01 (C3)</td>
<td>76 (9.38,4.20,0.43)</td>
<td>0/0/0</td>
<td>597 (9.88,3.41,0.20)</td>
</tr>
<tr>
<td>1DD1 (C3)</td>
<td>57 (5.54,2.11,0.52)</td>
<td>0/0/0</td>
<td>1 (1.78,1.73,0.75)</td>
</tr>
<tr>
<td>1F1G (C2)</td>
<td>1 (8.56,2.29,0.70)</td>
<td>0/0/2</td>
<td>2 (0.51,0.21,1.00)</td>
</tr>
</tbody>
</table>

Table 4.5: SymmDock and SymmRef docking results, by using the bound structure of one of the monomers, chosen arbitrarily. The table compares the performance of SymmRef, with and without a symmetric restriction on the interface side-chain conformations.

(a) The first acceptable solution (according to CAPRI criteria). The details of the solution are presented in the following format: rank (RMSD, IRMSD, Fnat).
(b) The number of high accuracy (***) medium accuracy (**) and acceptable (*) solutions in the top 10 solutions.
(c) The original SymmDock solution of the first acceptable solution of SymmRef.
<table>
<thead>
<tr>
<th>PDB</th>
<th>Original scoring function</th>
<th>Adjusted scoring function</th>
<th>SymmRef with symmetric side-chain optimization</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Top 10 quality&lt;sup&gt;(b)&lt;/sup&gt;</td>
<td>Top 10 quality&lt;sup&gt;(b)&lt;/sup&gt;</td>
<td>Top 10 quality&lt;sup&gt;(b)&lt;/sup&gt;</td>
</tr>
<tr>
<td>1URZ (C3)</td>
<td>4/1/4</td>
<td>4/2/3</td>
<td>0/0/0</td>
</tr>
<tr>
<td>1VIM (D2)</td>
<td>10/0/0</td>
<td>10/0/0</td>
<td>10/0/0</td>
</tr>
<tr>
<td>1F23 (C3)</td>
<td>0/10/0</td>
<td>0/10/0</td>
<td>0/10/0</td>
</tr>
<tr>
<td>3DS7 (C2)</td>
<td>5/1/0</td>
<td>5/1/0</td>
<td>3/3/0</td>
</tr>
<tr>
<td>1TR0 (D6)</td>
<td>10/0/0</td>
<td>10/0/0</td>
<td>10/0/0</td>
</tr>
<tr>
<td>1D2N (C6)</td>
<td>9/0/0</td>
<td>10/0/0</td>
<td>10/0/0</td>
</tr>
<tr>
<td>1JS0 (C3)</td>
<td>3/0/0</td>
<td>3/0/0</td>
<td>3/0/0</td>
</tr>
<tr>
<td>1BOC (C5)</td>
<td>0/0/0</td>
<td>0/0/0</td>
<td>0/0/0</td>
</tr>
<tr>
<td>1A3F (C3)</td>
<td>6/0/0</td>
<td>8/0/0</td>
<td>6/0/0</td>
</tr>
<tr>
<td>3MGE (C6)</td>
<td>10/0/0</td>
<td>10/0/0</td>
<td>0/0/0</td>
</tr>
<tr>
<td>2WQH (C2)</td>
<td>2/0/0</td>
<td>2/0/0</td>
<td>2/0/1</td>
</tr>
</tbody>
</table>

Table 4.6. Comparison of SymmRef ranking with the original vs. the adjusted energy scoring function on the training set of 11 symmetric docking cases. The cases with different ranking results are marked by bold font.

<table>
<thead>
<tr>
<th>PDB</th>
<th>Original scoring function</th>
<th>Adjusted scoring function</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Top 10 quality&lt;sup&gt;(b)&lt;/sup&gt;</td>
<td>Top 10 quality&lt;sup&gt;(b)&lt;/sup&gt;</td>
</tr>
<tr>
<td>1AP9 (C3)</td>
<td>0/0/1</td>
<td>0/0/1</td>
</tr>
<tr>
<td>1BK6 (C2)</td>
<td>0/0/0</td>
<td>0/0/1</td>
</tr>
<tr>
<td>1D01 (C3)</td>
<td>0/0/0</td>
<td>0/0/0</td>
</tr>
<tr>
<td>1DD1 (C3)</td>
<td>0/2/0</td>
<td>0/2/0</td>
</tr>
<tr>
<td>1F1G (C2)</td>
<td>3/0/0</td>
<td>3/0/0</td>
</tr>
<tr>
<td>1EK9 (C3)</td>
<td>10/0/0</td>
<td>10/0/0</td>
</tr>
<tr>
<td>1FXZ (C3)</td>
<td>2/0/0</td>
<td>2/0/0</td>
</tr>
<tr>
<td>1AF6 (C3)</td>
<td>7/0/0</td>
<td>7/0/0</td>
</tr>
<tr>
<td>1EUA (C3)</td>
<td>5/0/0</td>
<td>6/0/0</td>
</tr>
<tr>
<td>1G6O (C6)</td>
<td>9/1/0</td>
<td>9/1/0</td>
</tr>
<tr>
<td>1BV (C3)</td>
<td>10/0/0</td>
<td>10/0/0</td>
</tr>
<tr>
<td>1DBF (C3)</td>
<td>0/10/0</td>
<td>0/10/0</td>
</tr>
<tr>
<td>1RRE (C6)</td>
<td>7/0/0</td>
<td>8/0/0</td>
</tr>
<tr>
<td>1G2O (C3)</td>
<td>10/0/0</td>
<td>10/0/0</td>
</tr>
<tr>
<td>1HB7 (C3)</td>
<td>10/0/0</td>
<td>9/1/0</td>
</tr>
</tbody>
</table>

Table 4.7. Comparison of SymmRef ranking with the original vs. the adjusted energy scoring function on a test set of 15 symmetric docking cases. The cases with different ranking results are marked by bold font.
4.3.4 Unbound Docking Experiments

In an unbound experiment, we docked the unbound monomers by SymmDock and refined the top 1000 results by SymmRef. The results (Table 4.8) show the drastic improvement in the accuracy and ranking of the docking solutions. SymmDock ranked a near native solution in the top 10 solutions in 4 out of the 16 cases. The refinement and rescoring by SymmRef increased this number to 7, and in 4 of these cases the near native solution was ranked in the 1st place. In the case of 2WQH, for example, the first acceptable solution of SymmDock was ranked 275. After running SymmRef, a high accuracy model (according to CAPRI criteria) was ranked in the 1st place, three other high accuracy solutions were also ranked in the top 10 solutions. Figure 4.5 shows the structures of some of SymmDock solutions, before and after SymmRef refinement.

![SymmDock solution and Refinement by SymmRef](image)

**Figure 4.5.** Comparison between SymmDock solutions before and after refinement by SymmRef. The figure shows superimposition of the docking models on the native complex structure, presented in gray. The superimposition minimizes the RMSD of one of the monomers of the model and the native complex. The structure of the second monomer is shown in red for the SymmDock solution and in blue for the refined solution by SymmRef. (A) The docking solutions of 1VIM. On the left a SymmDock solution with RMSD of 8.97Å which was ranked 801. On the right, the refinement of this solution by SymmRef, with RMSD of 4.17Å which was ranked in the 1st place. (B) The docking solution of 2WQH. SymmDock solution with RMSD of 7.28Å and rank 455. SymmRef refined model with RMSD of 2.32Å and rank 1. (C) The docking solution of 1TR0. SymmDock solution with RMSD of 4.05Å and rank 96. SymmRef refined model with RMSD of 1.67Å and rank 1. The figure was taken from Mashiach-Farkash et al. 2011.
### Table 4.8. SymmDock, SymmRef and RosettaDock docking results, by using the unbound structure of a monomer.

(a) The first acceptable solution (according to CAPRI criteria). The details of the solution are presented in the following format: rank (RMSD, IRMSD, Fnat). Cases with no acceptable solutions are denoted by 'X'.

(b) The number of high accuracy (***) medium accuracy (**) and acceptable (*) solutions in the top 10 solutions.

(c) The original SymmDock solution of the first acceptable solution of SymmRef/RosettaDock.

<table>
<thead>
<tr>
<th>PDB</th>
<th>SymmDock First Acceptable(a)</th>
<th>SymmDock Top 10 quality(b)</th>
<th>SymmRef First acceptable(a)</th>
<th>SymmRef Original SymmDock solution(c)</th>
<th>RosettaDock First acceptable(a)</th>
<th>RosettaDock Original SymmDock solution(c)</th>
<th>RosettaDock Top 10 quality(b)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1URZ (C3)</td>
<td>430(8.61,6.09,0.2) 0/0/0</td>
<td>8 (7.52,6.87,0.24) 0/0/0</td>
<td>622 (6.21,5.74,0.29) 0/0/1</td>
<td>136 (6.34,6.07,0.21) 0/0/0</td>
<td>622 (6.21,5.74,0.29) 0/0/0</td>
<td>0/0/0</td>
<td></td>
</tr>
<tr>
<td>1VIM (D2)</td>
<td>2(4.45, 3.31, 0.15) 0/0/2</td>
<td>1 (4.17,3.80,0.40) 0/0/0</td>
<td>801 (8.97,4.57,0.11) 0/0/1</td>
<td>186 (3.22,3.42,0.46) 0/0/0</td>
<td>480 (29.03,14.96,0.00) 0/0/0</td>
<td>0/0/0</td>
<td></td>
</tr>
<tr>
<td>1F23 (C3)</td>
<td>1 (1.54,0.95,0.76) 1/6/0</td>
<td>1 (2.49,1.57,0.72) 0/0/0</td>
<td>37 (3.17,1.51,0.59) 0/0/0</td>
<td>1 (1.90,1.06,0.79) 0/0/0</td>
<td>3 (2.20,1.25,0.82) 0/0/0</td>
<td>4/6/0</td>
<td></td>
</tr>
<tr>
<td>3DS7 (C2)</td>
<td>44(10.36,3.99,0.53) 0/0/0</td>
<td>18 (2.30,1.24,0.74) 0/0/0</td>
<td>81 (7.92,3.34,0.41) 0/0/0</td>
<td>1 (2.98,1.07,0.79) 0/0/0</td>
<td>690 (31.46,14.58,0.15) 0/0/0</td>
<td>0/4/1</td>
<td></td>
</tr>
<tr>
<td>1TRO (D6)</td>
<td>1 (2.68,1.53,0.66) 0/9/1</td>
<td>1 (1.67,1.08,0.75) 0/0/0</td>
<td>96(4.05,1.71,0.46) 0/0/0</td>
<td>1 (2.35,1.32,0.74) 0/0/0</td>
<td>328 (5.22,2.32,0.75) 0/0/0</td>
<td>0/10/0</td>
<td></td>
</tr>
<tr>
<td>1DN2 (C6)</td>
<td>X 0/0/0</td>
<td>X 0/0/0</td>
<td>X 0/0/0</td>
<td>X 0/0/0</td>
<td>X 0/0/0</td>
<td>0/0/0</td>
<td></td>
</tr>
<tr>
<td>1JS0 (C3)</td>
<td>387 (9.80,3.71,0.33) 0/0/0</td>
<td>276 (8.42,3.11,0.25) 0/0/0</td>
<td>940 (14.56,9.79,0.04) 0/0/0</td>
<td>93 (6.11,2.76,0.33) 0/0/0</td>
<td>489 (11.12,9.36,0.11) 0/0/0</td>
<td>0/0/0</td>
<td></td>
</tr>
<tr>
<td>1BOC (C5)</td>
<td>134(14.9,3.56,0.13) 0/0/0</td>
<td>189 (14.52,3.46,0.17) 0/0/0</td>
<td>648 (14.21,3.47,0.35) 0/0/0</td>
<td>17 (15.46,3.64,0.13) 0/0/0</td>
<td>512 (15.97,3.80,0.09) 0/0/0</td>
<td>0/0/0</td>
<td></td>
</tr>
<tr>
<td>1A3F (C3)</td>
<td>29(6.05,2.60,0.62) 0/0/0</td>
<td>57 (3.51,1.69,0.42) 0/0/0</td>
<td>249 (8.11,2.98,0.62) 0/0/0</td>
<td>5 (3.54,1.91,0.39) 0/0/0</td>
<td>616 (5.31,2.39,0.68) 0/0/0</td>
<td>0/2/2</td>
<td></td>
</tr>
<tr>
<td>3MGE (C6)</td>
<td>X 0/0/0</td>
<td>X 0/0/0</td>
<td>X 0/0/0</td>
<td>X 0/0/0</td>
<td>X 0/0/0</td>
<td>0/0/0</td>
<td></td>
</tr>
<tr>
<td>2WQH (C2)</td>
<td>275(12.34,1.90,0.69) 0/0/0</td>
<td>1 (2.32,0.37,0.66) 0/0/0</td>
<td>455 (7.28,2.41,0.44) 0/0/0</td>
<td>75 (17.15,3.76,0.31) 0/0/0</td>
<td>532 (18.26,7.09,0.22) 0/0/0</td>
<td>0/0/0</td>
<td></td>
</tr>
<tr>
<td>1AP9 (C3)</td>
<td>164 (4.48,1.95,0.44) 0/0/0</td>
<td>82 (2.17,1.45,0.47) 0/0/0</td>
<td>164 (4.48,1.95,0.44) 0/0/0</td>
<td>25 (2.32,1.48,0.44) 0/0/0</td>
<td>164 (4.48,1.95,0.44) 0/0/0</td>
<td>0/0/0</td>
<td></td>
</tr>
<tr>
<td>1BKE6 (C2)</td>
<td>6 (10.40,3.28,0.71) 0/0/1</td>
<td>2 (6.82,3.15,0.25) 0/0/0</td>
<td>6 (10.40,3.28,0.71) 0/0/0</td>
<td>1 (2.87,3.15,0.25) 0/0/0</td>
<td>X 0/0/0</td>
<td>0/0/0</td>
<td></td>
</tr>
<tr>
<td>1D01 (C3)</td>
<td>65 (12.36,3.98,0.16) 0/0/0</td>
<td>22 (6.44,3.20,0.16) 0/0/0</td>
<td>438 (6.44,3.20,0.16) 0/0/0</td>
<td>32 (20.57,3.76,0.20) 0/0/0</td>
<td>156 (21.92,4.27,0.13) 0/0/0</td>
<td>0/0/0</td>
<td></td>
</tr>
<tr>
<td>1DD1 (C3)</td>
<td>213 (9.63,4.54,0.21) 0/0/0</td>
<td>439 (7.39,3.85,0.21) 0/0/0</td>
<td>242 (8.40,4.17,0.17) 0/0/0</td>
<td>183 (6.10,3.33,0.26) 0/0/0</td>
<td>521 (6.91,3.65,0.23) 0/0/0</td>
<td>0/0/0</td>
<td></td>
</tr>
<tr>
<td>1F1G (C2)</td>
<td>23 (14.12,3.54,0.40) 0/0/0</td>
<td>3 (8.20,3.72,0.30) 0/0/0</td>
<td>36 (9.42,3.12,0.56) 0/0/0</td>
<td>X 0/0/0</td>
<td>X 0/0/0</td>
<td>0/0/0</td>
<td></td>
</tr>
</tbody>
</table>
4.3.5 Comparison of SymmRef to the refinement and rescoring by RosettaDock

The RosettaDock method can be used for global and local docking, and also for refining and re-ranking of given symmetric docking solutions [176]. The method uses Monte-Carlo minimization to optimize the relative orientation and the conformation of the monomers. We used the symmetric docking protocol of RosettaDock for refining and re-ranking SymmDock unbound docking solutions and compared the results with SymmRef. The currently available download version of Rosetta does not allow backbone refinement during symmetric docking. Therefore, only side-chain flexibility was modeled by the RosettaDock refinement in this experiment. For each one of the top 1000 solutions of SymmDock, we performed 10 local refinements by the symmetric RosettaDock method and selected the lowest energy model. This is similar to the SymmRef method which also performs the refinement 10 times and chooses the solution with the best score (see the Methods section). Finally we ranked the 1000 refined solutions by the RosettaDock energy score and compared the results to the results of SymmRef (see Table 4.8).

Both refinement methods significantly improved SymmDock results. However, the refinement by SymmRef produced better results on the tested benchmark. In 5 cases (1URZ, 1VIM, 2WQH, 1BK6, 1F1G) SymmRef ranked a near native solution in the top 10 results, while RosettaDock failed to do so. 1URZ, 1VIM and 1BK6 are cases with relatively large backbone conformational change, with IRMSD of 4.97Å, 3.53Å, and 2.18Å between their bound and unbound structure. This can explain the superior results of SymmRef which models backbone movements. On the other hand, in two other cases (3D57, 1A3F) RosettaDock ranked near native solutions in the top 10 results, while SymmRef failed to do so.

Another important difference between the performances of the methods is the running times. The average running time of a single refinement by SymmRef is 23 seconds on a 2.33GHz Intel(R) Xeon(R) CPU with 16G of memory. The average running time of a single refinement by the symmetric RosettaDock is 6 minutes.

4.3.6 Comparison to other symmetric docking methods

Our full symmetric docking protocol uses the SymmDock rigid symmetric docking procedure for generating a set of 1000 models and then uses SymmRef to refine and re-score these models. The performance of this docking protocol on the dataset of unbound structures was promising. In 7 out of the 16 cases SymmRef produced a near native model in the top 10 solutions, in 4 of these cases the first near-native solution was ranked in the 1st place. In one other case a near native model was ranked in the top 20 solutions. We compared this protocol to other state-of-the-art methods of symmetric docking: HADDOCK [177] and M-ZDOCK [175]. The results are presented in Table 4.9.

The HADDOCK method requires information about the location of the interface to guide the docking. For that we used the CPORT web-server, which integrates the results of five interface
prediction methods [190,191,192,193,194]. The CPORT web-server was previously used [177] for predicting the interface of symmetric complexes prior to using HADDOCK and we used it in the same manner. However, one should note that HADDOCK is mostly used in cases where some biological information on the approximate location of the binding site is available. Since we wanted to compare between purely computational methods, we used the CPORT computational method for predicting the binding site. SymmDock can also use information on the interface location (not shown in the results). Given a set of residues that are suspected to be in the interface, SymmDock will only generate solutions in which at least one of these residues is located in the interface.

The HADDOCK webserver cannot predict hexamers, therefore we tested it on 14 out of the 16 cases in the dataset. Additionally, we analyzed only the best structure in the top 10 clusters (which are given as output by the webserver).

HADDOCK’s performance is comparable to our protocol. However, it greatly depends on the interface prediction. In 3 of the cases more than 50% of the residues predicted by CPORT were indeed in the interface. In all these cases HADDOCK predicted a near-native docking model in its top 10 clusters. In 11 cases, less than 50% of the residues predicted by CPORT were correct, and in only 4 of these cases HADDOCK predicted a near native docking model in the top 10 clusters.

The M-ZDOCK method produced a near-native model in the top 10 solutions in 4 out of the 16 cases. In 3 cases (1URZ, 1VIM and 2WQH) in which M-ZDOCK did not produce a near-native result in the top 50 solutions, our docking protocol ranked a near-native model in the top 10 solutions, in two of these cases in the 1st place. On the other hand, in the case of 1D2N, M-ZDOCK produced an acceptable solution which was ranked in the 2nd place, while SymmDock did not generate an acceptable solution in the top 1000 solutions, and therefore no near-native solutions were refined by SymmRef.

M-ZDOCK is a rigid-body symmetric docking method, which is based on the FFT approach. Similar to SymmDock it does not model any side-chain or backbone movements, but allows a certain amount of steric clashes (soft-docking). The scoring function of M-ZDOCK is based on surface complementarity, electrostatics and desolvation. However, since the structure of the monomers is not refined the values of the energy terms are less accurate. SymmRef models induced-fit conformational changes and calculates a more accurate energy score, and therefore achieved better results in most of the cases in our benchmark.

The SymmRef algorithm can refine rigid-docking solutions that are generated by any symmetric docking method. Therefore we tested its performance of refining and re-ranking the M-ZDOCK solutions. The results are presented in Table 4.10. The results show that SymmRef improves the accuracy and ranking of the solutions of M-ZDOCK. In 7 cases the number and quality of the near-native models in the top 10 solutions improved after SymmRef was used. In the case of 2WQH, for example, while M-ZDOCK did not rank any acceptable model in the top 100 solutions, SymmRef ranked 8 high accuracy models in the top 10 solutions.
Comparison of the performance of our symmetric docking protocol (SymmDock+SymmRef) to other state-of-the-art methods: M-ZDOCK and HADDOCK. The ‘X’ sign represents a failure in producing any near native solution. The ‘—’ sign represents cases on which HADDOCK could not be run on (hexamers).

(a) The first acceptable solution (according to CAPRI criteria). The details of the solution are presented in the following format: rank (RMSE, IRMSD, Fnat).

(b) The quality of the interface prediction of CPORT, which was given as an input to the HADDOCK method. This column shows the amount of True-Positive (TP) predictions and the amount of False-Positive (FP) predictions. The number in the brackets shows the percentage of correct interface residues in the predicted interface (precision).

### Table 4.9.

#### Comparison of M-ZDOCK and SymmRef docking results when using the unbound structure of a monomer.

(a) The first acceptable solution (according to CAPRI criteria). The details of the solution are presented in the following format: rank (RMSE, IRMSD, Fnat).

(b) The number of high accuracy (***) medium accuracy (**) and acceptable (*) solutions in the top 10 quality predictions.

(c) The original M-ZDOCK solution of the first acceptable symmetry.
4.4 Summary

The majority of protein complexes in the living cell are composed of several proteins that simultaneously interact with each other [195,196]. Most of them are symmetric multimers [169]. In this paper we described SymmRef, a novel method for symmetric docking refinement. The method refines and re-ranks symmetric docking solutions that can be generated by any global symmetric docking method. SymmRef models backbone and side-chain conformational changes and performs rigid-body minimization. The method is based on our previously developed protein-protein refinement method, FiberDock [179,197]. It finds the optimal rotamers for the interface residues by an Integer Linear Programming approach, and refines the backbone conformation by normal modes minimization. SymmRef is available for download at http://bioinfo3d.cs.tau.ac.il/SymmRef/download.html.

Previous symmetric docking methods restrict the monomers to be identical. However, most of the solved crystal structures of symmetric homo-oligomers show local differences in side-chain conformations. We analyzed this phenomenon and discovered that self-interaction of an amino-acid in a symmetric complex tends to break the symmetry at the level of side-chain conformations. Therefore, we concluded that the side-chain optimization procedure should not be restricted to select symmetric rotamers. In cases without self-interactions there is a better chance for the side-chain conformations to be identical in each monomer. However, in these cases the interface between two monomers contains different residues in each side. Therefore restricting the side-chain optimization procedure in these cases, to select symmetric rotamers, will not reduce the search space.

We tested the performance of SymmRef on unbound docking cases by refining the symmetric rigid-docking solutions of SymmDock [155,175] and M-ZDOCK [175]. The results show that in both cases SymmRef improves both the accuracy and the ranking of the symmetric rigid-docking solutions with RMSD in the range of 0Å – 10Å. Comparison of SymmRef with the refinement and re-scoring performance of RosettaDock showed that SymmRef produces better results on the tested benchmark. Moreover, the running time of SymmRef is much faster than the running time of RosettaDock (by a factor of 15) and therefore SymmRef is able to perform a significantly larger number of refinement iterations with the same computational resources. Alternatively, the great efficiency of SymmRef can enable the user to refine and re-score significantly more rigid-docking solutions.

We also compared the performance of M-ZDOCK and HADDOCK with our full symmetric docking protocol of refining and re-ranking the top 1000 solutions of SymmDock by SymmRef. Our experiments on the unbound cases dataset show that our protocol outperforms M-ZDOCK and is comparable to HADDOCK. Our docking protocol produced a near native model in the top three ranked solution in 6 out of the 16 cases. M-ZDOCK succeeded to rank a near native model in the top three ranked solution in 4 cases and HADDOCK in 5 cases. When comparing the number of
cases in which each docking method ranked a near native model in the top 10 solutions, our protocol succeeded in 7 cases, M-ZDOCK in 4 cases and HADDOCK in 7 cases.

Up until now, the docking community mainly concentrated on predicting the structure of binary protein-protein complexes. Studies have shown that the majority of protein complexes are composed of more than two proteins, and about half of all complexes contain more than five proteins [196]. Therefore, in order to model real-life complexes we have to be able to predict the structure of multi-molecular complexes. Predicting the structure of multi-molecular symmetric complexes is only the first step toward this direction [88,172,173,174,175,198]. Only a few methods were developed so far for predicting hetero-multi-molecular complexes [177,199,200,201]. In the future we plan to adapt our docking refinement technique for hetero-multi-molecular complexes. The refinement method could be used for improving the performance of current multi-molecular rigid-docking methods and for improving the accuracy of rigid fitting of protein structures into electron-microscopy density maps [202,203,204].
Chapter 5: Performance Evaluation of Our Full Docking Protocol

The CAPRI experiment (Critical Assessment of Predicted Interactions) simulates realistic and diverse docking challenges, each case having specific properties that may be exploited by docking algorithms [133,205,206,207]. Motivated by the different CAPRI challenges, we developed and implemented a comprehensive suite of docking algorithms. These were incorporated into a dynamic docking protocol, consisting of four main stages: (1) Biological and bioinformatics research aiming to predict the binding site residues, to define distance constraints between interface atoms and to analyze the flexibility of molecules; (2) Rigid or flexible docking, performed by the PatchDock or FlexDock method, which utilizes the information gathered in the previous step. Symmetric complexes are predicted by the SymmDock method; (3) Flexible refinement and re-ranking of the rigid docking solution candidates, performed by FiberDock; and finally, (4) clustering and filtering the results based on energy funnels. We analyzed the performance of our docking protocol on a large benchmark and on recent CAPRI targets. The analysis has demonstrated the importance of biological information gathering prior to docking, which significantly increased the docking success rate, and of the refinement and re-scoring stage which significantly improved the ranking of the rigid docking solutions. Our failures were mostly a result of mishandling backbone flexibility, inaccurate homology modeling, or incorrect biological assumptions. Most of the methods are available at http://bioinfo3d.cs.tau.ac.il/.

5.1 METHODS

Over the years our group has developed a set of efficient and practical docking algorithms. These methods were integrated into a comprehensive docking suite that can be used for predicting many types of molecular complexes with different properties (hinge motion, flexible loops, symmetric interactions, etc.) and restraints (biological information about the binding site location, distance constraints, etc.). Our docking protocol consists of four main stages, detailed below.

5.1.1 Biological and bioinformatics research of the interacting proteins

The goal of this preliminary stage is to define restraints that will reduce the search space of the docking. The methods we use in the following stage can receive as an input potential binding site residues and pair-wise atomic distance constraints. Reduction of the search space can be achieved by analysis of the biological function of the interaction. For example, if one of the proteins performs a modification in a certain site of the interacting molecule, then a distance
constraint can be defined between the active-site and the modified site. Information about the
binding site can be obtained from different sources, such as mutations that decrease the binding
affinity, sites that are known not to be in the interface (e.g. functional sites that are active
during the interaction) and bioinformatics conservation analysis [208] which predicts conserved
surface patches that often imply the binding site location. Multiple sequence and structure
alignment (e.g. BLAST [209], MultiProt [210]) often help in finding homologous proteins that
might form similar interactions.

Flexibility analysis determines the docking strategy. Hinges can be indentified using HingeProt
[211] and flexible loops can be recognized by B-factors, NMR and structural comparison of
different X-ray structures of the same or homologous proteins. Search space reduction
dramatically increases the docking success rate. However, if incorrect information is used, or the
results of the research are misinterpreted, the docking is highly likely to fail. Therefore, only high
confidence information should be used.

5.1.2 Rigid or hinge bent flexible docking

Rigid docking is performed by the PatchDock method [79,155], which is an efficient, geometry-
based technique. The method can explicitly reduce the search space based on the data collected
in the previous step, e.g. interface and non-interface residues and distance constraints. Hinge
movement is handled by the FlexDock algorithm [89] and if the target complex is a symmetric
multimer, the SymmDock method[88,155] is used.

Our rigid and flexible docking methods are very efficient and are probably the fastest techniques
available. Thousands of candidate complexes can be generated within minutes on a standard PC
computer. In order to deal with unbound real-life cases a small amount of steric clashes is
allowed at the interface. In order to rank the candidate complexes with energy based scoring
function, the interface is optimized and the clashes are removed by a refinement method.

5.1.3 Flexible refinement and re-ranking

In this stage the top 1000-5000 solutions from the rigid/flexible-docking stage are refined and
re-scored. Here we use the FiberDock [179,197] method which optimizes the side-chain
conformations in the interface, models backbone movements, minimizes the relative rigid-body
orientations of the molecules, and ranks the refined solutions by a binding-energy scoring
function. The FiberDock method is based on our previously developed FireDock method
[113,165] which also models side-chain movements, but keeps the backbone rigid. In the CAPRI
contest, we have used FireDock up to target 40, from which we started to use FiberDock in the
refinement stage. The refinement process results in a drastic improvement in the ranking and
accuracy of the predicted models (see the 5.2 RESULTS section).

For refining models of symmetric complexes, we developed a new refinement method called
SymmRef [198]. Similarly to FireDock [113], this method optimizes the side-chain conformations
in the interfaces and minimizes the rigid-body orientations of the symmetric units. However, unlike FireDock, here the refinement preserves the symmetry of the complex.

5.1.4 Clustering and filtering

In the final stage we try to identify the near native solutions from the top 50-100 refined models. First, we cluster the top ranked models and leave only the lowest energy model from each cluster as a representative. Then, we search for energy-score funnels around the remaining candidate docking solutions. Here we use our recently developed ValiDock server (http://bioinfo3d.cs.tau.ac.il/ValiDock/) which randomly samples rigid-body perturbations around each candidate solution, refines each perturbation by FiberDock and draws a graph of energy-score vs. RMSD from the candidate solution. Energy funnels are known to be a reliable indicator of near-native docking solutions [157,158], although in many cases energy funnels are also found in false docking solutions.

5.1.5 CAPRI participation

In the CAPRI challenge we participated as predictors, using three different docking protocols: (1) **PatchDock webserver** [155] (http://bioinfo3d.cs.tau.ac.il/PatchDock/) – rigid docking without human intervention. In cases of symmetric docking we used the SymmDock webserver [155] instead (http://bioinfo3d.cs.tau.ac.il/SymmDock/) (2) **FireDock** [165] /**FiberDock** [197] webserver (http://bioinfo3d.cs.tau.ac.il/FireDock/, http://bioinfo3d.cs.tau.ac.il/FiberDock/) – rigid docking by the PatchDock server followed by refinement of the top 1000 solutions by the FireDock/FiberDock server. (3) **Human prediction** – here we performed the full, four stage docking protocol described above. In addition, we participated in the scoring challenges [133], where we used FireDock/FiberDock to refine and re-rank the uploaded models.

5.2 RESULTS

In recent CAPRI rounds (13-19) we were among the best performing predictor teams by submitting near native models in 6 out of the 13 targets. In 4 targets, a near native model was also submitted by our automatic webserver (PatchDock, SymmDock, FireDock or FiberDock). In this section we briefly describe our successes and failures in these CAPRI rounds. For 6 of the targets, in which we failed, we identified failure causes, repeated the interactions prediction and in 3 of them we were able to obtain a near native model in the top 10 solutions (see Table 5.1). Additionally, we tested the performance of our docking protocol in "blind" docking of protein-protein docking benchmark cases with limited flexibility [212]. Our analysis demonstrates the significant contribution of the refinement and re-ranking stage in the docking protocol.
<table>
<thead>
<tr>
<th>Target</th>
<th>All groups</th>
<th>PatchDock/SymmDock&lt;sup&gt;a&lt;/sup&gt;</th>
<th>FireDock/FiberDock&lt;sup&gt;a&lt;/sup&gt;</th>
<th>Human Prediction&lt;sup&gt;b&lt;/sup&gt;</th>
<th>CAPRI retry&lt;sup&gt;c&lt;/sup&gt;</th>
<th>Scoring&lt;sup&gt;d&lt;/sup&gt;</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>***</td>
<td>**</td>
<td>*</td>
<td>***</td>
<td>**</td>
<td>*</td>
</tr>
<tr>
<td>T29</td>
<td>-</td>
<td>9</td>
<td>8</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>T30</td>
<td>-</td>
<td>-</td>
<td>2</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>T32</td>
<td>15</td>
<td>13</td>
<td>6</td>
<td>-</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>T33</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>T34</td>
<td>-</td>
<td>25</td>
<td>40</td>
<td>-</td>
<td>+1&lt;sup&gt;3&lt;/sup&gt;</td>
<td>X</td>
</tr>
<tr>
<td>T35</td>
<td>-</td>
<td>-</td>
<td>1</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>T36</td>
<td>-</td>
<td>-</td>
<td>1</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>T37</td>
<td>1</td>
<td>7</td>
<td>13</td>
<td>-</td>
<td>1</td>
<td>-</td>
</tr>
<tr>
<td>T38</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>T39</td>
<td>1</td>
<td>2</td>
<td>-</td>
<td>-</td>
<td>1</td>
<td>-</td>
</tr>
<tr>
<td>T40</td>
<td>79</td>
<td>54</td>
<td>31</td>
<td>1</td>
<td>3</td>
<td>5</td>
</tr>
<tr>
<td>T41</td>
<td>24</td>
<td>58</td>
<td>67</td>
<td>1</td>
<td>7</td>
<td>3</td>
</tr>
<tr>
<td>T42</td>
<td>9</td>
<td>5</td>
<td>6</td>
<td>1</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

Table 5.1. CAPRI achievements. The table shows the number of near native solutions found in each CAPRI target by all the participating groups, by our automatic webservers and by our group. The solutions are divided into three accuracy levels: high (***) *, medium (**), and acceptable (*). Cases without any solution are marked by '-' and cases in which we did not submit any models are marked by 'X'.

<sup>a</sup> The achievements of our webservers: PatchDock/SymmDock (SymmDock was used only in target 42) and FireDock/FiberDock (FiberDock was first used in target 40).

<sup>b</sup> The achievements of our group, where we used the full docking protocol, described in the Methods section.

<sup>c</sup> We analyzed six cases in which we failed and retried to dock the targets. In three of these targets we had a near native result in the top 10 solutions (see the Results section for details).

<sup>d</sup> Our achievements in the Scoring challenge of CAPRI.

<sup>†</sup> - Accurate results that were disqualified due to clashes or bad sequence identity.
5.2.1 Target 29: Trm8/Trm82 tRNA guanin-N(7)-methyltransferase

The docking challenge in target 29 was to predict the structure of Trm8 complexed with Trm82 tRNA guanin-N(7)-methyltransferase. Trm82 was given in the bound conformation and Trm8 in the unbound conformation. In this challenge we failed to predict the structure of the complex but succeeded in the scoring stage, where we identified 2 acceptable solutions by using FireDock. The given structure of Trm8 was truncated at the N-terminal (residue 73). In the rigid-docking stage of our protocol, we blocked the site of this residue, since we assumed that the missing N-terminal segment is occupying this area. This was a wrong assumption since in the native structure of the complex [213] residue 73 is interacting with Trm8. This false assumption prevented PatchDock from generating any near native solution. In addition, Trm8 has a flexible loop in the interface that could have been easily identified by the fact that part of it was missing in the X-ray structure. We repeated the docking experiment and ran PatchDock without blocking any site and after removing the flexible loop from the Trm8 structure. The first acceptable solution of PatchDock was ranked 2123 (RMSD of 4.88Å and IRMSD of 2.60Å). After refining and re-ranking the top 5000 PatchDock solutions by FiberDock the same solution was ranked in the 3rd place and its RMSD was improved to 4.33Å and IRMSD to 1.90Å.

5.2.2 Target 30: Rnd1-GTP bound to RBD dimer

In this target the challenge was to predict the structure of the complex of Rnd1-GTP and RBD, given their unbound structure. Based on the literature we correctly identified residues in the interface of both proteins [214,215,216]. However, in the CAPRI contest we used only the suspected binding site of Rnd1 and not of RBD and we did not get an acceptable solution in the top 10 models of FireDock. This was probably due to the flexible loop in the binding site of Rnd1 that decreased the shape complementarity of near native solutions. We repeated the experiment using the binding site information of both proteins and obtained an acceptable solution in the 5th place.

5.2.3 Target 32: Protease savinase bound to Bi-functional inhibitor BASI

In this challenge, the goal was to predict the structure of Protease savinase bound to its inhibitor given their unbound structures. In the literature we found a description of a homologous complex of proteinase K and its inhibitor (the coordinates of the complex were not available) [217]. Based on this description, we identified the interface of Protease savinase and the active loop of the inhibitor. This information was used by PatchDock and the results were refined and re-ranked by FireDock. The active loop of the inhibitor was suspected to be flexible. Therefore, we re-modeled it in the top solutions of FireDock, by ModLoop [218]. Then we refined these solutions again by FireDock. This process produced 2 high accuracy results, 2 medium results and 2 acceptable results in our 10 submitted models.
5.2.4 Targets 33-34: Rlma2 methyltransferase bound to its RNA substrate

Target 33 presented a challenge of modeling protein-RNA interaction. The unbound structure of Rlma2 methyltransferase protein was given, while for RNA molecule only homologous structure was available. The RNA molecule contains three hairpins numbered 33, 34 and 35. We predicted hinge-movements between the hairpins, and therefore tried to dock them separately. First, we docked hairpin 35 with a distance constraint between the active site of the methyltransferase and the methylated nucleotide (distance of up to 6Å). Then we docked hairpins 33-34 (together) to the top 10 results from the previous step. We were not able to produce a correct model in this target. An alignment between the modeled and the bound structure of the RNA molecule revealed a significant difference in the conformation of hairpin 35. This structural difference prevented PatchDock from generating a near native model in the first step of the docking process.

In target 34 we were given the bound structure of the RNA molecule. Here we used the same distance constraint and obtained 2 acceptable solutions in the top 10 models of the PatchDock server. However, they were disqualified due to low sequence identity that resulted from the fact that we docked the homologue protein of Rlma2 methyltransferase and not its model. As human predictors we suspected that the active-site of the protein can slightly open in the direction of the first normal mode. Therefore, we created an "open" form of the protein and used it in addition to the original "closed" form in cross-docking with PatchDock and FireDock and got 4 acceptable solutions in the top 10.

5.2.5 Targets 35-36: Xylanase Xyn10B

In target 35 the goal was to predict the structure of a covalently linked molecule with two domains: Polysaccharide binding module CBM22 and the catalytic module GH10. The two domains had to be modeled by homologues. In target 36 the bound structure of CBM22 was given. Like the vast majority of the predictors, we failed to predict the structure of the two-domain protein in both targets, due to inaccurate homology modeling.

5.2.6 Target 37: G-protein Arf6 bound to Leucine zipper of JIP4

In this challenge we had to model the structure of a Leucine Zipper domain and dock it to the G-protein Arf6. In the biological analysis stage of our docking protocol, we identified a conserved surface patch on one of the proteins (Arf6), by using ConSurf [208]. We used this patch as the location of the interface. In addition, we blocked the location of the GTP binding site in Arf6. These constraints were used in the rigid docking stage. After refining and rescoring the solutions by FireDock we were able to get an acceptable solution among the 10 submitted models. In the scoring challenge we also used FireDock and got plausible results: 1 medium accuracy and 2 acceptable solutions. The submitted medium accuracy model and one of the acceptable models were disqualified due to the number of clashes they had. These clashes could have been resolved by a second FireDock refinement.
5.2.7 Targets 38-39: Centaurin-\(\alpha_1\) bound to FHA domain of KIF13B

The goal in target 38 was to predict the structure of the complex of Centaurin-\(\alpha_1\) and the FHA domain of KIF13B, using the unbound structure of Centaurin-\(\alpha_1\) and a homology model of the FHA domain. In target 39 the bound structure of the FHA domain was given. In the docking of both targets we relied on published biological experiments [219] that showed that the FHA domain of KIF13B binds to the GAP domain of Centaurin-\(\alpha_1\). According to the published structure (PDB ID: 3FM8), this information was incorrect, and hence we failed in both targets. We retried to dock the two targets without any binding site information. We used PatchDock and refined and re-ranked the top 5000 results by FiberDock. In target 38, we did not have a near native solution in the top 5000 results of PatchDock due to a flexible loop in the interface of KIF13B. However, in target 39 PatchDock produced a near-native solution, with LRMSD of 3Å and IRMSD of 2.18Å, which was ranked 3059. FiberDock brought this model to the 6th place with LRMSD of 1.71Å and IRMSD of 1.01Å.

5.2.8 Target 40: A complex of Trypsin and protease inhibitor

In target 40 the goal was to predict the structure of Bovine Trypsin bound to the protease inhibitor. We were given the unbound structure of the Trypsin molecule and the bound structure of the inhibitor. Our FiberDock server obtained 1 high accuracy solution and 1 acceptable solution in the top 10 models. During this round, after submitting the webserver results, information on the location of the active sites of the inhibitor was published. The active sites were located on two different loops. We used this information when running PatchDock and FireDock, and validated the solutions by the existence of binding energy funnels. In addition, we found two known structures of Trypsin with peptide inhibitors (PDB ID: 1YF4 and 3BTG). Using these structures, we structurally aligned the active loops of our inhibitor with the peptides from these structures. The two approaches resulted in similar and high accuracy results. As human predictors we submitted 3 high accuracy and 3 acceptable models. In the scoring stage, we used our new FiberDock method and submitted 5 high accuracy, 2 medium accuracy and 1 acceptable models.

5.2.9 Target 41: Colicin E9 bound to Im2

In this challenge the goal was to predict the structure of Colicin E9 DNase domain in complex with the Im2 protein. Both proteins were given in their unbound structure. By running BLAST we indentified a high quality homologous complex of Colicin E9 DNase domain in complex with Im9 (PDB ID: 1BXI). The PDB file of Im2 contained 60 NMR structures which we structurally superimposed [210] on the Im9 structure in the homologue structure. We refined these 60 models by the FiberDock webserver. The first solution was of high accuracy and the remaining 9 were medium accuracy models. As human predictors we randomly sampled 100 perturbations, for each NMR structure, and refined these 6000 models by FiberDock. Now the top 10 submitted models contained 7 medium accuracy and 3 acceptable models. This case shows that increasing the number of solution candidates is not always beneficial, since the scoring function
is not accurate enough to reliably differentiate between high, medium and acceptable accuracy models.

5.2.10 Target 42: TPR repeat dimer

In this challenge we had to predict the dimer structure of a designed protein. We modeled the structure of the protein by a known homologous structure of TPR repeat (PDB ID: 1NA0). The sequence of the two proteins is identical, except for a single amino acid in the TPR repeat. An Aspartate in the homologue was changed to Tyrosine. For the webserver submission, we ran the SymmDock server [155] and refined the results by our new SymmRef method, in order to resolve the clashes in the solutions. The submitted models included 1 high accuracy result. As human predictor we suspected that the mutations might be in the interface and stabilize the interaction. Hence, we repeated the SymmDock and SymmRef runs with the information that the mutated Tyrosines are in the interface. This assumption was correct, and in our top 10 submitted models we had one high accuracy model and one acceptable model.

5.2.11 Blind docking experiment

We tested our automated docking protocol (docking by PatchDock and refinement and re-ranking by FiberDock) on the 88 "rigid-body" cases (with IRMSD<1.5Å) from the protein-protein benchmark3.0 [212]. The experiment showed the importance of the refinement and re-ranking stage of our docking protocol (see Figure 5.1). For each case we ran PatchDock and refined and re-ranked the top 1000 solutions by FiberDock, which modeled backbone flexibility in the receptor and side-chain movements in both proteins. PatchDock produced a near native solution (LRMSD < 10Å or IRMSD<4Å) among the top 1000 results in 71.6% of the cases. However, after dividing the cases into three categories, Antibody-Antigen (AA), Enzyme Inhibitor (EI) and Others (O) we noticed that PatchDock produced acceptable results in the top 1000 solutions in 95.2% of the AA cases, 85.2% of the EI cases, and 50% of the other cases. In only 10.2% of the 88 cases PatchDock ranked an acceptable solution in the top 10 results. However, after refining, re-scoring and re-ranking the solutions by FiberDock, in 28.4% of the cases an acceptable solution was present in the top 10 results. When the top 50 models are considered, FiberDock achieves an acceptable solution in around 70% of the AA and EI cases. The superior results of the AA and EI cases stems from the fact that PatchDock and FiberDock were optimized to handle these types of complexes, and further improvement should be done for general protein-protein complexes.
DISCUSSION

Recent CAPRI challenges have emphasized two key issues. The first is the importance of gathering information on the specific target prior to docking, which significantly increases the success rate of the docking. The second point is our limited ability to handle docking of unbound molecules with significant backbone flexibility in the interacting area or docking of homology models, which often have inaccurate backbone conformation.

In recent CAPRI rounds (13-19), we succeeded in our docking predictions in 6 out of 13 targets. In all of these cases we used biological information and bioinformatics analysis. For targets 32 and 40 we found biological information on the interface of both interacting proteins which drastically reduced the docking search space. In target 40, however, we also achieved a high accuracy result by our automatic FireDock server [165] without binding site information. In target 34 we used a distance constraint between the active-site of the methyltransferase and the methylated nucleotide in the RNA molecule. In target 37 we detected a conserved surface patch on one of the proteins and used it as the location of the interface. On the other hand, in

---

The fraction of cases with correct docking prediction

Figure 5.1. Blind docking experiment. We tested our "blind" docking protocol on the 88 benchmark cases with IRMSD < 1.5 Å. For each case we ran PatchDock and refined and re-ranked the top 1000 solutions by FiberDock. The graphs show the fraction of cases where a near native solution was present in the top 10, 50, 200 and 1000 solutions of PatchDock (shades of blue) and FiberDock (shades of red). The dark red/blue bars show the fraction of cases with medium accuracy solutions (LRMSD < 5 Å or IRMSD < 2 Å) and the light bars show the fraction of the additional cases with acceptable solutions (LRMSD < 10 Å or IRMSD < 4 Å). The top-left graph shows the results of all the 88 cases, the top-right graph shows the results of the Antibody-Antigen (AA) cases, the bottom left shows the Enzyme-Inhibitor (EI) cases, and the bottom right the other cases. This experiment demonstrates the importance of the refinement and re-ranking stage by FiberDock. The figure was taken from Mashiach et al. 2010.

5.3 DISCUSSION

Recent CAPRI challenges have emphasized two key issues. The first is the importance of gathering information on the specific target prior to docking, which significantly increases the success rate of the docking. The second point is our limited ability to handle docking of unbound molecules with significant backbone flexibility in the interacting area or docking of homology models, which often have inaccurate backbone conformation.

In recent CAPRI rounds (13-19), we succeeded in our docking predictions in 6 out of 13 targets. In all of these cases we used biological information and bioinformatics analysis. For targets 32 and 40 we found biological information on the interface of both interacting proteins which drastically reduced the docking search space. In target 40, however, we also achieved a high accuracy result by our automatic FireDock server [165] without binding site information. In target 34 we used a distance constraint between the active-site of the methyltransferase and the methylated nucleotide in the RNA molecule. In target 37 we detected a conserved surface patch on one of the proteins and used it as the location of the interface. On the other hand, in
targets 29, 38 and 39 we used wrong biological assumptions which prevented us from predicting the correct structure of the complexes. When these wrong assumptions were removed we obtained correct docking solutions in 2 out of these 3 targets.

Docking challenges with homology models or with significant backbone movements in the interacting molecules were found to be the most difficult tasks. In target 30 we indentified the correct interface prior to docking, but a flexible loop in the interface prevented us from docking the proteins correctly. Target 33 required homology modeling of an RNA. Since the modeled conformation of one of its hairpins was inaccurate, the rigid-docking stage failed. In three other targets (35, 36, 38) we also failed due to inaccurate homology modeling.

Each CAPRI target reveals strengths and weaknesses of the methods we use, and guides us in developing new methods to face similar challenges more successfully. In the initial rounds of CAPRI, when our major tool was the PatchDock algorithm, we were quite successful in detecting acceptable solutions, but were less successful in detecting high and medium quality solutions [64]. This motivated the development of the FireDock refinement algorithm, which proved its efficacy in the currently reported rounds. As for now, the major obstacle in the docking field is to handle backbone conformational changes which occur in the interface. Motivated by this challenge, we recently developed the FiberDock method which mimics an induced-fit process and models backbone movements during the refinement of rigid docking solutions. FiberDock showed superior results over FireDock in cases with interface backbone flexibility [179], and we hope to see its contribution in the next CAPRI rounds. All our docking methods are rapid and efficient and most of them are publicly available as web servers on our website: http://bioinfo3d.cs.tau.ac.il/.
Chapter 6: Conclusions

Handling protein flexibility has always been a major challenge in the docking field. The CAPRI challenges and many benchmark tests showed satisfying results in modeling side-chain flexibility. However, no docking method that reliably models backbone flexibility has yet been developed. Nevertheless, there has been a great progress in this direction in the last few years. Some methods model backbone flexibility prior to the docking, according to protein flexibility analysis, to generate an ensemble of possible conformations. These conformations are then used for ensemble docking or for docking using mean field approach [83,85]. Other methods are specific for modeling hinge motions [86,87,88,89,90]. Relatively small backbone movements, with RMSD below 5Å, can be handled by performing soft rigid docking, followed by flexible refinement. Since backbone flexibility adds a huge number of degrees of freedom to the search space, refinement methods often limit the modeled flexibility of the docked proteins. Refinement by molecular dynamics is accurate but extremely inefficient and time consuming. Therefore it is used to model very subtle movements [91]. When brute-force methods, such as Monte-Carlo simulation [95] and genetic algorithm [94], are used for refinement, the flexible regions are usually limited to the interface between the docked proteins, to reduce the search space. Normal mode based refinement methods usually limit the modeled motions to the 5-10 lowest frequency normal modes, i.e. to global conformational changes [53,92]. In this work we chose to tackle this challenging task of flexible refinement and develop new methods that do not limit the backbone flexibility of the docked proteins.

We have developed FiberDock - a novel flexible docking refinement method [179,197]. FiberDock models both backbone and side-chain movements and refines the relative rigid-body orientation of the proteins. The refined solutions are then re-ranked in order to identify the near native models. The side-chain movements are modeled by a linear programming approach that finds the optimal combination of rotamers for the interface residues. The backbone flexibility is modeled by a new normal-mode based approach that uses both low and high frequency normal modes, and thus enables both global and local conformational changes. The method was thoroughly tested on a benchmark that includes cases in which backbone conformational changes occur upon binding. FiberDock performance was compared to other state-of-the-art docking refinement methods (FireDock and RosettaDock) and achieved superior results. Moreover, the results showed that the method successfully models backbone movements that occur during molecular interactions.

In nature, the majority of protein-protein complexes are symmetric multi-molecular complexes (homo-dimers, homo-trimers, etc.) [168,169]. Their symmetry can be exploited by docking methods for achieving more accurate predictions. FiberDock is suited only for complexes that are composed of two proteins without symmetry constraints. Therefore, we have developed the SymmRef algorithm, a novel method for refinement and re-ranking of symmetric multi-molecular docking solutions [198]. The method models backbone and side-chain movements and optimizes the rigid-body orientation of the monomers, while preserving the symmetry of
the initial docking model. Both FiberDock and SymmRef were incorporated into our full docking protocol which was used in recent CAPRI challenges and achieved impressive results [220].

While our methods significantly improve the accuracy of docking models, their main contribution is the ranking improvement (Figure 5.1). Docking solutions are usually ranked by an energy based scoring function which includes many physico-chemical terms such as van der Waals energy, electrostatics, hydrogen bonds, and more. The accuracy of these energy calculations strongly depends on the accuracy of the molecular model. Therefore, the refinement of the near-native models directly influences their ranking.

Another important aspect of our methods is their high efficiency. The average running time of refining a single docking solution is 14 seconds for FiberDock and 23 seconds for SymmRef, much faster than other refinement methods. The refinement process can easily be parallelized to achieve even better performances. High efficiency is very important. It enables us to pass more rigid-docking solutions to the refinement stage and thus improve our chances of finding a near native solution. In cases where we suspect large conformational changes, we can pre-generate a set of possible conformations and perform an ensemble docking. Ensemble docking generates many more solutions that must be refined and re-scored by an efficient method. Finally, short running-time refinement enables us to search for energy funnels around the top ranked solutions to pinpoint the most accurate models among them.

We believe that the research described in this thesis significantly contributes to the structural bioinformatics research. Since FiberDock and SymmRef were published, some newly developed methods in structural bioinformatics have incorporated them into their algorithm. FiberDock was used as a part of a docking-based hierarchical folding scheme for protein structure prediction [221]. It has also been incorporated into a novel template-based docking protocol [222]. SymmRef was proposed to be used for refinement of the results of symmetric molecular fitting into electron microscopy density maps [223]. Furthermore, the free and user-friendly webserver of FiberDock has exposed the method to the biological community. Indeed, it has already been used for many important biological studies [224,225,226,227,228,229,230]. We hope and believe that these methods will continue to contribute to the research of protein-protein interactions and drug discovery.
References


