15 Structure Prediction of Protein Complexes

Brian Pierce, Andrew T. Phillips, and Zhiping Weng

15.1 Introduction

Protein–protein interactions are critical for biological function. They directly and indirectly influence the biological systems of which they are a part. Antibodies bind with antigens to detect and stop viruses and other infectious agents. Cell signaling is performed in many cases through the interactions between proteins. Many diseases involve protein–protein interactions on some level, including cancer and prion diseases.

A very useful source of information about the interaction between two proteins is the 3D structure of their macromolecular complex. Using this, it is possible to easily identify such details as the residues that are directly involved with binding, the nature of the interface itself, and the conformational change undergone by the protein partners. X-ray crystallography has provided us with many structures of protein–protein complexes, but time and experimental limitations have left many protein complex structures yet unsolved. While techniques such as homology modeling and mutational analysis yield some information about binding sites, more information is needed to understand the nature of protein–protein interactions. Thus, it is useful to employ protein docking to predict the complex structure.

15.1.1 Protein Docking: Definition

Protein–protein docking can be defined as the determination of the complex structure between two proteins, given the coordinates of the individual proteins. Protein–protein docking can be further classified as bound docking, which uses the structures from the complex structure as input, and unbound docking, which uses the structures from the individually crystallized subunits as input. Obviously the former has little predictive use, but it is useful for verification and testing. The focus of this chapter will be unbound docking, where docking research is currently focused.

Other types of docking being studied and modeled are protein–DNA docking, protein–RNA docking, and protein–small molecule docking, which are similar in general but due to the nature of the binding partners, the heuristics and approaches end up differing for these cases. More information on protein–small molecule docking can be found in Chapter 16. For the remainder of this chapter, protein–protein docking will simply be referred to as protein docking.
15.1.2 **Protein–Protein Interactions: Underlying Principles**

There are many reviews that summarize the existing data regarding protein–protein recognition (e.g., Jones and Thornton, 1996). In addition, the Recommended Reading section at the end of this chapter includes some relevant works.

The kinetics of forming a protein–protein complex can be modeled with a two-step pathway, where the free proteins first form an encounter complex, then if the encounter complex is adequately similar to the actual complex (i.e., the short-range energies are favorable), the complex is formed. The encounter complex is generally guided by steering through long–range electrostatics, while the step between encounter complex and actual complex arises from short-range specificities such as van der Waals, hydrophobicity, and hydrogen bonding (Janin, 1997).

The energy change during complexation can be approximated as

\[ \Delta G_{\text{binding}} = \Delta G_{\text{elec}} + \Delta E_{\text{vdW}} + \Delta G_{\text{des}} + \Delta E_{\text{int}} - T \Delta S_{\text{sc}} - T \Delta S_{\text{bb}}. \]

\( \Delta G_{\text{elec}} \) and \( \Delta E_{\text{vdW}} \) are the electrostatic and van der Waals energy changes, while \( \Delta E_{\text{int}} \) refers to energy changes due to conformational changes upon binding. \( \Delta G_{\text{des}} \) is the energy change from desolvation, while \(-T\Delta S_{\text{sc}}\) and \(-T\Delta S_{\text{bb}}\) are the energy contributions from side chain and backbone entropy, respectively. See Chapter 2 for a more detailed discussion of physical force fields.

15.1.3 **Will the Proteins Interact?**

It is important to point out that an underlying assumption employed during protein docking is that the proteins will interact to form a complex. Experimental approaches, such as the yeast two-hybrid, are generally employed first to determine whether or not there is an interaction. Table 15.1 lists various experimental approaches for determining whether proteins will interact.

Protein docking will generally output its best predictions for what the structure of the complex would look like, along with scores for these predictions. As these scores are heuristic, they are generally not energies and would not be a good criterion for determining whether or not there will be an interaction.

While not strictly protein docking, one study has recently addressed the issue of whether two proteins will interact, given their structures as input (Carugo and Franzot, 2004). It involved measurement of shape complementarity between surface patches on different proteins, and the results were promising.

Given that two proteins interact, there is also the issue of whether their complex is obligatory or transient. Obligatory complexes only exist in complex form, and the chains are never seen individually. Correct folding may require the other members of the complex, for instance. This class of complex is only useful for bound docking, as the structures of the unbound proteins are not available. Transient complexes are known to exist both in the complex form and as individual subunits in vivo. They encompass a vast range of complexes, including enzyme/inhibitor, antibody/antigen,
15. Structure Prediction of Protein Complexes

Table 15.1 Some experimental techniques used to determine the existence of protein interactions

<table>
<thead>
<tr>
<th>Method</th>
<th>Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Yeast two-hybrid</td>
<td>“Analyzing protein–protein interactions using the two-hybrid system” (Bartel and Fields, 1995)</td>
</tr>
<tr>
<td>Tandem affinity purification</td>
<td>“The tandem affinity purification (TAP) method: a general procedure of protein complex purification” (Puig et al., 2001)</td>
</tr>
<tr>
<td>Mass spectrometry</td>
<td>“Systematic identification of protein complexes in S. cerevisiae by mass spectrometry” (Ho et al., 2002)</td>
</tr>
<tr>
<td>Immunoprecipitation</td>
<td>“Complex formation between the UL16 and UL21 tegument proteins of pseudorabies virus” (Klupp et al., 2005)</td>
</tr>
<tr>
<td>Pulldown assay</td>
<td>“Protein interactions among the vaccinia virus late transcription factors” (Dellis et al., 2004)</td>
</tr>
<tr>
<td>Phage display</td>
<td>“Protein–protein interactions in hematology and phage display” (Mullaney and Pallavicini, 2001)</td>
</tr>
<tr>
<td>Protein chips</td>
<td>“Global analysis of protein activities using proteome chips” (Zhu et al., 2001)</td>
</tr>
</tbody>
</table>

and viral fusion proteins (for example). Predicting the structures of these complexes is the goal of protein docking.

15.1.4 Input Structures

After two proteins are known to interact, docking them requires the structures of the individual proteins. As most docking algorithms rely on atom-level representations of the structures, X-ray crystal structures of the proteins are most useful, as they can provide single structures (versus multiple possible models) at high resolution (<2.5 Å).

In the absence of X-ray crystal structures, other means can be utilized to obtain structures for inputs for protein docking. Protein structures determined through nuclear magnetic resonance (NMR) can sometimes be used, with some caveats. As NMR structures are provided with different conformations, one generally needs to choose a particular conformation for the docking input. If surface residues or key residues exhibit much movement between the conformations, choosing a particular model might lead to erroneous docking results. Alternatively, docking each NMR conformation could be attempted, but this would take longer and require sorting through many more predictions.

In addition to NMR structures, homology models, in particular those constructed from X-ray crystal structures and with high sequence similarities from the template structure, can also be used. With the variety of modeling tools available, this is becoming a fairly common and useful tool (Tovchigrechko et al., 2002), but should be used if there is no X-ray structure of the input protein and with caution regarding the docking results.
15.1.5 History of Docking

The first computational protein docking tools were developed in the late 1970s. Greer and Bush (1978) introduced a grid-based measure of complementarity between molecules, and used it to score interfaces between hemoglobin subunits. An early docking study by Wodak and Janin (1978) used a “simplified protein model” with one sphere per amino acid, which they used to dock BPTI to trypsin. The search involved rotating BPTI and varying its center-of-mass distance with trypsin.

During the 1980s and early 1990s, more advanced shape complementarity metrics were produced. A shape matching program for small-molecule binding was developed by Kuntz et al. (1982) in their program DOCK, which used spheres of different radii to represent the small molecule and ligand to find their optimal fit. This approach was later adapted by Shoichet and Kuntz for protein–protein docking (Shoichet and Kuntz, 1991). A means of mathematically describing curvature as “bumps and holes” on a protein surface provided the basis for possible ways to find complementary surfaces (Lee and Rose, 1985). Connolly determined a means of calculating a smooth surface around a macromolecule based on solvent accessibility (Connolly, 1983), and later used “knobs and holes” matching of surfaces as part of a docking procedure (Connolly, 1986).

As docking progressed, some groups used grids for discretizing the structures for their docking searches. An early docking effort (Zielenkiewicz and Rabczenko, 1984) projected molecular properties onto a 2D grid, searching for optimal translations, and applied this to predicting the structure of the insulin hexamer. Jiang and Kim used cubes on a grid to represent surfaces, and performed soft docking (Jiang and Kim, 1991). A notable improvement to the grid search was use of the fast Fourier transform (FFT) to perform correlations in grid-based translational searching (Katchalski-Katzir et al., 1992). This, coupled with advances in computer speed, allowed for more energy terms and greater resolution in docking searches, which in turn improved docking results. Currently there are many docking tools available for predicting protein complexes, reflecting many useful and innovative developments. Though it is not possible to include every available tool (due to their amount, as well as new ones appearing), several docking servers and docking programs available on the web are listed in Tables 15.2 and 15.3.

15.2 Unbound Docking: Current Approaches

The most obvious way to dock two proteins would be to perform a full simulation of the proteins using basic physical principles (molecular dynamics) and see how they bind in this context. This would also be a good means of determining whether two proteins would interact, a topic described above. Unfortunately, this sort of simulation is computationally intractable, so simplified models have been developed to predict the binding mode in a meaningful manner without performing a full simulation.

Protein docking involves searching for regions of shape (and usually biochemical) complementarity, but (at least in the case of unbound docking) it must also allow for conformational change upon binding. Therefore, some overlap must be
15. Structure Prediction of Protein Complexes

Table 15.2  Protein docking servers

<table>
<thead>
<tr>
<th>Name</th>
<th>Address</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>ClusPro</td>
<td><a href="http://nrc.bu.edu/cluster">http://nrc.bu.edu/cluster</a></td>
<td>Allows docking with DOT and ZDOCK, clustering, symmetric multimers</td>
</tr>
<tr>
<td>ZDOCK</td>
<td><a href="http://zdock.bu.edu">http://zdock.bu.edu</a></td>
<td>Runs ZDOCK, has support for contacts/blocking</td>
</tr>
<tr>
<td>GRAMM-X</td>
<td><a href="http://vakser.bioinformatics.ku.edu/resources/gramm/grammxx/">http://vakser.bioinformatics.ku.edu/resources/gramm/grammxx/</a></td>
<td>FFT initial search followed by minimization and rescoring</td>
</tr>
<tr>
<td>PatchDOCK</td>
<td><a href="http://bioinfo3d.cs.tau.ac.il/PatchDock">http://bioinfo3d.cs.tau.ac.il/PatchDock</a></td>
<td>Also has symmetric multimer docking</td>
</tr>
<tr>
<td>Dcomplex</td>
<td><a href="http://phyz4.med.buffalo.edu/czhang/complex.html">http://phyz4.med.buffalo.edu/czhang/complex.html</a></td>
<td>No docking search; scores complex predictions</td>
</tr>
</tbody>
</table>

permitted, and flexibility must be taken into account. In order to adequately account for this, docking is typically broken up into two main stages (see Fig. 15.1). First is the initial stage, which involves a rigid body search with a “soft” scoring function that allows for overlap. This stage produces many possible predictions, typically on the order of $10^4$. These predictions are then assessed in the refinement stage, which performs small flexible motions of the side chain and/or backbone, and a detailed rescoring which will ideally filter out the false positives produced by the initial stage.

It should be noted that biological and experimental information is commonly incorporated into various stages of the docking process, when available. For example, alanine scanning (Clackson and Wells, 1995) and mutagenesis experiments can

Table 15.3  Protein docking tools on the Web

<table>
<thead>
<tr>
<th>Name</th>
<th>Address</th>
<th>Download$^b$</th>
</tr>
</thead>
<tbody>
<tr>
<td>AutoDOCK</td>
<td><a href="http://www.scripps.edu/mb/olson/doc/autodock/">http://www.scripps.edu/mb/olson/doc/autodock/</a></td>
<td>N$^a$</td>
</tr>
<tr>
<td>BiGGER</td>
<td><a href="http://www.cqfl.fct.unl.pt/bioin/chemera/Chemera/">http://www.cqfl.fct.unl.pt/bioin/chemera/Chemera/</a></td>
<td>N</td>
</tr>
<tr>
<td>ClusPro</td>
<td><a href="http://structure.bu.edu/Projects/PPDocking/cluspro.html">http://structure.bu.edu/Projects/PPDocking/cluspro.html</a></td>
<td>N</td>
</tr>
<tr>
<td>DOCK</td>
<td><a href="http://dock.compbio.ucsf.edu/">http://dock.compbio.ucsf.edu/</a></td>
<td>Y</td>
</tr>
<tr>
<td>DOT</td>
<td><a href="http://www.sdsc.edu/CCMS/DOT/">http://www.sdsc.edu/CCMS/DOT/</a></td>
<td>Y</td>
</tr>
<tr>
<td>FTDOCK</td>
<td><a href="http://www.bmm.icnet.uk/docking/ftdock.html">http://www.bmm.icnet.uk/docking/ftdock.html</a></td>
<td>Y</td>
</tr>
<tr>
<td>GRAMM</td>
<td><a href="http://vakser.bioinformatics.ku.edu/resources/gramm/">http://vakser.bioinformatics.ku.edu/resources/gramm/</a></td>
<td>Y</td>
</tr>
<tr>
<td>HADDOCK</td>
<td><a href="http://www.nmr.chem.uu.nl/haddock/">http://www.nmr.chem.uu.nl/haddock/</a></td>
<td>Y</td>
</tr>
<tr>
<td>HEX</td>
<td><a href="http://www.csd.abdn.ac.uk/hex/">http://www.csd.abdn.ac.uk/hex/</a></td>
<td>Y</td>
</tr>
<tr>
<td>ICM</td>
<td><a href="http://www.molsoft.com/docking.html">http://www.molsoft.com/docking.html</a></td>
<td>N$^c$</td>
</tr>
<tr>
<td>MolFit</td>
<td><a href="http://www.weizmann.ac.il/Chemical_Research_Support/molfit/">http://www.weizmann.ac.il/Chemical_Research_Support/molfit/</a></td>
<td>Y</td>
</tr>
<tr>
<td>PatchDock</td>
<td><a href="http://bioinfo3d.cs.tau.ac.il/PatchDock/patchdock.html">http://bioinfo3d.cs.tau.ac.il/PatchDock/patchdock.html</a></td>
<td>Y</td>
</tr>
<tr>
<td>RosettaDock</td>
<td><a href="http://graylab.jhu.edu/docking/rosetta/">http://graylab.jhu.edu/docking/rosetta/</a></td>
<td>Y</td>
</tr>
<tr>
<td>ZDOCK</td>
<td><a href="http://zlab.bu.edu/zdock/">http://zlab.bu.edu/zdock/</a></td>
<td>Y</td>
</tr>
</tbody>
</table>

$^a$ Whether the program is available for free download (academic users) via the website.

$^b$ License form must be downloaded and filled out, then will receive download information.

$^c$ Browser available for free; docking portion of the program must be purchased.
Fig. 15.1  The general protein docking algorithm. The initial stage performs a full coarse-grained search and outputs approximately 1000–10,000 predictions. The refinement stage then improves these predictions through energy minimization, followed by a more detailed rescoring (and possibly clustering by position and score). Ideally, the top scoring prediction output from the refinement stage will be similar to the correct complex.

identify residues that are involved in binding. Also, data from NMR experiments of the protein complex have been used (Clore, 2000; Clore and Schwieters, 2003; Dominguez et al., 2003). These data can be used to constrain/score the initial search, or as part of a filter later on.

15.2.1 Rigid Body Docking: Search

A common and effective means of performing protein docking is to treat the proteins as rigid bodies, which allows for a fast and efficient search. To search the rigid-body degrees of freedom, it is necessary to explore a six-dimensional space: three translational degrees and three rotational degrees. The size of most proteins causes this space to be quite large, particularly when it is needed to sample at the atomic level.
(for example, most algorithms sample with spatial increments of ~1.2 Å). Therefore, many interesting ways of searching the rigid body space have been developed over the years, in an attempt to intelligently sample what is necessary to make correct predictions without taking prohibitively long.

15.2.1.1 Fast Fourier Transform

One popular means of quickly searching for the best docking conformation is using the FFT (Press, 2002). To perform the search, the ligand and receptor are discretized onto separate three-dimensional grids. To compute the score for a particular \( x, y, z \) translation \( (i, j, k) \), the grids are overlaid with this offset and the overlapping cells are multiplied together, the sum of these products representing the score for that particular translation:

\[
S(i, j, k) = \sum_{x, y, z} L(x + i, y + j, z + k) \times R(x, y, z).
\]

As to perform a global translational search this score must be computed for all values of \( i, j, k \), the operation ends up costing \( O(n^6) \), with \( n \) being the number of grid points in each dimension. By using an FFT, the search becomes encoded into the correlation between the grids:

\[
S = \frac{1}{N^3} IFT(IFT(R) \times DFT(L)).
\]

This way the entire scores grid is computed at once by a series of inverse Fourier transforms and discrete Fourier transforms. Each Fourier transform is \( O(\log(n^3)) = O(\log(n)) \), while the multiplication of the transformed grids in inverse space is still \( O(n^3) \). Therefore, the running time goes from \( O(n^6) \) to \( O(n^3\log(n)) \). Tools that use the FFT for their translational search include FTDock (Gabb et al., 1997), GRAMM (Vakser, 1995), and ZDOCK (Chen et al., 2003). In addition, the HEX program (Ritchie and Kemp, 2000) uses the FFT for searching angular space, discretized into spherical harmonics (while performing a full search in translational space).

While the FFT can mean improvements in running time, there are several disadvantages to using this approach. One is that the FFT automatically computes the scores for all points in the grid, although at many points the scores may not be needed as they involve either heavy clash or no contact. In other words, the exhaustive search is performed no matter what. Also, the spacing between the grid points needs to be regular; the resolution cannot change based on surface topology of the molecules.

15.2.1.2 Other Search Techniques

Another technique that has been used (Palma et al., 2000) involves discretizing the proteins onto grids and searching the translations in real space, without an FFT.
This is done efficiently by using grids of bits (0’s and 1’s) to represent the proteins, speeding up the computation significantly over using floats or integers.

Geometric hashing is another algorithm used to sample the search space for protein complexes (Fischer et al., 1995). This was originally a tool used for object recognition in computer vision (Lamdan et al., 1990). In the context of protein docking, the proteins are represented by the relative positions of atoms (or surface characteristics such as knobs and holes) in various reference states. Then, using these hash tables, the proteins are compared to see if the characteristics correlate highly between the two proteins in a particular reference state. One advantage of this algorithm is that the learning stage (construction of the hash tables) can be computed separately for the proteins to be docked; the matching itself just involves using the precomputed hash tables.

Some groups use Monte Carlo sampling in 6D space to perform the initial search (Gray et al., 2003; Zacharias, 2003). Monte Carlo searches involve taking random samples in 6D space, usually constrained by geometric/distance criteria. These random predictions have their energies evaluated by the scoring function to see whether the sample is kept or discarded. A variant of this approach is also found in the pseudo-Brownian rigid-body docking of the ICM program (Abagyan et al., 1994).

Another technique for performing a rigid-body search in 6D space is genetic algorithms (GAs) (e.g., see Gardiner et al., 2001). The six coordinates are encoded on a chromosome and the standard GA search is performed, using the energy score as the fitness parameter. Random starting positions are used to obtain the initial set of chromosomes. The program AutoDock (Morris et al., 1998) uses random positions and orientations for its global search, then applies simulated annealing or genetic algorithms (depending on user preference) to perform a search in local translational/rotational space.

15.2.2 Rigid Body Docking: Scoring

Scoring for rigid-body docking must be both easy to evaluate (sometimes fitting a particular form, depending on the search scheme) and accurate. In particular, it is important to have some correct predictions among the predictions that are produced. Then the refinement stage can ideally discriminate these predictions from the false positives. The main scoring terms employed during initial stage docking are shape complementarity, electrostatics, and desolvation.

15.2.2.1 Shape Complementarity

Shape complementarity is the most popular scoring term used for protein docking, as it is simple yet generally accurate. The physical basis for shape complementarity is the van der Waals potential, of which the Lennard-Jones 6-12 potential is a commonly used case:

\[ V_{L-J} = \frac{A}{r^{12}} - \frac{B}{r^{6}}. \]
Fig. 15.2  The Lennard-Jones potential \( (V) \) as a function of the distance \( (r) \) between two atoms. The distance \( r_{vdW} \) is the sum of the van der Waals radii for the atoms, while the dotted line gives an example of how this potential can be “softened” to avoid large penalty for atomic clashes.

A plot of this function is shown in Fig. 15.2. The energy minimum is at the sum of the van der Waals radii \( (r_{vdW}) \), while having the atoms much closer results in a strong repulsive energy (dominated by the \( r^{-12} \) term). Details about this (and other physical energy terms) can be found in Chapter 2.

It was noted (Betts and Sternberg, 1999) in a study of conformational changes upon binding that many complexes have a “lock and key” recognition and that the change is small. However, the complexes considered were primarily enzyme–inhibitor and it was acknowledged that for other types of complexes the change may be greater. Another study confirmed this finding (Norel et al., 1999), that shape complementarity is a primary contributor to a successful docking energy function.

A key consideration in evaluating shape complementarity for unbound docking is to have a “softness” to allow for a small amount of overlap in the predictions. This is illustrated in Fig. 15.2 by the dotted line, and results in a reduced clash penalty at close distances. In this manner, the RosettaDock program (Gray et al., 2003) employs a short-range linearization of the LJ potential for its scoring function. For rigid body docking, some overlaps can be acceptable as they can represent conformational changes (either side chain or backbone) during complexation. Shape complementarity has been implemented for both explicit (atom-based) models of proteins as well as grid-based docking methods.
15.2.2.2 Electrostatics

Many groups use electrostatics as a part of their energy functions. Electrostatics involves a solution of the Poisson–Boltzmann equation (Honig and Nicholls, 1995):

\[ \nabla \cdot (\varepsilon(r) \nabla \phi(r)) - \varepsilon(r) \kappa^2(r) \sinh(\phi(r)) + \rho(r) = 0. \]

In this equation, \( \varepsilon \) is the dielectric term, \( \phi \) is the electrical potential, \( \rho \) is the charge density, and \( \kappa \) is the charge screening parameter for mobile ions. Once the potential \( \phi \) is solved, the electrostatic energy of the charge distribution of the macromolecule can be computed. In order to make the solution easier to compute, this equation is sometimes simplified, e.g., by ignoring mobile ions and eliminating the middle term. Also, the dielectric term can be said to be invariant in certain regions (e.g., for the interior of the protein), leaving Poisson’s equation:

\[ \nabla^2 \phi(r) = -\frac{\rho(r)}{\varepsilon}. \]

The force from this potential is the familiar 1/r^2 force from Coulomb’s law.

For initial stage docking, electrostatics is often combined with other terms to make up the scoring function. The docking program DOT, for example, uses a sum of Poisson–Boltzmann electrostatics and van der Waals energy; this led to successful predictions for several complexes (Mandell et al., 2001). A notable example of the use of electrostatics in docking is the use of an electrostatic potential for initial stage docking where electrostatic forces guide the molecules together via force field trajectories (Fitzjohn and Bates, 2003). As the electrostatics force falls off more slowly than other forces involved in docking (1/r^2), it is seen as important in long-range steering of the molecules before binding. It should also be noted, though, that electrostatics is used in docking refinement as well.

15.2.2.3 Desolvation and Statistical Potentials

Desolvation in protein binding is the energy needed to change water–protein bonds with bonds between proteins. Desolvation energy is also known as the hydrophobic effect, which has long been known to play an important part in protein binding (Chothia and Janin, 1975). One way that this is modeled is the atom-based Atomic Contact Energy (ACE) (Zhang et al., 1997), which defines 18 atom types and determines their contact energies based on what is seen in known structures. As it is based on the statistics of protein structures, it is known as a statistical potential. An earlier example of this is the work of Miyazawa and Jernigan, who developed a statistical potential for residue–residue contacts (Miyazawa and Jernigan, 1996).

Other statistical functions are used for evaluating docking predictions. One function with some success is the DFIRE (distance-scaled, finite, ideal gas reference) function (Zhou and Zhou, 2002), which is generated from crystal structures of monomeric proteins, using distance cutoffs to bin the atoms seen within certain
15. Structure Prediction of Protein Complexes

Distances. It is more advanced than a function such as ACE, which only counts whether or not there is a pair within a particular distance cutoff. DFIRE has been applied to both protein folding and protein docking; a recent version of DFIRE has proven rather successful in discriminating and ranking structures from various decoy sets (Zhang et al., 2004). The DFIRE potential has also been incorporated into a Web-based server, Dcomplex (see Table 15.2). For a discussion of empirical scoring functions as they apply to protein modeling in general, see Chapter 3.

15.2.2.4 Hydrogen Bonding

Hydrogen bonding has been determined to be a good determinant of specificity of binding and thus is useful as a filter and in the refinement stage. However, its geometric nature (as it is based on a dipole interaction) makes it difficult to integrate into an easily calculable energy function (for initial stage FFT searching in particular). In one case, a hydrogen bonding potential energy function was constructed based on existing PDB structures, with good success in discriminating protein docking predictions (Kortemme et al., 2003).

Other methods can be used to filter the output from a docking run or to rank predictions. Computational predictions of binding sites can be used to find likely binding sites on individual proteins. Mutational data can yield what residues are important in an interaction, thus indicating what residues are at or near the interface of a protein complex. One docking program, HADDOCK, or High Ambiguity Driven protein–protein Docking, explicitly uses residue–residue contact data from mutagenesis and NMR experiments as a scoring term in its docking procedure (Dominguez et al., 2003). This approach has been shown to be effective in unbound docking of a number of complexes.

15.2.3 Refinement

After the full initial search is performed, the predictions from this search are refined with small movements, sometimes using 6D rigid body movements, and sometimes using molecular mechanics software such as CHARMM (Brooks et al., 1983). At this stage, more detailed energy models for the molecules can be used, with refined versions of the terms described above, and introduction of terms such as hydrogen bonding.

15.2.4 Clustering

The energy landscape of protein association includes an energy funnel at the binding configuration (Li et al., 2003), which makes complexation possible. Therefore, many low-energy configurations can be seen at or near a binding site. The idea behind clustering is that a global search for possible complexes would have a greater density of low-scoring predictions near the binding site. So a large cluster of predictions with good scores may be more likely to be in the vicinity of the binding site.
Clustering has been implemented by a number of groups, notably Comeau et al. (2004). In particular, clustering involves computing the pairwise RMSDs (root mean square deviation; see Section 15.3) between the ligands in the predictions, after superposing the receptors. Then using a specific distance cutoff (typically 7–9 Å), these predictions are combined into clusters that are ranked according to population size. Other terms such as the score of the top scoring prediction in the cluster can be used for evaluation of the cluster as well.

15.2.5 Side Chain Searching

The side chains of proteins are often of interest in protein docking; particularly the larger chains can change conformation and have significant impact on complementarity with the protein partner and binding affinity. Often, groups use rotamers (Dunbrack, 2002) to model side chain positions of amino acids. These rotamers are preferred states for side chains that help to narrow the number of side chain conformations sampled.

Even using rotamers, the number of possible side chain conformations in the interface takes prohibitively long to sample for many docking predictions. Various groups have addressed this in their side chain searching algorithms. For example, the program SCWRL uses a graph-theoretical approach to speed up the search for the optimal side chain configurations (Canutescu et al., 2003).

15.2.6 Backbone Searching

While side chain searches are able to model the small-scale motions of proteins upon docking, motions of the protein backbone are of relevance in certain complexes. For example, the backbone of the actin–gelsolin interface (Choe et al., 2002) moves over 6.5 Å upon binding. This presents a challenge to protein docking, as there are many degrees of freedom involved and the energy model must be adequate to choose the correct backbone conformation.

Flexibility has been explored in other types of docking previously, using normal modes in a protein–DNA docking study (Zacharias and Sklenar, 1999), and ensembles of protein structures in docking a small molecule to a protein receptor (Knegt et al., 1997). Another study involved the use of a hinge motion to model the receptor closing in on a ligand (Sandak et al., 1998). Overall, the modeling of backbone flexibility in protein docking is still being explored, and future studies will no doubt address this issue further.

15.3 Evaluation of Docking Algorithms

15.3.1 Determining Accuracy of Predictions

Once a protein docking tool has made a prediction, it is useful to quantitatively evaluate this prediction so that its accuracy can be measured. The most popular
15. Structure Prediction of Protein Complexes

means of measuring the accuracy of predictions is the root mean square deviation (RMSD) between the prediction and the crystal structure of the complex. Here is the equation for the 3D RMSD between the prediction \( p \) and reference \( r \) coordinates:

\[
RMSD = \sqrt{\frac{\sum_{i=1}^{n}(x_p(i) - x_r(i))^2 + (y_p(i) - y_r(i))^2 + (z_p(i) - z_r(i))^2}{n}}.
\]

Before the RMSD is calculated, the structures are optimally aligned in some fashion (typically with an algorithm that minimizes RMSD). For instance, the receptors are aligned and the ligand RMSD is calculated, or the ligand–receptor interface is aligned (using the list of interface atoms from the crystal structure) and the ligand and receptor interface RMSD is calculated. Finally, only the backbone or C\(\alpha\) atoms can be used for RMSD calculations (instead of all atoms), in order to place less stress on the side chains.

Another way to evaluate the accuracy of a docking prediction is the fraction of native and nonnative contacts, \( f_{\text{nat}} \) and \( f_{\text{non-nat}} \). \( f_{\text{nat}} \) is the fraction of native contacts, the number of residue contacts that are in the prediction divided by the number of contacts in the crystal structure. \( f_{\text{non-nat}} \) is the fraction of nonnative contacts in the prediction, which is the number of contacts in the prediction not in the reference, divided by the total number of contacts in the prediction. Both of these values can vary between 0 and 1.0. In the CAPRI experiment, \( f_{\text{nat}} \) and \( f_{\text{non-nat}} \) and ligand and interface RMSDs are used to evaluate the predictions.

15.3.2 Docking Benchmark

To provide the docking community with a set of test cases to test their algorithms, a protein docking benchmark has been produced (Chen et al., 2003b), known as Benchmark 1.0. This benchmark consists of 59 test cases, of which 22 are enzyme–inhibitor, 19 are antibody–antigen, 11 other, and 7 difficult. For all of the test cases (of which 31 are unbound–unbound and 28 are unbound–bound), the input PDB files and the crystal structure of the complex is provided. Various groups have used this benchmark to test the efficacy of their predictive algorithms (Duan et al., 2005; Gray et al., 2003; Smith et al., 2005).

There is now a newer protein docking benchmark available, Benchmark 2.0 (Mintseris et al., 2005). It includes 84 test cases, separated into rigid-body, medium, and difficult, according to the amount of conformational change between unbound and bound forms of the receptor and ligand. The rigid-body and some of the medium cases are within the predictive scope of rigid-body docking algorithms. However, the difficult cases would need to have their backbone motion accounted for by a docking algorithm to have successful predictions produced.

Both docking benchmarks are available on the Web: http://zlab.bu.edu/zdock/benchmark.shtml.
15.3.3 CAPRI Experiment

In order for groups to test their protein docking methods and compare their results with other groups, the CAPRI (Critical Assessment of PRediction of Interactions) experiment was started in 2001 (Janin et al., 2003). It was modeled after CASP, which is the analogous competition for protein folding, which started in 1994. So far, 17 targets (ligand–receptor pairs to be predicted, and one symmetric trimer) have been employed over five rounds. The most recent round featured approximately 30 international groups participating.

CAPRI is a blind competition, where the structure of the complex is not released until after the submissions of that target are due. Each group submits their 10 top predictions, ranked in order of confidence. The evaluation of the submissions is based on a combination of ligand RMSD, interface RMSD, native and nonnative contacts for each prediction (Mendez et al., 2003).

15.4 Case Study: ZDOCK and RDOCK

To illustrate a docking protocol, an example will be given: the initial-stage docking program ZDOCK (Chen et al., 2003) and the refinement program RDOCK (Li et al., 2003). In addition, our recent work in 6D rigid refinement of docking predictions will be presented (Pierce and Weng, unpublished).

15.4.1 ZDOCK Algorithm

ZDOCK is an initial-stage docking program. It uses an FFT-based grid search to scan for optimal translational conformations. The algorithm for ZDOCK is as follows:

1. Parse in the receptor and ligand files.
2. Discretize the receptor.
3. Rotate the ligand and discretize.
4. Compute the FFT to obtain a correlation between the discretized ligand and receptor.
5. Record the position and score of the grid point with the maximum score.
6. Repeat steps 3–5 for the remainder of the Euler angles.

Also, prior to running ZDOCK it is necessary to preprocess the ligand and receptor files with the supplied program mark_sur to mark the surface atoms as well as assign the ACE types to each of the atoms.

One useful feature of ZDOCK is that it allows the user to “block” residues, making it unfavorable for selected residues to be in the interface. This can be used, for example, to constrain the antigen for an antibody–antigen complex to be bound to the CDR loops of the antibody. This is also useful when experimental and/or biological information is known regarding the location of the binding site.
15. Structure Prediction of Protein Complexes

15.4.1.1 Angular Search
To explore the rigid-body rotational space of the ligand (with respect to the fixed receptor), ZDOCK uses evenly distributed Euler angle sets. Two angle sets are used, representing different sampling densities: 15° sampling (3600 angles) and 6° sampling (54,000 angles). Using 15° sampling, ZDOCK 2.3 takes an average of 4 minutes for docking Benchmark 1.0 proteins on a 16-processor IBM-SP4.

15.4.1.2 ZDOCK Scoring
The scoring component of ZDOCK has evolved; currently there are three versions of ZDOCK available, each with its own scoring scheme:

- ZDOCK 1.3: GSC, desolvation, electrostatics
- ZDOCK 2.1: PSC
- ZDOCK 2.3: PSC, desolvation, electrostatics

GSC and PSC are different shape complementarity scoring schemes. PSC differs from GSC in that it uses the number of atoms within a particular radius rather than just the presence or absence of atoms for the favorable component of its function. In terms of performance, PSC yielded a significant improvement for ZDOCK on Benchmark 1.0. Desolvation is calculated with ACE, and electrostatics is calculated using Coulomb’s equation with a distance-dependent dielectric constant.

15.4.2 RDOCK
To complement the initial-stage docking of ZDOCK, the RDOCK protocol was developed to perform refinement of initial-stage predictions (Li et al., 2003). It is composed of two parts: energy minimization, followed by reranking.

15.4.2.1 RDOCK: Energy Minimization
The energy minimization in RDOCK is performed using the molecular dynamics program CHARMM (Brooks et al., 1983), using a three-step procedure:

1. Remove clashes: 50 steps of minimization of vdW energies.
2. Optimize polar interactions: minimize electrostatics, vdW and internal energies for 60 steps, keeping ionic side chains neutral to prevent their dominating the energy.
3. Optimize charged interactions: 20 steps of minimizing total energy without any constraints.

As discussed by the RDOCK authors, the key point in the minimization stage is to only employ full electrostatics for only a few steps at the end. Otherwise the electrostatics term dominates the energy and causes poor results.
15.4.2.2 RDOCK: Scoring

After minimizing the energies and removing clashes, the RDOCK algorithm reranks the minimized structures. This scoring function is a simple combination of desolvation and electrostatics:

\[ \Delta G_{\text{binding}} = \Delta G_{\text{ACE}} + \beta \times \Delta E_{\text{elec}}. \]

The scaling factor \( \beta \) is set to 0.9 by default. The electrostatics term is obtained from CHARMM, while the ACE value is computed by summing over the interacting atom pairs of the prediction.

By using the minimization and ranking of RDOCK, significant improvements over ZDOCK performance are seen in protein–protein docking Benchmark 1.0. The details are included in the original RDOCK publication (Li et al., 2003).

15.4.3 6D Refinement

As an alternative to using molecular dynamics programs, it is also possible to rigidly “jiggle” the ligand of a prediction in 6D space to improve the prediction. Kuntz et al. have performed an early version of this rigid-body minimization with their DOCK program (Meng et al., 1993). This is advantageous in that it is in some senses simpler: the bounds of the search can be constrained, the degrees of freedom are far fewer, and the scoring function can be optimized for evaluating transient protein complexes. We have developed a program to perform this refinement, optimizing a scoring function based on a set of complexes and then performing searching in a local translational/rotational space using this optimized function.

15.4.3.1 Development of a Scoring Function

The scoring function has seven terms that are weighted linearly: van der Waals attractive and repulsive, electrostatics short range attractive and repulsive, electrostatics long range attractive and repulsive, and desolvation. The vDW terms and electrostatic terms come generally from the energy function of Gray et al. (2003), though the electrostatic term has been modified to include partial charges from CHARMM for short-range interactions. The desolvation term comes from ACE (Zhang et al., 1997).

These terms are weighted by a set of energy term weights (ETWs). To optimize the ETWs, complexes from complexes of Benchmark 2.0 were perturbed in local space, generating values for each energy term and an RMSD from optimal position for each perturbation. Using a 7D downhill simplex (one dimension for each energy term), ETWs were found that simultaneously optimized the scoring function across all the perturbed complexes. Scoring function optimization was evaluated by the “slope score”, which is essentially the slope of the plot of score versus RMSD (see Fig. 15.3). Thus, a lower (more negative) slope score represents a more optimal scoring function.
15.4.3.2 Exploring the Search Space

Using the optimized scoring functions, near-native predictions from the ZDOCK 2.3 program were refined by rigid-body movements in 6D space (3 rotational degrees
Fig. 15.4  A representation of the CGU search methodology, shown here for a one-dimensional search. The energy function (solid line) is evaluated for various values of the position (x), and this is used to obtain a quadratic function (dashed line) that approximates the funnel-shaped energy minimum.

and 3 translational degrees). For this study, two different minimization algorithms were employed to search the rotational and translational space: a downhill simplex (Press, 2002) and a convex global underestimator (CGU) (Phillips et al., 1995). For each algorithm, the search space explored was 3.0 Å and 0.3 radian for spatial and angular dimensions, respectively.

The CGU uses quadratic functions to approximate the energy surface and search for a global minimum (Fig. 15.4). It has been applied to both protein folding and protein docking (Dill et al., 1997; Mitchell et al., 1999), with considerable success.

The other rigid-body search method involved Monte Carlo sampling in local space, followed by use of a downhill simplex. The Metropolis criterion (Press, 2002) was used with the score to determine when to run the simplex; this algorithm was thus dubbed MCMS (Monte Carlo minimization with a simplex).

15.4.3.3 Results

The results for this study are shown in Fig. 15.5. The vast majority of the structures, 37 out of 39 for CGU, and 36 out of 39 for MCMS, improved in quality upon 6D minimization. In addition, 9 of the predictions improved by over 1.0 Å during CGU minimization. Though both did quite well, the CGU seemed to perform better than the MCMS in terms of the quality of the results.

15.5 Summary/Future Directions

15.5.1 CAPRI Success/Lessons

The field of protein–protein docking has progressed greatly since researchers first sought to use computers to predict modes of protein interactions decades ago. One clear illustration of this is the success that predictors have had in the blind competition, CAPRI. Out of 17 targets, 12 were predicted at high quality by at least
Fig. 15.5  Ligand interface Cα RMSD changes for ZDOCK predictions after 6D rigid-body minimization. Near-native predictions (<6.0 Å ligand interface RMSD) for 39 Benchmark 1.0 test cases were minimized. The leftmost column in each case indicates the complexes that became worse upon minimization; clearly overall the predictions were improved. Top: using the convex global underestimator (CGU) for minimization. Bottom: using the Monte Carlo minimization with simplex (MCMS)
one group. In addition, the overall predictive success of groups has improved over the rounds, indicating progress in the algorithms and improved ability to select correct predictions from those algorithms (Janin, 2005).

Several lessons can be drawn from CAPRI about what works for docking and what does not (Wodak and Mendez, 2004). The use of known biological information to constrain or filter a docking search clearly helps, as the predictions for targets that had such data available had a better success rate. It is also clear from performance on some targets that backbone flexibility is an important next step in protein docking developments.

15.5.2 New Developments

In addition to the lessons learned, the CAPRI experiment has led to various specific improvements and developments in docking. For instance, CAPRI Target 10 required docking of a tick-borne encephalitis viral protein to form a symmetric trimer. This has led to various algorithms for protein docking specifically geared toward predicting symmetric multimers (Pierce et al., 2005; Schneidman-Duhovny et al., 2005). CAPRI has also been a useful testing ground for the many new protein docking servers that are available on the Web (Table 15.2).

There are several more exciting directions where protein docking is heading. One is the combination of binary docking predictions to produce predictions of macromolecular complexes (Inbar et al., 2005). In addition, homology modeling of proteins and using the models for docking is being actively explored (Tovchigrechko et al., 2002); although some models do not have atomic-level precision (which influences the success rate of most docking programs), this technique opens up many more possible complexes that can be predicted. In contrast to protein docking of models, several groups have used modeling of homologues onto known structures of complexes to predict interactions of proteins on a genomic scale (Aloy et al., 2004; Aloy and Russell, 2002; Lu et al., 2003).

The field of protein docking is rapidly evolving, with extensions of well-known methods being complemented by new docking techniques imported from other fields of research. Dockers are now poised to tackle more difficult problems, such as including more backbone flexibility in the searching algorithm. This will allow for prediction of more difficult test cases, and increase the scope of protein complexes that can be successfully predicted through docking. So far, the success seen to date is encouraging; clearly there is much more to be done that will yield meaningful results for years to come.

Recommended Reading

Books

Protein–Protein Recognition (Kleanthous, 2000)
An Introduction to Protein Architecture: The Structural Biology of Proteins (Lesk, 2001)
15. Structure Prediction of Protein Complexes

**Review Articles**

“Principles of protein–protein interactions” (Jones and Thornton, 1996)

“Prediction of protein–protein interactions by docking methods” (Smith and Sternberg, 2002)

“Protein–protein docking: Is the glass half-full or half-empty?” (Vajda and Camacho, 2004)

“Structural basis of macromolecular recognition” (Wodak and Janin, 2002)

“Principles of docking: An overview of search algorithms and a guide to scoring functions” (Halperin et al., 2002)


**References**


15. Structure Prediction of Protein Complexes


15. Structure Prediction of Protein Complexes


